Learning Not to Regret

David Sychrovský [1,2] Michal Šustr [2,5] Elnaz Davoodi [3]
Michael Bowling [4] Marc Lanctot [3] Martin Schmid [1,5]

[1]Charles University [2]Czech Technical University [3]Google DeepMind

[4]University of Alberta [5]EquiLibre Technologies

Summary

We accelerate Nash-equilibrium approximation on a distribution of games by meta-learning regret minimizers, often by an order of magnitude.

Abstract

The literature on game-theoretic equilibrium finding predominantly focuses on single games or their repeated play. Nevertheless, numerous real-world scenarios feature playing a game sampled from a distribution of similar, but not identical games, such as playing poker with different public cards or trading correlated assets on the stock market. As these similar games feature similar equilibra, we investigate a way to **accelerate equilibrium finding** on such a distribution. We present a novel "learning not to regret" framework, enabling us to meta-learn a regret minimizer tailored to a specific distribution. Our key contribution, **Neural Predictive Regret Matching (NPRM)**, is uniquely meta-learned to converge rapidly for the chosen distribution of games, while having regret minimization guarantees on any game. We validated our algorithms' faster convergence on a distribution of river poker games. Our experiments show that the meta-learned algorithms outpace their non-meta-learned counterparts, achieving more than tenfold improvements.

Meta-Learning Framework

On a distribution of regret minimization tasks G, we aim to find an online algorithm m_θ with some parameterization θ that efficiently minimizes the expected external regret after T steps. We thus want, given some observed rewards $\{\boldsymbol{x}^\tau\}_{\tau=1}^{t-1}$, to minimize the following loss

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{g\sim G}\left[R^{\text{ext},T}\right] = \mathop{\mathbb{E}}_{g\sim G}\left[\max_{a\in A}\sum_{t=1}^{T}\boldsymbol{r}_a\left(\boldsymbol{\sigma}_\theta^t,\boldsymbol{x}^t\right)\right], \tag{1}$$

where $\boldsymbol{\sigma}_\theta^t$ is the strategy selected at step t by the online algorithm m_θ, and $\boldsymbol{r}(\sigma,\boldsymbol{x}) = \boldsymbol{x} - \langle\boldsymbol{\sigma},\boldsymbol{x}\rangle\mathbf{1}$ is the instantaneous regret. We train a recurrent neural network θ to minimize (1). By utilizing a recurrent architecture we can also represent algorithms that are history and/or time dependent.

The choice to minimize external regret in particular is arbitrary. This is because the rewards $\{\boldsymbol{x}^\tau\}_{\tau=1}^{T}$ that come from the environment are constant w.r.t. θ and the derivative of any element of the cumulative regret vector $\boldsymbol{R}^T = \sum_{t=1}^{T}\boldsymbol{r}^t$ is thus the same, meaning

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\sigma}_\theta^t} = \frac{\partial}{\partial\boldsymbol{\sigma}_\theta^t}\mathop{\mathbb{E}}_{g\sim G}\left[\sum_{\tau=1}^{T}-\langle\boldsymbol{\sigma}_\theta^\tau,\boldsymbol{x}^\tau\rangle\right] = -\mathop{\mathbb{E}}_{g\sim G}\left[\boldsymbol{x}^t\right].$$

Consequently, if the objective (1) is reformulated using other kinds of regrets, it would result in the same meta-learning algorithm. This is because regrets measure the difference between reward accumulated by some fixed strategy, and by the algorithm m_θ. Since the former is a constant at meta-train time, minimizing (1) is equivalent to maximizing the reward $\langle\boldsymbol{\sigma}^t,\boldsymbol{x}^t\rangle$ the algorithm m_θ gets at every $t \le T$ in a task $g \sim G$, given the previous rewards $\{\boldsymbol{x}^\tau\}_{\tau=1}^{t-1}$, similar to policy gradient.

Algorithms and Their Computational Graphs

Neural online algorithm (NOA) directly outputs strategy $\boldsymbol{\sigma}^t$, and is not guaranteed to minimize regret. *Neural predictive regret matching* (NPRM) uses the predictive regret framework [1] to meta-learn the prediction. NPRM combines the strongest regret minimization guarantees with policy gradient.

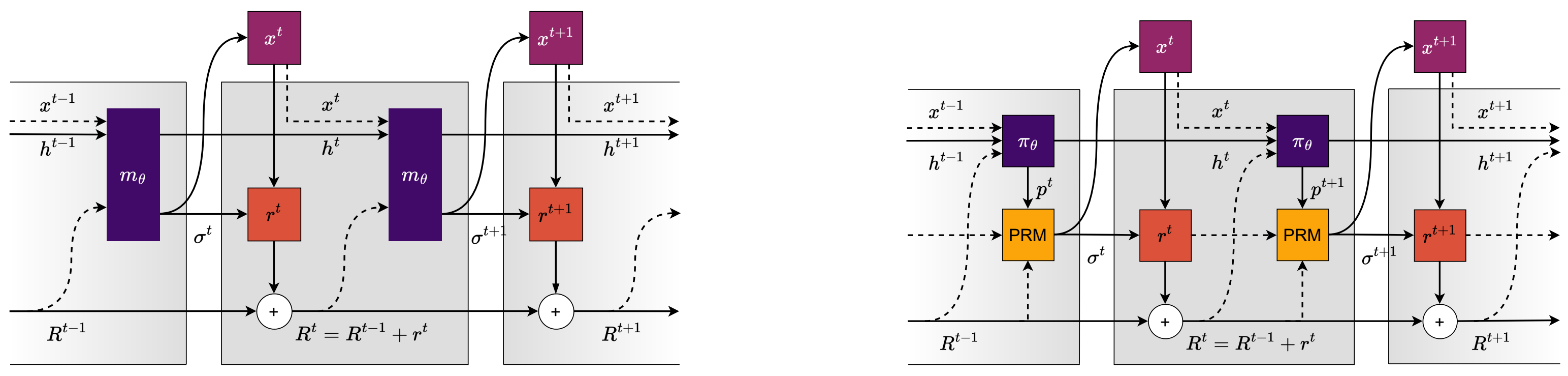

Figure 1. Neural online algorithm (NOA; left), and Neural predictive regret matching (NPRM; right). The gradient flows only along the solid edges. The $\boldsymbol{h}$ denotes the hidden state of the neural network.

Empirical Evaluation

We evaluate on `rock_paper_scissors`, a matrix game where one matrix element is randomized. Furthermore, `river_poker` is a distribution of river endgames of Texas Hold'em Poker with $\approx 40k$ information states. The public cards and the beliefs are drawn at random.

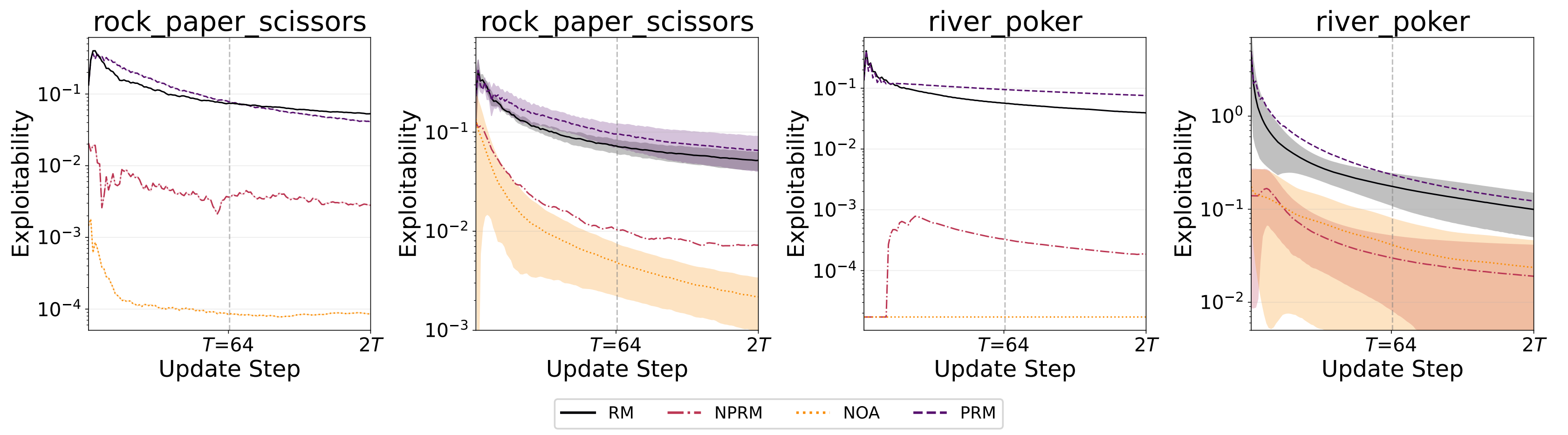

Figure 2. Comparison of non-meta-learned algorithms (RM, PRM) with meta-learned algorithms (NOA, NPRM), on a small matrix game and a large sequential game and for a single fixed game versus a whole distribution over games. The figures show exploitability of the average strategy $\overline{\boldsymbol{\sigma}}$. The y-axis uses a logarithmic scale. Vertical dashed lines separate two regimes: training (up to T steps) and generalization (from T to $2T$ steps). Colored areas show standard errors.

Relative Speedup

We tracked how many steps it takes to reach a solution of specified target quality. When used within continual resolving framework, one typically sets a target solution quality to reach at each iteration. Both NOA and NPRM outperform (P)RM for all target exploitabilities, with better solutions requiring an order of magnitude less steps.

Target	$4\cdot10^{-1}$	10^{-1}	$6\cdot10^{-2}$	$2\cdot10^{-2}$
RM	20	128	212	615
PRM	36	158	261	793
NOA	**1**	18	41	157
NPRM	**1**	**16**	**26**	**118**

Table 1. Number of steps each algorithm requires to reach target exploitability on `river_poker`(sampled).

Convergence in Policy Space

To further illustrate the differences between the meta-learned algorithms and (P)RM, we plot the current and average strategies selected by each algorithm on `rock_paper_scissors`(sampled). Both NOA and NPRM are initially close to the equilibrium and converge relatively smoothly. In contrast, (P)RM visit large portion of the policy space even in later steps, making the convergence slower.

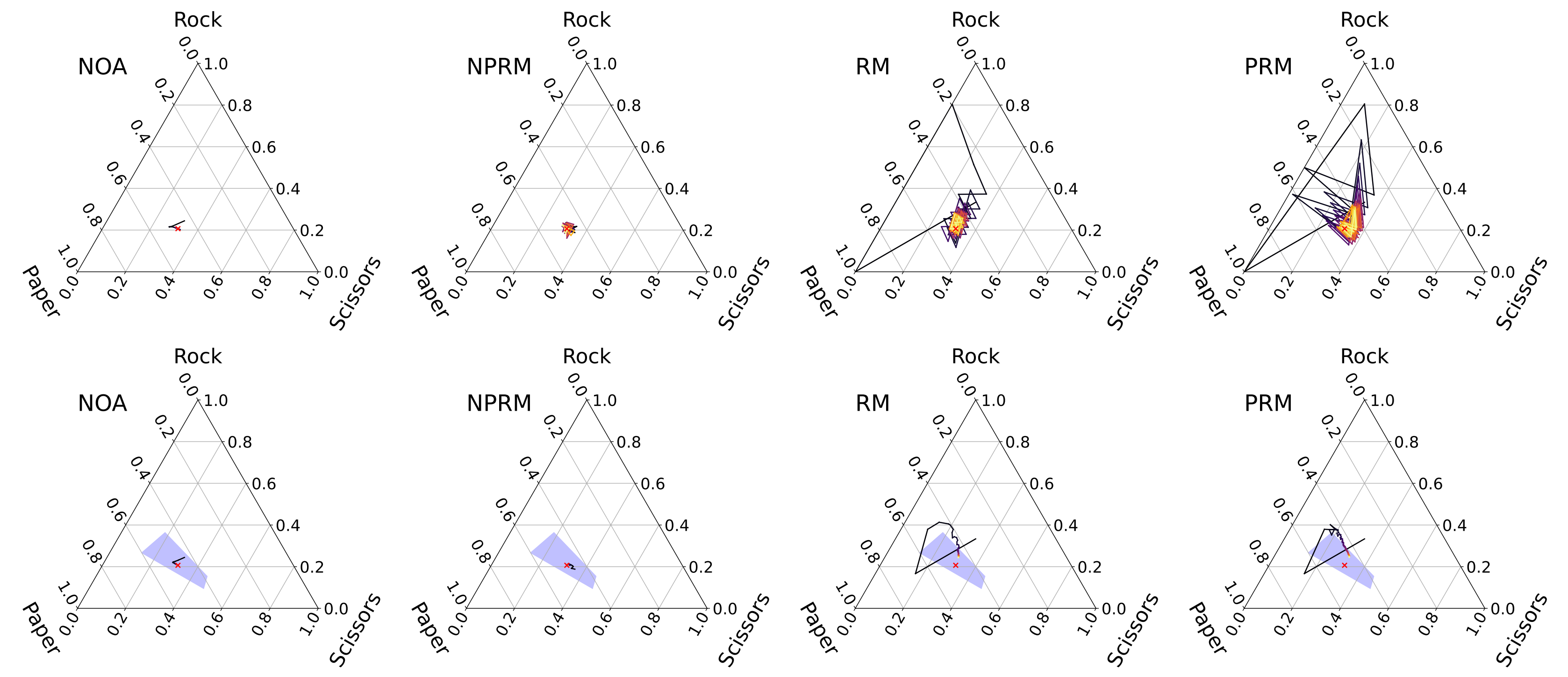

Figure 3. For each algorithm, we show the trajectories of current strategies (top row) and average strategies (bottom row) on `rock_paper_scissors` (sampled) for $2T = 128$ steps. The red cross shows the equilibrium of the sampled game. The trajectories start in dark colors and get brighter for later steps. The blue polygon is the set of all equilibria in the distribution `rock_paper_scissors` (sampled). Notice how the strategies of our meta-learned algorithms begin in the polygon and refine their strategy to reach the current equilibrium. In contrast, (P)RM are initialized with the uniform strategy and visit a large portion of the policy space.

Computational Requirements

Using the neural network in our algorithms incurs an additional computational overhead. We present evaluation as a function of time, rather than the number of steps. Our algorithms can outperform their non-meta-learned counterparts, even when accounting to this extra cost. The difference is greater in `river_poker`, since each interaction with the environment is more expensive.

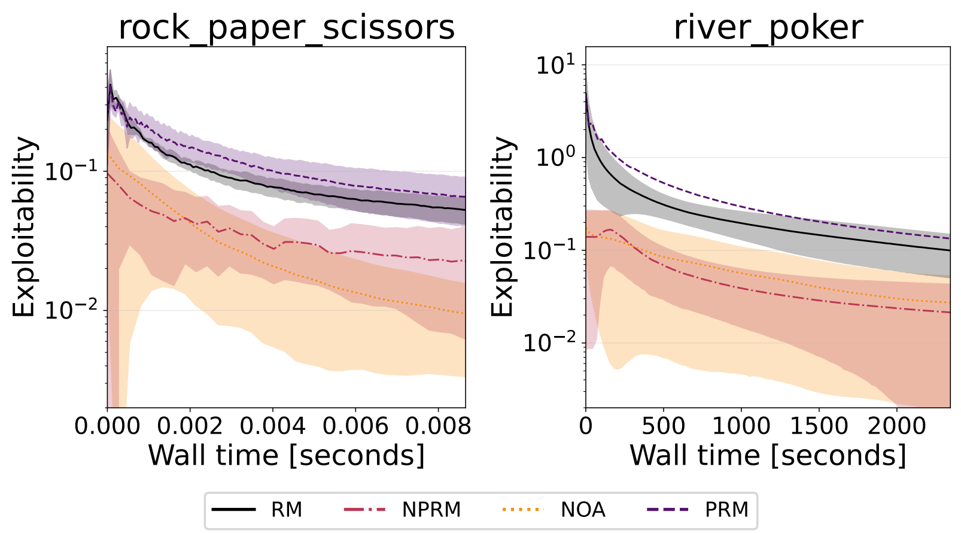

Figure 4. Comparison of regret minimization algorithms as a function of wall time, rather than number of steps.

Out of Distribution Convergence

The performance can deteriorate when the meta-learned algorithms are deployed on a distribution they were not meta-learned on. However, in contrast to NOA, NPRM is guaranteed to minimize regret on an arbitrary task.

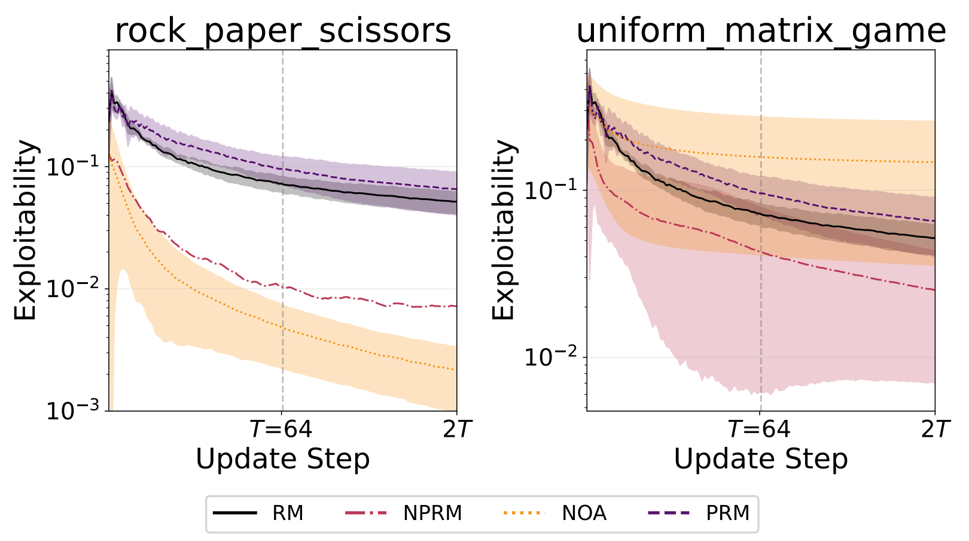

Figure 5. Comparison of the converge guarantees of NOA and NPRM. Both were trained on `rock_paper_scissors` (sampled). Left figure shows NOA and NPRM can out outperform (P)RM on the distribution it was trained on. However, right figure shows that when evaluated on `uniform_matrix_game` (sampled), the performance of NOA deteriorates significantly.

Conclusion

We introduced two new meta-learning algorithms for regret minimization in a new *learning not to regret* framework. Our algorithms are meta-learned to minimize regret fast against a distribution of potentially adversary environments. We evaluated our methods in games, where we minimize regret against an (approximate) value function and measure the exploitability of the resulting strategy. Our experiments show that our meta-learned algorithms attain low exploitability approximately an order of magnitude faster than prior regret minimization algorithms.

In the future, we plan to extend our results to the self-play settings. We also plan to apply our methods with hindsight rationality [3] for games which change over time. This is also an opportunity to combine our offline meta-learning with the online meta-learning of [2].

References

[1] Gabriele Farina, Christian Kroer, and Tuomas Sandholm.
Faster game solving via predictive blackwell approachability: Connecting regret matching and mirror descent.
In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 6, pages 5363–5371, 2021.

[2] Keegan Harris, Ioannis Anagnostides, Gabriele Farina, Mikhail Khodak, Zhiwei Steven Wu, and Tuomas Sandholm.
Meta-learning in games.
arXiv preprint arXiv:2209.14110, 2022.

[3] Dustin Morrill, Ryan D'Orazio, Reca Sarfati, Marc Lanctot, James R Wright, Amy R Greenwald, and Michael Bowling.
Hindsight and sequential rationality of correlated play.
In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 6, pages 5584–5594, 2021.

Acknowledgements

The authors would like to thank Martin Loebl, Matej Moravčík, Viliam Lisý, and Milan Hladík for their insightful comments. This work was supported by the Czech Science Foundation grant no. GA22-26655S and CoSP Project grant no. 823748. Computational resources were supplied by the project "e-Infrastruktura CZ" (e-INFRA LM2018140) provided within the program Projects of Large Research, Development and Innovations Infrastructures.

sychrovsky@kam.mff.cuni.cz