Evaluating Strategy Exploration in Empirical Game-Theoretic Analysis

Yongzhao Wang
University of Michigan
wangyzh@umich.edu

Qiurui Ma
Harvard University
qiurui_ma@g.harvard.edu

Michael P. Wellman
University of Michigan
wellman@umich.edu

ABSTRACT
In empirical game-theoretic analysis (EGTA), game models are extended iteratively through a process of generating new strategies based on experience with prior strategies. The strategy exploration problem in EGTA is how to direct this process so to construct effective models with minimal iteration. A variety of approaches have been proposed in the literature, including methods based on classic techniques and novel concepts. Comparing the performance of these alternatives can depend sensitively on criteria adopted and measures employed. We investigate some of the methodological considerations in evaluating strategy exploration, proposing and justifying new evaluation methods based on examples and experimental observations. In particular, we emphasize the fact that empirical games create a space of strategies and evaluation should reflect how well it covers the strategically relevant space. Based on this fact, we suggest that the minimum regret constrained profile (MRCP) provides a particularly robust basis for evaluating a space of strategies, and propose a local search method for computing MRCP. However, MRCP computation is not always feasible especially in large games. To evaluate strategy exploration in large games, we propose a new evaluation scheme that measures the strategic coverage of an empirical game. Specifically, we highlight consistency considerations for comparing across different approaches. We show that violation of the consistency considerations could yield misleading conclusions on the performance of different approaches. In accord with consistency considerations, we propose a profile-selection method, which effectively discovers the profile that can represent the strategic coverage of an empirical game through its regret information. We show that our evaluation scheme reveals the authentic learning performance of different approaches compared to previous evaluation methods.

ACM Reference Format:

1 INTRODUCTION
Recent years have witnessed dramatic advances in developing game-playing strategies through iterative application of (deep) reinforcement learning (RL). DeepMind’s breakthroughs in Go and other two-player strategy games [17, 18] demonstrated the power of learning through self-play. In self-play the learner generates an improved strategy assuming its opponent plays the current strategy. For many games, iterating best-response in this manner will cycle or otherwise fail to converge, which has led to consideration of alternative approaches to generate new strategies. For example, DeepMind’s milestone achievement in the complex video strategy game StarCraft II entailed an elaborate population-based search approach [22] informed by game-theoretic concepts.

Many recent works [1, 7, 10] have likewise appealed to game-theoretic methods to direct iterative RL for complex games. At each iteration, a new strategy is generated for one agent through RL, fixing other agents to play strategies from previous iterations. A general formulation of this approach is the Policy Space Response Oracle (PSRO) algorithm [7]. PSRO can be viewed as a form of empirical game-theoretic analysis (EGTA) [21, 25], a general name for the study of building and reasoning about game models based on simulation. In EGTA, game models are induced from simulations run over combinations of a particular set of strategies. The strategy exploration problem in EGTA [5] considers how to extend the considered strategy set, based on the current empirical game model. For example, one natural approach is to compute a Nash equilibrium (NE) of the current model, and generate a new strategy that optimizes payoff when other agents play that equilibrium. This approach of iteratively extending strategy sets by best-response to equilibrium was introduced by McMahan et al. [9] for two-player games and called the double oracle (DO) method.

PSRO defines an abstract operation on empirical games, termed meta-strategy solver (MSS), that extracts an opponent profile from the current empirical game as target for the next best-response calculation. In this framework, choosing an MSS determines the strategy exploration method. For example with NE-calculation as MSS in a two-player game, PSRO reduces to DO. An MSS that simply selects the most recently added strategy corresponds to self-play (SP). A variety of MSSs have been proposed and assessed in the literature on iterative RL-based approaches to games. We survey some of these ideas in §2, as well as alternative approaches to strategy exploration outside the PSRO framework (e.g., not involving RL or varying from best-response).

In practical terms, the proof of a method is whether it produces a superior solution (e.g., a champion Go program). However, we also seek to understand the relative effectiveness of strategy exploration methods across problem settings, and this remains an open problem for EGTA methodology. Comparing the performance of alternative methods is subtle because each generates a distinct sequence of strategies, and thus the empirical game model at any point reflects a distinct strategy space. The relevant comparisons are across different strategy spaces, which may not be faithfully represented by a simple summary such as an interim solution. This fact has tended to be neglected in studies proposing and evaluating new ideas on strategy.
exploration [7, 10], and as we demonstrate below, this can lead to misleading conclusions on the performance of different approaches.

The present study illuminates several methodological considerations for strategy exploration. First, we identify a key distinction between PSRO and other learning dynamics, which is that the empirical game model evolves through extending a space of strategies and hence its evaluation should reflect how well it covers the strategically relevant space.

Second, we seek a principled evaluation metric for empirical games, and suggest the proposal by Jordan et al. [5] that the regret of the minimum-regret constrained-profile (MRCP) can serve this purpose. We show the effectiveness and advantages of using MRCP as a metric through examples. To find MRCP more accurately, we propose a variant of the amoeba method [11] that outperforms previous approaches in matrix games. MRCP calculation is not always computationally feasible, so we identify some desiderata for alternative evaluation metrics. Per the first point above we highlight the importance of evaluating the whole space of strategies in an empirical game. Further, we propose some consistency considerations for comparing across different MSSs. We point out the MSS used for evaluation is not necessarily the same as the MSS in strategy exploration and define solver-based regret for evaluation purposes. Based on these considerations, we propose a new evaluation solver selection scheme for EGTA, which leads to a sensible comparison across MSSs. We demonstrate the significance of our considerations and approaches in both synthetic and real-world games.

Finally, we consider the problem of regret-based evaluation in situations where calculating exact best response is infeasible thus accurate regret is not available. One alternative that is widely applied is using generated strategies for evaluation purpose where regret calculation for different MSSs only considers deviations within the generated strategies. We test this approach from a game-theoretic perspective and find that high-regret profiles in the true game may exhibit low regret in the combined game, thus casting doubt on the accuracy of this approach.

Contributions of this study include:

1. Recognition that empirical games create a space of strategies and evaluation should reflect how well they cover the strategically relevant space. To serve this purpose, we suggest MRCP as evaluation metric and present evidence that MRCP provides a particularly robust basis for evaluation.
2. The notion of solver-based regret for evaluation, with focus on consistency considerations for comparing MSSs. We demonstrate the potential for misleading results when consistency is violated as in the prior literature.
3. A new evaluation solver selection scheme which leads to a sensible comparison across MSSs;
4. A variant of the amoeba method that outperforms previous approaches in matrix games, plus some insight on MRCP approximation in games wherein regret calculation is restricted.

2 RELATED WORK ON STRATEGY EXPLORATION

In the first instance of automated strategy generation in EGTA, Phelps et al. [14] employed genetic search over a parametric strategy space, optimizing performance against an equilibrium of the empirical game. Schwartzman & Wellman [16] combined RL with EGTA in an analogous manner. Questioning whether best response to equilibrium is an ideal way to add strategies, these same authors framed and investigated the general problem of strategy exploration in EGTA [15]. They identified situations where adding a best response to equilibrium would perform poorly, and proposed some alternative approaches. Jordan et al. [5] extended this line of work by proposing exploration of strategies that maximize the gain to deviating from a rational closure of the empirical game.

Investigation of strategy exploration was furthered significantly by introduction of the PSRO framework [7]. PSRO entails adding strategies that are best responses to some designated other-agent profile, where that profile is determined by the meta-strategy solver (MSS) applied to the current empirical game. The prior EGTA approaches cited above effectively employed NE as MSS as in the DO algorithm [9]. Lanctot et al. [7] argued that with DO the new strategy may overfit to the current equilibrium, and accordingly proposed and evaluated several alternative MSSs, demonstrating their advantages in particular games. For example, their projected replicator dynamics (PRD) employs an RD search for equilibrium [19, 20], but truncates the replicator updates to ensure a lower bound on probability of playing each pure strategy. Any solution concept for games could in principle be employed as MSS, as for example the adoption by Muller et al. [10] of a recently proposed evolutionary-based concept, a-rank [12], within the PSRO framework.

The MSS abstraction also connects strategy exploration to iterative game-solving methods in general, whether or not based on EGTA. Using a uniform distribution over current strategies as MSS essentially reproduces the classic fictitious play (FP) algorithm [3], and as noted above, an MSS that just selects the most recent strategy equates to self-play (SP). Note that these two MSS instances do not really make substantive use of the empirical game, as they derive from the strategy sets alone.

Wang et al. [24] illustrated the possibility of combining MSSs, employing a mixture of NE and uniform which essentially averages DO and FP. Motivated by the same aversion to overfitting the current equilibrium, Wright et al. [27] proposed an approach that starts with DO, but then fine-tunes the generated response by further training against a mix of previously encountered strategies.

In the literature, a profile’s fitness as solution candidate is measured by its regret in the true game. Jordan et al. [5] defined MRCP (minimum-regret constrained-profile) as the profile in the empirical game with minimal regret relative to the full game. Regret of the MRCP provides a measure of accuracy of an empirical game, but we may also wish to consider the coverage of a strategy set in terms of diversity. Balduzzi et al. [1] introduced the term Gamescape to refer to the scope of joint strategies covered by the exploration process to a given point. They employed this concept to characterize the effective diversity of an empirical game state, and proposed a new MSS called rectified Nash designed to increase diversity of the Gamescape. Finally, we take note of a couple of recent works that characterize Gamescapes in terms of topological features. Omidsfahiei et al. [13] proposed using spectral analysis of the a-rank best response graph, and Czarnecki et al. [4] visualize the strategic topography of real-world games as a spinning top wherein layers are transitive and strategies within a layer are cyclic.
3 PRELIMINARIES

A normal-form game $G = (N, (S_i), (u_i))$ consists of a finite set of players $N$ indexed by $i$; a non-empty set of strategies $S_i$ for player $i \in N$; and a utility function $u_i : \prod_{j \in N} S_j \to \mathbb{R}$ for player $i \in N$, where $\prod$ is the Cartesian product.

A mixed strategy $\sigma_i$ is a probability distribution over strategies in $S_i$, with $\sigma_i(s_j)$ denoting the probability player $i$ plays strategy $s_j$. We adopt conventional notation for the other-agent profile: $\sigma_i = \prod_{j \neq i} \sigma_j$. Let $\Delta(\cdot)$ represent the probability simplex over a set. The mixed strategy space for player $i$ is given by $\Delta(S_i)$. Similarly, $\Delta(S) = \prod_{i \in N} \Delta(S_i)$ is the mixed profile space.

Player $i$’s best response to profile $\sigma$ is any strategy yielding maximum payoff for $i$, holding the other players’ strategies constant:

$$br_i(\sigma_{-i}) = \arg\max_{\sigma_i \in \Delta(S_i)} u_i(\sigma_i, \sigma_{-i}).$$

Let $br(\sigma) = \prod_{i \in N} br_i(\sigma_{-i})$ be the overall best-response correspondence for a profile $\sigma$. A Nash equilibrium (NE) is a profile $\sigma^*$ such that $\sigma^* \in br(\sigma^*)$.

Player $i$’s regret in profile $\sigma$ in game $G$ is given by

$$\rho_i^G(\sigma) = \max_{s_j \in S_j} u_i(s_j, \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}).$$

Regret captures the maximum player $i$ can gain in expectation by unilaterally deviating from its mixed strategy in $\sigma$ to an alternative strategy in $S_i$. An NE strategy profile has zero regret for each player. A profile is said to be an $\epsilon$-Nash equilibrium ($\epsilon$-NE) if no player can gain more than $\epsilon$ by unilateral deviation. The regret of a strategy profile $\sigma$ is defined as the sum over player regrets:

$$\rho^G(\sigma) = \sum_{i \in N} \rho_i^G(\sigma).$$

Some treatments employ max instead of sum for this; when necessary to disambiguate we refer to (1) as sum-regret. Both are relevant measures of distance from equilibrium, and we appeal to the max-regret variant in our approach to approximating MRCP in §5.3.

An empirical game $G_{S|X}$ is an approximation of the true game $G$, in which players choose from restricted strategy sets $X_i \subseteq S_i$, and payoffs are estimated through simulation. That is, $G_{S|X} = (N, (X_i), (u_i))$, where $\hat{u}$ is a projection of $u$ onto the strategy space $X$.

4 EVALUATING STRATEGY EXPLORATION

The purpose of evaluating strategy exploration is to understand the relative effectiveness of different exploration methods (e.g., MSSs) across different problem settings. We achieve this purpose through analyzing the intermediate empirical game models they generate during exploration.

4.1 Evaluating an Empirical Game Model

From the perspective of strategy exploration, the key feature of an empirical game model is what strategies it incorporates. In EGTA, the restricted strategy set $X$ is typically a small slice of the set of all strategies $S$, so the question is how well $X$ covers the strategically relevant space. There may be several ways to interpret “strategically relevant”, but one natural criterion is whether the empirical game $G_{S|X}$ covers solutions or approximate solutions to the true game $G$.

The profile in the empirical game closest to being a solution of the full game is the MRCP, as described above. Formally, $\hat{\sigma}$ is an MRCP of $G_{S|X}$ iff:

$$\hat{\sigma} = \arg\min_{\sigma \in \Delta(\mathbb{X})} \sum_{i \in N} \rho_i^G(\sigma).$$

The regret of MRCP thus provides a natural measure of how well $X$ covers the strategically relevant space. In prior literature, MRCP was studied in games with fixed strategy sets rather than a setting where strategy sets are iteratively built. We extend the study of its properties to our strategy exploration setting. We first note that the regret of MRCP decreases monotonically as the empirical game model is being extended, since adding strategies can only increase the scope of minimization. Moreover, MRCP tracks convergence in that the regret of MRCP reaches zero exactly when an NE of $G_{S|X}$ is contained in the empirical game, that is, $X$ covers the support of the NE. We claim both properties of MRCP are important and desirable for evaluation purposes.

Unfortunately, direct use of MRCP as a means for evaluating strategy exploration can be computationally challenging. Calculating regret of a profile, the quantity we are minimizing, generally requires a best-response oracle for the full game, which itself can be quite computationally expensive (which is why we often find RL the best available method). And even given an effective way to calculate regret, the search for MRCP is a non-convex optimization problem over the profile space of the empirical game.

4.2 Solver-Based Regret

Given the general difficulty of computing MRCP, studies often employ some other method to select a profile from the empirical game to evaluate. Any such method can be viewed as a meta-strategy solver, and so we use the term solver-based regret to denote regret in the true game of a strategy profile selected by an MSS from the empirical game. In symbols, the solver-based regret using a particular MSS is given by $\rho^G(MSS(G_{S|X}))$. By definition, MRCP is the MSS that minimizes solver-based regret.

An MSS that is commonly employed for solver-based regret is NE. NE-based regret measures the stability in the true game of a profile that is perfectly stable in the empirical game. Whereas any MSS is eligible to play the role of solver, not all are well-suited for evaluating strategy exploration. For example, SP simply selects the last strategy added, and is completely oblivious to the rest of the strategy set $X$. This clearly fails to measure how well $X$ as a
whole captures the strategically relevant part of $S$, which is the main requirement of an evaluation measure as described above.

### 4.3 Solver Consistency for Evaluation

Our framework as described to this point employs MSSs in two distinct ways: to direct a strategy exploration process, and to evaluate intermediate results in strategy exploration. It may seem natural to evaluate exploration that employs MSS $M$ in terms of solver-based regret with $M$ as solver. Indeed, much prior work in PSRO exploration has done exactly this [7, 8, 10].

As we demonstrate below, however, evaluating alternative MSSs $M$ and $M'$ for exploration using their respective MSSs as solvers can produce misleading comparisons, caused by neglecting the principle of evaluating the empirical game as a whole. Instead, we argue, one should apply the same solver-based regret measure to evaluate results under $M$ and $M'$. In other words, the MSS employed in solver-based regret should be fixed and independent of the MSSs employed for exploration. We term this the consistency criterion.

To illustrate the necessity of solver consistency, we offer two examples to demonstrate how a violation of our consistency criterion could lead to a misleading conclusion.

**Example 1.** Consider the symmetric zero-sum matrix game of Table 1. Starting from the first strategy of each player, we perform PSRO with uniform and NE as MSSs, respectively. The first few iterations of PSRO are presented in Table 2. Due to symmetry, the two players’ strategy sets and MSS-proposed strategies are identical.

<table>
<thead>
<tr>
<th>Iter#</th>
<th>Strategy Sets</th>
<th>DO proposed strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$a_1^1$, $a_2^1$</td>
<td>(1), (1)</td>
</tr>
<tr>
<td>2</td>
<td>$a_1^2$, $a_2^2$, $a_3^2$</td>
<td>(0, 1), (0, 1)</td>
</tr>
<tr>
<td>3</td>
<td>$a_1^3$, $a_2^3$, $a_3^3$</td>
<td>(0, 0), (0, 0)</td>
</tr>
</tbody>
</table>

**Table 1** A symmetric zero-sum game (Example 1).

Fig. 1a presents regret curves for both MSSs using NE-based regret, as well as the uniform-based regret curve for FP. If we violate the consistency criterion and compare uniform-based regret of FP with the NE-based regret of DO (i.e., green versus blue curves in Fig. 1a), we would conclude FP converges faster than DO in the first two iterations. However, FP cannot actually be better at strategy exploration, as the strategies introduced, $a_1^1$ and $a_2^3$, are identical under two MSSs. Moreover, at the third iteration, FP fails to add any new strategy, and so the improvement shown is not attributable to the exploration process.

Comparing the two MSSs under NE-based regret (i.e., green versus orange regret curves), we see that where FP and DO generate identical empirical games their evaluations coincide. Thus, following

$^3$Although Li and Wellman [8] is not focused on strategy exploration, it does present some plots (Figs. 2 and 3) with multiple curves using different MSSs for evaluating regret. For other works, we verified this by examining the published code and through our own efforts to reproduce the results in these papers. Specifically, we found the code published as part of OpenSpiel [6] evaluates progress in exploration by regret of the MSS employed for exploration. We also reproduced the learning performance of PSRO with different MSSs and inferred that the MSS used for evaluation is the same as the one for strategy exploration, which is often apparent by examination of regret curves. For example, the NE-based regret curve of fictitious play oscillates dramatically while its uniform-based regret curve is much more smooth. So it is easy to identify which MSS was used for evaluation.

the rule of consistency avoids reaching a misleading conclusion about exploration. Note that we would reach the same conclusion if the two MSSs are evaluated under uniform-based regret (i.e., red versus blue curves). However, we observe that not all MSSs are equally effective for evaluation. In this example, although uniform-based regret consistently evaluates equivalent empirical games, its low weight on newly added strategies fails to adequately reflect exploration achievements. For example, the uniform-based regret curve remains well above zero even after the full-game NE has been covered in the empirical game. In Section 4.4, we provide a detailed discussion of this phenomenon and propose a scheme for evaluation solver selection.

Of course, if the goal is just to evaluate DO and FP as online algorithms, then the green versus blue comparison is appropriate. A key virtue of the PSRO framework, however, is that it highlights exploration as a distinct issue and provides the MSS abstraction for addressing it. Within an iterative EGTA approach, the choice of solver to employ for decision making at any stage is completely orthogonal to the method used to extend the game model, and so focusing attention on algorithms that couple these in particular ways (e.g., using the same MSS for solving and exploration) is unnecessarily limiting.

**Example 2.** We further verify our observations in a synthetic zero-sum game with 100 strategies per player. Resulting regret curves averaged over 10 random starts are shown in Fig. 1b. As for the previous example, comparing uniform-based regret of FP against NE-based regret of DO—breaking our consistency criterion—would lead us astray. First, we see that FP performs best initially, but is ultimately overtaken by DO. More importantly, as we demonstrate in §5.1 below, even the assessment that FP’s strategy exploration is more effective than DO’s over the first thirty iterations is invalid. Indeed, the blue-versus-green comparison up to iteration 30 shows that the uniform strategy profile in the empirical game of FP is more stable (has lower regret) than NE in the empirical game of DO. But as in the prior example, this is an artifact of selecting the uniform rather than the NE profile for evaluation. Moreover, as illustrated below in Fig. 3, we should generally expect there to exist non-NE profiles in the empirical game of DO with significantly lower regret in the true game.
This example demonstrates mixed use of evaluation metrics may result in improper comparison among the performance of MSSs. Indeed, we have found that this phenomenon is quite common in prior work, leading in particular to misleading evaluations of FP as a strategy exploration approach. In formulating the general consistency criterion, we emphasize that improper comparisons could be made with any two MSSs; the issue is not limited to FP or any specific MSSs employed in these examples.

### 4.4 Evaluation Solver Selection

We further examine the consistency criterion in simplified poker games, specifically two-player Kuhn poker and Leduc poker. These poker games have been commonly employed in prior work within the PSRO framework, facilitating comparison of experimental results. Specifically, we evaluate FP, PRD, and NE as MSSs. Moreover, to select an effective solver to implement the consistency criterion, we propose a new evaluation solver selection scheme, designed to reveal the authentic performance of MSSs for strategy exploration.

#### 4.4.1 Solver Consistency with FP

For Leduc poker (Fig. 2a), DO performs better than FP under NE-based regret. However, the uniform-based regret is quite misleading as a measure of exploration performance of DO. It actually increases over much of the range, which would seem to suggest that adding strategies makes the game model worse, which intuitively makes little sense.

In Kuhn poker (Fig. 2b), DO again outperforms FP under NE-based regret. Uniform-based regret of DO is misleading for Kuhn as it is for Leduc poker. FP shows much faster convergence under NE-based rather than uniform-based regret after twenty iterations or so. Indeed, the uniform-based regret is far from zero even at a hundred iterations. As we saw in the examples above, uniform-based evaluation may misleadingly show smooth improvement where there is none. Here we see again that it can also leave the impression of slow progress even when the empirical game actually contains the key strategies needed for accurate solution.

#### 4.4.2 Solver Consistency with PRD

We show experimental results of PSRO with PRD in Leduc poker in Fig. 2c. We first note that following the rule of consistency, there is little performance gap between PRD and DO (i.e., the blue and orange curves). If we violate consistency and compare PRD-based regret of PRD against NE-based regret of DO (green versus blue curves), however, we would be prone to conclude that PRD clearly and significantly outperforms DO. For Kuhn poker (Fig. 2d) we would conclude there is little difference, but looking closely and ignoring consistency might lead us to conclude that PRD is slightly worse in the limit. In both cases, we see that the choice of evaluation solvers can drive assessments about exploration performance.

#### 4.4.3 An Evaluation Solver Selection Scheme

Recall that MRCP is the MSS minimizing solver-based regret and thus the regret of the MRCP of an empirical game measures how well the empirical game covers the strategically relevant space. If we could feasibly compute the MRCP or an approximation, that would be a natural choice for solver-based regret. Though this is infeasible in general, we can capture the spirit of MRCP by attempting to minimize solver-based regret. Toward this end, we propose a heuristic evaluation solver selection scheme that chooses the solver with lowest-regret curve among running solvers. We demonstrate the significance of our scheme for evaluating different MSSs by checking the previous PRD example.

In the example, if we merely adhere to solver consistency with NE-based regret (i.e., comparing blue versus orange regret curves in Fig. 2c), we would not distinguish the performance difference between PRD and DO. In this case, NE in the empirical game exhibits
Figure 2 Experimental regret curves for poker games.

relatively high regret with respect to the true game. We know it is far from MRCP, as the green curve in this plot demonstrates the existence of lower-regret profiles in the same empirical games. Although we cannot tell exactly where the MRCP lies, the PRD solver in this example clearly provides a better approximation than does the NE solver. Considering PRD as the solver for evaluation and following solver consistency, we can likewise evaluate DO using PRD-based regret. The result is shown in the purple curve of Fig. 2e (other regret curves are as in Fig. 2c). PRD-based regret of DO is indeed lower than NE-based regret of DO (purple versus blue curves), and thus PRD as an evaluation solver successfully identifies the profiles with lower regret in the empirical games across DO iterations. This achieves our purpose of identifying profiles closer to MRCP as the basis for evaluation.

By comparing the PRD-based regret curves of DO and PRD, we observe that they exhibit similar improvement rates through early iterations, but eventually PRD shows a small consistent advantage. This we regard as the best available evidence from these experiments on the authentic relationship between PRD and DO. Had we ignored solver consistency and compared the green and blue curves, we would have correctly concluded PRD’s superiority but grossly overestimated the performance gap.

To state our proposal more explicitly: we argue for selecting the solver that minimizes regret in the given context. Specifically, fix
When applying the amoeba method to the MRCP optimization problem, which are high-dimensional vectors, may not be reached given fixed scaling parameters. We fix this problem instead by projecting an infeasible point onto the unit simplex to maintain feasibility. Our algorithm is specified in detail in Appendix A.

5 PERFORMANCE OF MRCP AND CALCULATION REFINEMENT

5.1 Evaluation Performance of MRCP

Though computation of MRCP in large games is generally infeasible, for experimental purposes we can evaluate it in a feasible context. Here we present such an evaluation on matrix games of fixed and modest size. Fig. 3 displays averaged regret curves of PSRO runs on the same synthetic matrix game of Example 2, with FP and DO evaluated by MRCP-based regret. We observe that the MRCP-based regret by definition is lower than its NE-based regret counterpart. In this instance, the comparison using MRCP-based regret validates the qualitative comparison using NE-based regret. Notice that the gap between NE-based regret and MRCP-based regret diminishes as DO and FP gradually converge to a true game NE (i.e., all regrets approach zero). We also observe that the MRCP-based regret curves are much smoother than the NE-based regret curves. MRCP is monotone by definition, the steady performance improvement reflects more accurately the progress in quality of empirical game model achieved by strategy exploration.

5.2 MRCP Calculation Refinement

In matrix games, MRCP can be approximated by solving an optimization problem, for example, using the amoeba method [11]. When applying the amoeba method to the MRCP optimization problem, we have to reconcile the fact that the optimization problem is constrained while the amoeba method is an unconstrained optimization technique. To handle this issue, Jordan et al. [5] propose a binary search (BS) to select the maximum feasible reflection and expansion scaling parameters (step sizes), respectively. However, this approach handles infeasibility by compromising the quality of the reflected and expanded points since the optimal solution points, which are high-dimensional vectors, may not be reached given fixed scaling parameters. We fix this problem instead by projecting an infeasible point onto the unit simplex to maintain feasibility. Our algorithm is specified in detail in Appendix A.

5.3 MRCP Approximation in Large Games

Calculating MRCP in large games can be infeasible since it demands a large number of regret queries each entailing an expensive best-response calculation. We therefore seek an affordable way to approximate MRCP in large games. We start by deriving an upper bound for the regret of a mixed-strategy profile through the deviation payoff of a finite set of pure-strategy profiles. We then approximate MRCP by minimizing the upper regret bound. This approach allows us to focus on pure-strategy deviations which is a more manageable space compared to the search over mixed-strategy profiles.

We derive the upper regret bound as follows:

\[
ρ^G_1(σ) = \max_{s_i \in S_i} u_i(s_i') - u_i(σ_i, σ_{-i})
\]

\[
= \max_{s_i' \in S_i} \sum_{s_{-i} \in S_{-i}} σ(s_{-i})u_i(s_i', s_{-i}) - \sum_{s_i \in S_i} \sum_{s_{-i} \in S_{-i}} σ(s_i)σ(s_{-i})u_i(s_i, s_{-i})
\]

\[
\leq \sum_{s_{-i} \in S_{-i}} σ(s_{-i}) \max_{s_i' \in S_i} u_i(s_i', s_{-i}) - \sum_{s_i \in S_i} \sum_{s_{-i} \in S_{-i}} σ(s_i)σ(s_{-i})u_i(s_i, s_{-i}).
\]

Note that the utility structure of a game may affect the quality of our regret bound. For example, in two-player zero-sum games, since the sum of players’ utilities is zero for every profile, the term \(\sum_{s_i \in S_i} \sum_{s_{-i} \in S_{-i}} σ(s_i)σ(s_{-i})u_i(s_i, s_{-i})\) (i.e., expected utility of playing σ for player i) is canceled when we sum the regret bound over players. As a result, minimizing the summation of upper bounds always produces a pure strategy profile, which could result in a large estimation error.

We handle this issue by replacing sum-regret (1) with the maximal regret over players. Our approximate MRCP \(\tilde{σ}\) employs the max-regret variant:

\[
\tilde{σ}^X = \arg\max_{σ \in A(X)} \max_{i \in \mathbb{N}} ρ^G_1(σ)
\]

This modification prevents the expected utility term from being canceled, leading to a more effective result from minimizing the regret bound.

To verify that using max-regret does not unduly distort results, we can evaluate the sum-regret of the profile produced by minimizing
As noted above, calculating profile regret for purposes of evaluating MSSs generally requires identifying a best-response strategy. However, computing the exact best response may not be feasible in complex games. A particular approach is to collect the strategies generated across a set of PSRO runs, and evaluate regret with respect to that set. This idea that evaluation is conducted based on generated strategies is general and has been widely applied especially when the true game is extremely large, e.g., the attack-graph game [27]. We refer to the game with all generated strategies as the combined game. In general, regret with respect to the combined game is a lower bound on regret with respect to the true full game. This idea that evaluation is conducted based on generated strategies generally requires identifying a best-response strategy. However, computing the exact best response may not be feasible in complex games. A particular approach is to collect the strategies generated across a set of PSRO runs, and evaluate regret with respect to that set. This idea that evaluation is conducted based on generated strategies is general and has been widely applied especially when the true game is extremely large, e.g., the attack-graph game [27]. We refer to the game with all generated strategies as the combined game. In general, regret with respect to the combined game is a lower bound on regret with respect to the true full game. Since the combined game has been used in practice as a heuristic approach to evaluate strategy exploration, it is important to examine its effectiveness.

To test the effectiveness of this approach, we compare results for evaluation with respect to a combined game with that of exact best response (i.e., the ground truth in our context), for some games where calculating exact best responses is feasible. Results are shown in Fig. 2. We observe that high-regret profiles in the true game may exhibit quite low regret in the combined game. Most concerning is that the slack in the regret bound may vary across MSSs being evaluated, thus producing misleading comparisons. Specifically in Fig. 2f, despite the apparent higher regret of FP profiles in the true game, FP profiles exhibit lower regret in the combined game. Our explanation for the phenomenon is that when one MSS can explore certain strategy to which strategies generated by other MSSs can deviate largely but not vice versa, the combined game fails to identify the correct ordering of MSSs. Details of our combined-game analysis are provided in Appendix B.

### 6 EVALUATION WITHOUT EXACT BEST RESPONSE

<table>
<thead>
<tr>
<th>Index</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\rho(\bar{\sigma}))</td>
<td>359</td>
<td>262</td>
<td>232</td>
<td>428</td>
<td>305</td>
<td>176</td>
<td>124</td>
<td>487</td>
<td>364</td>
<td>75</td>
<td>95</td>
<td>228</td>
<td>627</td>
<td>103</td>
<td>322</td>
</tr>
<tr>
<td>(\rho(\bar{\sigma}))</td>
<td>505</td>
<td>275</td>
<td>265</td>
<td>532</td>
<td>353</td>
<td>253</td>
<td>144</td>
<td>727</td>
<td>365</td>
<td>106</td>
<td>575</td>
<td>397</td>
<td>794</td>
<td>183</td>
<td>322</td>
</tr>
<tr>
<td>(\rho(\bar{\sigma}^*))</td>
<td>615</td>
<td>275</td>
<td>242</td>
<td>554</td>
<td>949</td>
<td>535</td>
<td>144</td>
<td>806</td>
<td>737</td>
<td>377</td>
<td>491</td>
<td>514</td>
<td>973</td>
<td>172</td>
<td>507</td>
</tr>
</tbody>
</table>

**Table 3** MRCP quality with approximation in symmetric-zero sum games.

either version. Let \(\bar{\sigma}^X\) be the profile minimizing sum-regret with respect to strategy set \(X\), and \(\delta^X\) the corresponding MRCP using max-regret (4). Note that for any \(X\), \(\rho(\delta^X) \leq \rho(\bar{\sigma}^X)\). Table 5 in Appendix compares the two MRCP definitions in five instances of Kuhn poker, for each of three sizes of two-player Kuhn poker. As we see, the MRCP calculated using max-regret is quite close to the actual sum-regret MRCP in minimizing sum-regret.

We now measure the quality of our approximation using the upper regret bound (3) with the max-regret version of MRCP (4). Our experiment employs a synthetic two-player zero-sum game with 200 strategies and utilities uniformly sampled from \([-R, R]\), \(R = 1000\). Table 3 compares the regrets of exact MRCP \(\bar{\sigma}\), approximated MRCP \(\bar{\sigma}\) (we overload the notation for convenience), and NE \(\sigma^*\) (i.e., a benchmark). We observe in some sampled empirical games, the approximation gives profiles with very similar regret as that of the MRCP. However, the approximation quality is not consistent across all sampled empirical games. One explanation is that when the true MRCP is close to a pure strategy profile, the approximation gives good results while the performance of approximation declines when the randomness in MRCP is high. Future study on this thread could include tightening the bound to get better approximation of MRCP.

### 7 CONCLUSION

The primary contributions of this study are methodological considerations for evaluating strategy exploration in EGTA, within the PSRO framework. Our observations address nuances that have not been observed before, and may have led to misleading conclusions about the effectiveness of proposed methods. In particular, we propose an evaluation scheme with a consistency condition, dictating that progress in strategy exploration under different MSSs be evaluated with respect to the same solver. This condition, while seemingly obvious, has not always been followed, perhaps because it is natural in online learning settings to evaluate a method at any point based on its own solution criterion. In the context of strategy exploration, in contrast, what is important is not what the latest strategy is, but how it affects the solution of the model it is being added to.

We further investigate the significance of MRCP for strategy exploration, presenting evidence that the MRCP provides a principled basis for robust evaluation. We propose an algorithm to calculate MRCP that outperforms previous methods.

Finally, we examine the effectiveness of the combined game for evaluating strategy exploration when calculating the exact best response is computationally infeasible. Based on our results, we cast doubt on the accuracy of this evaluation approach.
REFERENCES


Appendices

A ACCURACY IMPROVEMENT ON MRCP CALCULATION

We show the procedure of projected amoeba method in Algorithm 1. Our goal is to minimize the cumulative regret function \( f(\sigma) = \sum_{i \in N} P_i^G(\sigma) \) shown in Equation (4.1). Denote the projection operator as \( P(\sigma_i) = \arg\min_{\sigma'_i \in \Delta(S_i)} \| \sigma'_i - \sigma_i \| \) for player \( i \in N \). Denote by \( P(\sigma) \) the projection operator for each \( \sigma_i \in \sigma \). For amoeba, we follow the default values of \( \alpha = 1, \gamma = 2, \rho = 1/2, \sigma = 1/2 \).

**Algorithm 1 Projected Amoeba Method**

**Input:** A full game model with regret function \( f \) and an empirical game model.

**Parameter:** Amoeba method parameters \( \alpha, \gamma, \rho, \sigma \) corresponding to the reflection, expansion, contraction and shrink coefficients.

**Output:** MRCP \( \sigma \).

1: while \( t = 1, \ldots, T \) do
2: Select current test profiles \( \sigma^1, \ldots, \sigma^{n+1} \).
3: Order according to the regrets at these profiles: \( f(\sigma^1) \leq \ldots \leq f(\sigma^{n+1}) \).
4: Calculate \( \sigma^\prime \), the centroid of profiles except \( \sigma^{n+1} \).
5: Reflection: Compute reflected point \( \sigma^\prime = \sigma^0 + \alpha(\sigma^0 - \sigma^{n+1}) \).
6: Project \( \sigma^\prime \) to probability simplex \( \sigma^0 = P(\sigma^\prime) \).
7: if \( f(\sigma^1) \leq f(\sigma^0) \) then
8: \( \sigma^{n+1} \leftarrow \sigma^\prime \) and Continue.
9: else
10: Continue.
11: end if
12: Expansion:
13: if \( f(\sigma^0) < f(\sigma^1) \) then
14: \( \sigma^0 = \sigma^0 + \alpha(\sigma^0 - \sigma^\prime) \).
15: \( \sigma^0 = P(\sigma^0) \).
16: if \( f(\sigma^0) < f(\sigma^\prime) \) then
17: \( \sigma^{n+1} \leftarrow \sigma^0 \) and Continue.
18: else
19: \( \sigma^{n+1} \leftarrow \sigma^\prime \) and Continue.
20: end if
21: end if
22: Contraction: \( \sigma^0 = \sigma^0 + \alpha(\sigma^{n+1} - \sigma^0) \).
23: \( \sigma^0 = P(\sigma^0) \).
24: if \( f(\sigma^0) < f(\sigma^\prime) \) then
25: \( \sigma^{n+1} \leftarrow \sigma^0 \) and Continue.
26: end if
27: Shrink: \( \sigma^0 = \sigma^0 + \alpha(\sigma^{n+1} - \sigma^0) \).
28: \( \sigma^0 = P(\sigma^0) \) and Continue.
29: end while
30: return \( \sigma \).

We compare the performance of amoeba method with two approaches—BS and projection—in two-player Kuhn poker. Empirical games with different sizes are first sampled from the true game and then MRCP is approximated with different approaches. Table 4 shows the regret of MRCP calculated by different approaches. To illustrate the regret gap between different MRCP calculations, we also provide the regret of NE of the empirical game as a benchmark. We observe that for each size of a empirical game, calculating MRCP with projection results in a profile with significantly lower regret, merely with a different infeasibility handling approach. We also notice that when the size is small, the performance of two approaches is close. However, as the size increases, the BS approach does not lead to a good MRCP approximation and even NE of the empirical game could have lower regret than the MRCP approximation.

Moreover, to understand the stability of results given by different approaches, we calculate the regret of approximated MRCP of a fixed empirical game for multiple times. We find that the variance of regrets given by BS approach from multiple runs is very large while the variance of our method is tiny. The improvement in the accuracy of MRCP approximation could dramatically benefit the study of evaluating strategy exploration since MRCP serves as a theoretical evaluation metric.

B THE COMBINED GAME EXAMPLE

The exact best response oracle provides a regret calculation tool for evaluating MSSs. Despite its availability in the poker games that we experiment with, in many complex games, calculation for the exact best response is not feasible due to the large game trees. An alternative that have been employed in many prior works is applying the concept of the combined game as a heuristic evaluation approach. In this section, we verify its effectiveness from a game-theoretic perspective.

A combined game is an encompassing empirical game whose strategy set is a union over all strategy sets of the empirical games that are generated across different MSSs or random seeds. The missing payoff entries are further simulated. Evaluation is conducted by viewing the combined game as the approximation of the true game and regret calculation only considers deviations within the combined game. The general idea of the combined game—using generated strategies for evaluation purposes—is widely applied and can be viewed as an instance of agent vs agent by Baldizzi et al. [2].

We test this approach in games where exact best responses are available and thus the ground truth performance of different MSSs is known. We consider three MSS PRD, NE and uniform in two-player Leduc poker. For each MSS, we perform 3 differently seeded PSRO runs with 150 strategies, thus each player has 1350 strategies in the combined game. We compare results of evaluation conducted with the combined game and the exact best response. In Fig. 2f, each curve is an average of three runs with different seeds. The three averaged NE-based regret curves on the top are generated with the exact best response oracle, evaluating with respect to the true game, while the bottom ones are created by only considering deviations within the combined game. The stratification is caused by the fact that the regret of a profile in the true game is lower-bounded by the empirical game.

We observe that the regret curves based on the combined game does not truthfully reflect the order of performance with exact best response. In Fig. 2f, despite the apparent higher regret of FP profiles in the true game, FP profiles exhibit low regret in the combined game. The disparity might stem from the strategy generation process of FP: new strategies are trained against a uniform opponent strategy, which makes it harder to be exploited by the counterpart of DO and
Table 4 MRCP quality with different infeasibility handling methods.

<table>
<thead>
<tr>
<th>Index</th>
<th>Size = 5</th>
<th>Size = 7</th>
<th>Size = 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho(\vec{\sigma}) ) w. BS</td>
<td>0.39</td>
<td>0.36</td>
<td>0.35</td>
</tr>
<tr>
<td>( \rho(\vec{\sigma}) ) w. Proj</td>
<td>0.39</td>
<td>0.30</td>
<td>0.30</td>
</tr>
<tr>
<td>( \rho(\sigma^*) )</td>
<td>0.50</td>
<td>0.39</td>
<td>0.78</td>
</tr>
</tbody>
</table>

Table 5 MRCP quality with two definitions.

<table>
<thead>
<tr>
<th>Index</th>
<th>Size = 5</th>
<th>Size = 10</th>
<th>Size = 15</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho(\vec{\sigma}) ) w. BS</td>
<td>0.46</td>
<td>0.49</td>
<td>0.45</td>
</tr>
<tr>
<td>( \rho(\vec{\sigma}) ) w. Proj</td>
<td>0.07</td>
<td>0.35</td>
<td>0.17</td>
</tr>
<tr>
<td>( \rho(\sigma^*) )</td>
<td>0.29</td>
<td>0.50</td>
<td>0.26</td>
</tr>
</tbody>
</table>

C CHOICE OF SOLUTION CONCEPTS IN EMPIRICAL GAME

In EGTA, an empirical game is used to approximate the true game and NE of the empirical game is viewed as an approximate NE to the true game. In some scenarios, e.g., games with large strategy space, limited number of strategies are available to be considered in the empirical game due to constraints on computational resources or the difficulty of hand-crafting strategies. When the true game may not be well-approximated, we pose a following question: is NE of the empirical game truly the best solution concept to adopt?

Through our experiments, we notice that the empirical NE could have quite large regret in the true game compared with other profiles in the empirical game. Even worse, it is possible that the empirical NE possesses the largest regret among all possible profiles in some extreme cases. Consider matrix game Table 6 with empirical game \((a_1^2, a_2^2)\) for both players. The NE \((a_1^2, a_2^2)\) is the most unstable pure strategy profile in the empirical game. Table 7 list the regret of each pure strategy profile.

<table>
<thead>
<tr>
<th>( a_1^2 )</th>
<th>( a_2^2 )</th>
<th>( a_2^3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( (0, 0) )</td>
<td>( (-1, 1) )</td>
<td>( (-2, 2) )</td>
</tr>
<tr>
<td>( (1, -1) )</td>
<td>( (0, 0) )</td>
<td>( (-5, 5) )</td>
</tr>
<tr>
<td>( (2, -2) )</td>
<td>( (5, -5) )</td>
<td>( (0, 0) )</td>
</tr>
</tbody>
</table>

Table 6 Handcrafted Symmetric Zero-Sum Game.

The example inspires us to reconsider the solution concept to apply to real world games: NE strategies of the empirical game
Table 7 Regrets of Pure Strategy Profiles.

<table>
<thead>
<tr>
<th>Profiles</th>
<th>regret</th>
</tr>
</thead>
<tbody>
<tr>
<td>((a_1^1, a_2^1))</td>
<td>4</td>
</tr>
<tr>
<td>((a_1^1, a_2^2))</td>
<td>7</td>
</tr>
<tr>
<td>((a_1^2, a_2^2))</td>
<td>7</td>
</tr>
<tr>
<td>((a_1^2, a_2^2))</td>
<td>10</td>
</tr>
</tbody>
</table>

could be far from optimum in a global perspective and regularization to NE may be required to obtain a stable performance.

D EXPERIMENTAL PARAMETERS

D.1 Synthetic Games

In the experiment shown in Fig. 1b, we have two-player zero-sum matrix games with 100 strategies per player with uniformly sampled integer payoffs from \(-10\) to \(10\), both inclusive.

In the experiment shown in Fig. 3, we have two-player zero-sum matrix games with 200 strategies per player with uniformly sampled integer payoffs from \(-10\) to \(10\), both inclusive.

In the MRCP approximation experiment, we have two-player zero-sum matrix games with 200 strategies per player with uniformly sampled integer payoffs from \(-1000\) to \(1000\), both inclusive. The range of payoffs is enlarged to prevent same payoffs assigned to large number of actions.

D.2 Poker Games

We use OpenSpiel [6] default parameter sets for experiments on Leduc and Kuhn poker: each payoff entry in an empirical game is an average of 1000 repeated simulations; DQN is adopted as a best response oracle, its parameters are shown in Table 8. The poker games are asymmetric in the sense that one player always moves first.

PRD is implemented with lower bound for strategy probability \(1e-10\), maximum number of steps \(1e5\) and step size \(1e-3\).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>learning rate</td>
<td>(1e-2)</td>
</tr>
<tr>
<td>Batch Size</td>
<td>32</td>
</tr>
<tr>
<td>Replay Buffer Size</td>
<td>(1e4)</td>
</tr>
<tr>
<td>Episodes</td>
<td>(1e4)</td>
</tr>
<tr>
<td>optimizer</td>
<td>adam</td>
</tr>
<tr>
<td>layer size</td>
<td>256</td>
</tr>
<tr>
<td>number of layer</td>
<td>4</td>
</tr>
<tr>
<td>Epsilon Start</td>
<td>1</td>
</tr>
<tr>
<td>Epsilon End</td>
<td>0.1</td>
</tr>
<tr>
<td>Exploration Decay Duration</td>
<td>(3e6)</td>
</tr>
<tr>
<td>discount factor</td>
<td>0.999</td>
</tr>
<tr>
<td>network update step</td>
<td>10</td>
</tr>
<tr>
<td>target network update steps</td>
<td>500</td>
</tr>
</tbody>
</table>

Table 8 DQN parameter