Leading the Pack: N-player Opponent Shaping

Alexandra Souly[1] **Timon Willi**[2] **Akbir Khan**[1] **Robert Kirk**[1]
Chris Lu[2] **Edward Grefenstette**[1] **Tim Rocktäschel**[1]
[1]University College London [2]University of Oxford
alexandrasouly@gmail.com

Abstract

Reinforcement learning solutions have great success in the 2-player general sum setting. In this setting, the paradigm of Opponent Shaping (OS), in which agents account for the learning of their co-players, has led to agents which are able to avoid collectively bad outcomes, whilst also maximizing their reward. These methods have currently been limited to 2-player game. However, the real world involves interactions with many more agents, with interactions on both local and global scales. In this paper, we extend Opponent Shaping (OS) methods to environments involving multiple co-players and multiple shaping agents. We evaluate on over 4 different environments, varying the number of players from 3 to 5, and demonstrate that model-based OS methods converge to equilibrium with better global welfare than naive learning. However, we find that when playing with a large number of co-players, OS methods' relative performance reduces, suggesting that in the limit OS methods may not perform well. Finally, we explore scenarios where more than one OS method is present, noticing that within games requiring a majority of cooperating agents, OS methods converge to outcomes with poor global welfare.

1 Introduction

Multi agent systems are increasingly deployed to the real world, in which interactions may be either cooperative, competitive or both. These can be considered general-sum games where interactions combine opportunities for coordination with self-interested motivations, where each player is trying to maximise their own reward. Real-world examples of such games include global pollution, deforestation, over-fishing and arms races. In these settings, agents are incentivised to exploit the system and/or other players for their own gains by defecting, but cooperation would result in greater social welfare.

General sum games have multiple equilibria, many of which do not produce high social welfare. Naive learning algorithms that work well in zero-sum settings do not perform well in general-sum settings: they learn the Nash-equilibrium strategies by best-responding to the opponent's past behaviour, but in these settings many Nash-equilibria often coincides with worst-case outcomes for all participants. [Letcher et al., 2021]

Opponent shaping (OS) methods are a family of approaches that take into account the evolving social dynamics instead of simply best-responding to the opponent's behaviour [Foerster et al., 2018b, Lu et al., 2022, Zhao et al., 2022, Willi et al., 2022, Khan et al., 2023]. They act to influence other participant's learning dynamics to result in a favorable equilibrium. These methods have been successfully used in 2-player general sum settings to achieve high rewards, however they do not always achieve pro-social outcomes, and can sometimes successfully exploit their opponent.

The study of OS methods has so far been limited to only two interacting agents, whilst the real world has many more agents. The interactions of more players give rise to more complicated and unstable social dynamics, making cooperation harder to achieve. In particular, in diverse populations

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

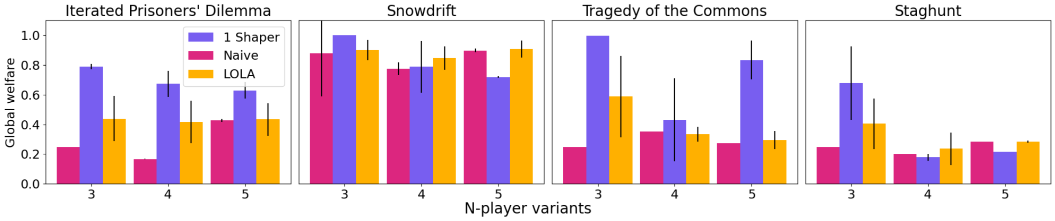

Figure 1: **Normalised global welfare with a single shaping agent.** We report the converged (normalised) global welfare for training a shaping agent with unseen co-players across games and player variants. Across most games SHAPER outperforms LOLA and as number of players increase, both OS methods fail to find better welfare maximising equilibria than the naive baseline. Results are reported over 5 seeds with mean and standard deviation.

equilibrium selection requires modelling not only a single co-player's beliefs but also their beliefs of other co-players.

In this work, we introduce a natural extension for OS methods to more-than-2-player setting. We then evaluate this extension over a total of 12 varying games. We find that OS methods consistently help achieve higher global welfare than baseline methods. While OS-methods are only maximising their own reward, they do so while effectively avoiding collective welfare minimising outcomes, creating better social dynamics and thus also improving their co-players' rewards. We also find that between model-based and model-free OS methods, model-free methods perform better in games with more players. In some instances where OS methods are unable to exploit their opponents, they are also willing to take lower rewards in order to avoid a collectively bad outcome. This is promising for scenarios that involve some more capable self-interested agents in a population of less capable agents.

2 Background

Model-based OS methods such as LOLA Foerster et al. [2018b], COLA Willi et al. [2022], POLA Zhao et al. [2022], SOS Letcher et al. [2021] and Meta-MAPG Kim et al. [2021] assume access to the other agents parameters, and as such, anticipate their co-player's updates as they learn. For example, LOLA conditions its update on a co-player's next set of parameters (effectively thinking one step ahead). COLA, POLA and SOS are further adaptations and extensions of LOLA, addressing different shortcomings of the original method. Meta-MAPG also differentiates through the opponent's current parameters via an additional peer learning objective.

In comparison, model-free OS methods [Lu et al., 2022, Khan et al., 2023, Fung et al., 2023, Lu et al., 2023] frame opponent-shaping as a meta-reinforcement learning problem, where the goal is to maximise reward over a series of games against a co-player that is learning. In these approaches a shaping agent trains over T *trials* consisting of E *episodes*. Each trial consists of the following: the shaping agent plays the game with the co-player for E episodes, with the co-player updating after every episode according to its update rule. After every episode, the shaping agent updates its hidden state to add the played games into its memory, whilst the opponent takes a full gradient learning step. After a trial the shaping agent updates its policy parameters. Importantly, the shaping agent does not update during an episode unlike the co-player.

3 Methods

We expand SHAPER Khan et al. [2023] to a multi-player setting and generalise the algorithm to arbitrary numbers of shaping-agents and co-players. Co-players can be any type of other agents, including SHAPER, other meta-agents or naive learners.

We extend opponent-shaping to a generic number of shaping agents and co-players naturally. Similar to single SHAPER single co-player shaping, shaping agents play the game with their co-players for E episodes but with all co-players simultaneously updating after an episode. After every episode,

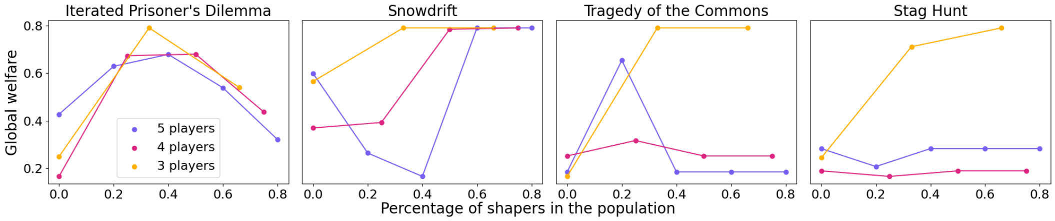

Figure 2: **Normalised global welfare with multiple shaping agents.** In IPD, the optimal amount of SHAPERs is around a third of the population, too many decreases global welfare. In comparison in Snowdrift - a game where only a single agent is required to cooperate, more shaping agents help coordinate which naive learner is exploited. In ToC and Stag Hunt, where cooperation requires a majority of players, increasing the number of SHAPERs did not increase welfare.

all shaping agents simultaneously update their respective hidden states. Shaping agents learn to collectively shape their naive co-players.

We update SHAPERs using evolutionary strategies [Salimans et al., 2017], which are computationally better suited to the long timescale of trials and are efficiently parallelisable. Given a game with N SHAPERs and K co-players, at the start of a trial we sample a population of groups of SHAPERs. These groups then each play against copies of the same co-players. At the end of the trial, we sample a new generation of groups based on the parameters of the most successful groups. Importantly, there is no mixture between the groups. SHAPERs do not play with members of other groups, only with the SHAPERs in their own group, therefore the whole group evolves together. We present a multiplayer extension in Appendix B.

4 Experiments

For all games, one episode consists of 100 iterated cooperate-defect choices. One trial consists of 1000 episodes. We evaluate on the following general sum games:

Iterated Prisoner's Dilemma (IPD) is a game in which agents must commit to cooperating (staying silent) or defecting (implicating your comrades) [Yao and Darwen, 2001]. For each player, assuming that the actions of others players is fixed, defecting always results in a better payoff than cooperating (that is to say, defection is the dominant choice). All players choosing cooperation is global welfare maximising, but unstable with naive learners.

Snowdrift is a game in which players choose to shovel (cooperate) or ignore the snow (defect) [Wettergren, 2021]. The shovelling players share a fixed cost. If a player doesn't shovel, they don't incur a cost but still receive the benefits of the snow is shovelled. If no one shovels, all players receive a low payoff. This game differs from IPD in that mutual defection is not the best choice - if no one else cooperates, it is still better for the player to shovel alone than if everyone defects and the road stays blocked.

Tragedy of the Commons (ToC) is an N-player social dilemma. We use a general formulation in which there is a social benefit B such that if enough players pay a cost C, everyone gets the benefit. The social benefit can be thought of as the shared resource, and the cost as the opportunity loss of not exploiting the resource. This game encourages free-riding, as long as enough of the co-players cooperate to prevent the tragedy.

Stag Hunt N players choose either to hunt a stag (cooperate) or to hunt a hare (defect) [Pacheco et al., 2008]. A certain number of hunters are required to successfully hunt a stag (in our case more than half the total population), the hunt is unsuccessful otherwise. If there is a successful hunt, the payoff is related to the number of cooperating hunters (more hunters implies bigger game). The stag hunters pay the cost of the hunt but everyone equally shares the benefits if the hunt is successful. This setup also encourages free-riding similarly to ToC.

Methods - We consider three different methods for shaping: naive learners, a model-based OS method (LOLA) and the model-free OS method SHAPER. Naive learners (NLs) are agents that don't

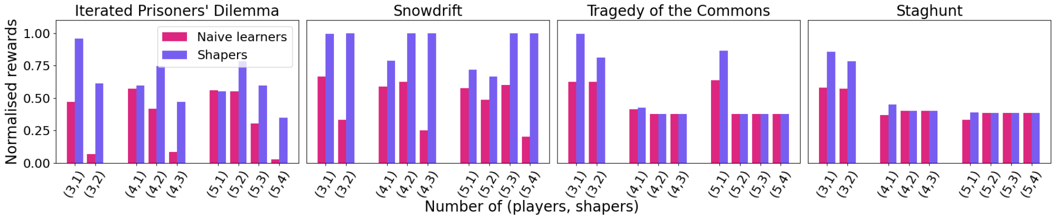

Figure 3: **Normalised SHAPER versus naive co-player welfare.** We report the (normalised) converged scores attained by SHAPERs and their naive co-players. In all games, SHAPERs either score similarly to their naive co-players, or heavily exploit them, particularly in IPD and Snowdrift.

account for the learning or adaptation of other agents in the environment. Updates are myopic, at the end of single episodes, and make no modelling assumptions of other players. Our naive learners are parameterized by a recurrent neural network and keep their hidden states between episodes. Their networks are updated using proximal policy optimisation (PPO) Schulman et al. [2017] and trained until convergence. For our model-free OS method we use LOLA-PG - a policy gradient approximation to estimate the co-player's gradient update. We use the Infinitely Differentiable Monte Carlo Estimator [Foerster et al., 2018a, DICE] to approximate the gradients, and train until convergence. For our model free OS method we use SHAPER, a meta-agent parameterised by a LSTM network, specifically a GRU with hidden state of size 16. Models are trained until convergence.

5 Results, Discussion & Conclusion

Single Shaping Agent We firstly conduct experiments in which we evaluate a single SHAPER with $N-1$ naive learners across the many environments. In line with prior work [Lu et al., 2022] we find that SHAPER outperforms LOLA across most tasks in improving global welfare (see Figure 1). We also find that current OS methods provide diminishing results when applied on games with more than 2 players. In all 3-player settings, SHAPER exploits its co-players, but has varying success with more co-players (Figure 3). In particular, SHAPER has difficulties on Staghunt and ToC, in scenarios where a majority of players is required for cooperation to be rewarded.

We note that in ToC the difficulty of cooperation is not monotonic in number of players. In the 4-player game, 3 are required to coordinate to get welfare maximising outcome, whilst in the 5-player variant the required number is also 3. Hence, with randomly searching agents it is easier in the 5-player variant to discover welfare maximising outcomes than the 4-player one. (50% vs 31%).

Many Shaping Agents With many shaping agents, we explore the stability of OS methods when interacting with one another. In games where cooperation is related to only a small percentage of a population cooperating, more SHAPERs improves global welfare (see Figure 2). In Snowdrift, only a single agent must cooperate, therefore increasing the number of SHAPERs increases performance as exploiting the naive learners becomes easier. Conversely, in IPD, where incentives for cooperation are lower, we find multiple SHAPERs reduce global welfare when the majority of the population become SHAPERs, due to overly exploiting the remaining naive learners (Figure 3). In games such as ToC and Staghunt, cooperation is only beneficial if enough players coordinate. Two SHAPERs successfully coordinate in the 3 player game (where only one player can defect), with one SHAPER cooperating and the other defecting. However they do not succeed at coordination when more players or SHAPERs are present, suggesting a limitation to the paradigm.

In conclusion, while OS methods improve global welfare over naive learning in multi-agent games, they might do so by exploitation of their co-players. When the game dynamics do not allow for this, they are willing to take lower rewards by cooperating. This is promising for improving the dynamics of finite-resource games like ToC by including more capable agents in the population. The method, however, faces limitations as the number of agents and SHAPERs increases, especially in games requiring majority cooperation. Future work could extend this to the STORM environments [Willi et al., 2023, Rutherford et al., 2023] with partial observability and temporally-extended action spaces.

References

Jakob Foerster, Gregory Farquhar, Maruan Al-Shedivat, Tim Rocktäschel, Eric P. Xing, and Shimon Whiteson. Dice: The infinitely differentiable monte-carlo estimator, 2018a.

Jakob N. Foerster, Richard Y. Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness, 2018b.

Kitty Fung, Qizhen Zhang, Chris Lu, Timon Willi, and Jakob Nicolaus Foerster. Analyzing the sample complexity of model-free opponent shaping. In *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*, 2023. URL `https://openreview.net/forum?id=Dm2fbPpU6v`.

Akbir Khan, Newton Kwan, Timon Willi, Chris Lu, Andrea Tacchetti, and Jakob Nicolaus Foerster. Context and history aware other-shaping, 2023. URL `https://openreview.net/forum?id=54F8woU8vhq`.

Dong Ki Kim, Miao Liu, Matthew D Riemer, Chuangchuang Sun, Marwa Abdulhai, Golnaz Habibi, Sebastian Lopez-Cot, Gerald Tesauro, and Jonathan How. A policy gradient algorithm for learning to learn in multiagent reinforcement learning. In *International Conference on Machine Learning*, pages 5541–5550. PMLR, 2021.

Steven Kuhn. Prisoner's Dilemma. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2019 edition, 2019.

Alistair Letcher, Jakob Foerster, David Balduzzi, Tim Rocktäschel, and Shimon Whiteson. Stable opponent shaping in differentiable games, 2021.

Chris Lu, Timon Willi, Christian Schroeder de Witt, and Jakob Foerster. Model-free opponent shaping, 2022.

Chris Lu, Timon Willi, Alistair Letcher, and Jakob Nicolaus Foerster. Adversarial cheap talk. In *International Conference on Machine Learning*, pages 22917–22941. PMLR, 2023.

Jorge Pacheco, Francisco Santos, and Max Souza. Evolutionary dynamics of collective action in n-person stag hunt dilemmas. *Proceedings. Biological sciences / The Royal Society*, 276:315–21, 09 2008. doi: 10.1098/rspb.2008.1126.

Alexander Rutherford, Benjamin Ellis, Matteo Gallici, Jonathan Cook, Andrei Lupu, Gardar Ingvarsson, Timon Willi, Akbir Khan, Christian Schroeder de Witt, Alexandra Souly, Saptarashmi Bandyopadhyay, Mikayel Samvelyan, Minqi Jiang, Robert Tjarko Lange, Shimon Whiteson, Bruno Lacerda, Nick Hawes, Tim Rocktaschel, Chris Lu, and Jakob Nicolaus Foerster. Jaxmarl: Multi-agent rl environments in jax, 2023.

Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning, 2017.

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

Thomas A. Wettergren. Replicator dynamics of an n-player snowdrift game with delayed payoffs. *Applied Mathematics and Computation*, 404:126204, 2021. ISSN 0096-3003. doi: https://doi.org/10.1016/j.amc.2021.126204. URL `https://www.sciencedirect.com/science/article/pii/S0096300321002940`.

Timon Willi, Alistair Letcher, Johannes Treutlein, and Jakob Foerster. Cola: Consistent learning with opponent-learning awareness, 2022.

Timon Willi, Akbir Khan, Newton Kwan, Mikayel Samvelyan, Chris Lu, and Jakob Foerster. Pax: Multi-agent learning in jax. `https://github.com/ucl-dark/pax`, 2023.

Xin Yao and Paul Darwen. An experimental study of n-person iterated prisoner's dilemma games. *Informatica*, 18, 07 2001. doi: 10.1007/3-540-60154-6_50.

Stephen Zhao, Chris Lu, Roger Baker Grosse, and Jakob Nicolaus Foerster. Proximal learning with opponent-learning awareness, 2022.

A Additional Environment Details

For each game, we present payoff matrices for their work.

IPD Global welfare increases linearly with the number of cooperators, but agents are incentivised to defect for individually higher payoffs. Therefore this game presents a free-riding problem. The payoff matrix we use is structured as in Table 1: defection results in 1 more point than cooperation given the actions of the co-players, and for each co-player cooperating our reward increases by 2.

		No. of C among remaining n-1 players				
		0	1	2	...	$n-1$
Player A	**C**	0	2	4	...	$2(n-1)$
	D	1	3	5	...	$2n-1$

Table 1: N-player IPD payoff where n is the total number of players, C is cooperation action and D is defection action.

StagHunt

The payoffs are calculated with the following equations, where N is the population size, and the reward is more than the individual cost of the hunt:

$$\begin{aligned}
&\text{if} \quad \text{num coop} \geq \lceil \frac{N}{2} \rceil : \\
&\qquad \text{C payoff} = \frac{\text{num coop} * \text{reward}}{N} - \text{hunt cost} \\
&\qquad \text{D payoff} = \frac{\text{num coop} * \text{reward}}{N} \\
&\quad \text{otherwise:} \\
&\qquad \text{C payoff} = -\text{hunt cost} \\
&\qquad \text{D payoff} = 0
\end{aligned}$$

In our experiments we set the cost of the hunt to 3 and the reward for a successful hunt to 6.

Tragedy of the Commons

The payoff matrix we use is presented in Table 2, as per Kuhn [2019]. In our experiments, we set the benefit to 5 and the cost to 3.

	more than T choose C	**T or fewer choose C**
C	benefit - cost	-cost
D	benefit	0

Table 2: Tragedy of the Commons payoff where T is the threshold for resource exhaustion, C is cooperation action and D is defection action

B Multi-Shaping Method

Following algorithm:

Algorithm 1 Training multiple SHAPER agents $\{S\}$ against co-players $\{-S\}$
.

```
Require: Stochastic game $\mathcal{M}$
Initialise all SHAPER parameters $\phi_i$ for $i \in \{S\}$
for $t = 0$ to $T$ do
    Initialise trial reward $\bar{J} = 0$
    Initialise co-players $\phi_i$ for $i \in \{-S\}$
    Initialise all agent hidden state $h_i = \mathbf{0}$ for $i \in \{S\} \bigcup \{-S\}$
    for $e = 0$ to $E$ do
        Roll out trajectories $J_i, h'_i = \mathcal{M}(\phi_{\{S\}}, \phi_{-\{S\}}, h_{\{S\}}, h_{-\{S\}})$ with $i \in \{S\} \bigcup \{-S\}$
        Update co-player's parameters $\phi_{-i}$ with $i \in \{-S\}$
        Update all hidden states $h_i \leftarrow h'_i$ with $i \in \{S\} \bigcup \{-S\}$
        Update trial reward $\bar{J} \leftarrow \bar{J} + J_i$
    end for
    Update SHAPER parameters $\phi_i$ with respect to $\bar{J}$ with $i \in \{S\}$
end for
```

C Hyperparameters

Hyperparameter	Value
Number of Actor Hidden Layers	1
Number of Critic Hidden Layers	1
Torso GRU Size	[16]
Length of Trial	1000
Length of Episode	100
Number of Generations	until convergence
Number of parallel environments	2
Number of parallel opponents	10
Population Size	100
OpenES sigma init	0.04
OpenES sigma decay	0.999
OpenES sigma limit	0.01
OpenES init min	0.0
OpenES init max	0.0
OpenES clip min	-1e10
OpenES clip max	1e10
OpenES lrate init	0.01
OpenES lrate decay	0.9999
OpenES lrate limit	0.001
OpenES beta 1	0.99
OpenES beta 2	0.999
OpenES eps	1e-8

Table 3: Hyperparameters for SHAPER

Hyperparameter	Value
Number of Minibatches	10
Number of Epochs	4
Gamma	0.96
GAE Lambda	0.95
PPO clipping epsilon	0.2
Value Coefficient	0.5
Clip Value	True
Max Gradient Norm	0.5
Anneal Entropy	False
Entropy Coefficient Start	0.1
Entropy Coefficient Horizon	400000000
Entropy Coefficient End	0.01
LR Scheduling	False
Learning Rate	0.0003
ADAM Epsilon	1e-5
With CNN	False

Table 4: Hyperparameters for PPO