NeuPL: Neural Population Learning

read paper

Siqi Liu[1, 2], Luke Marris[1, 2], Daniel Hennes[2], Josh Merel[2], Nicolas Heess[2], Thore Graepel[1, 2]

University College London, UK[1]
DeepMind[2]

ICLR 2022

Learning in Games

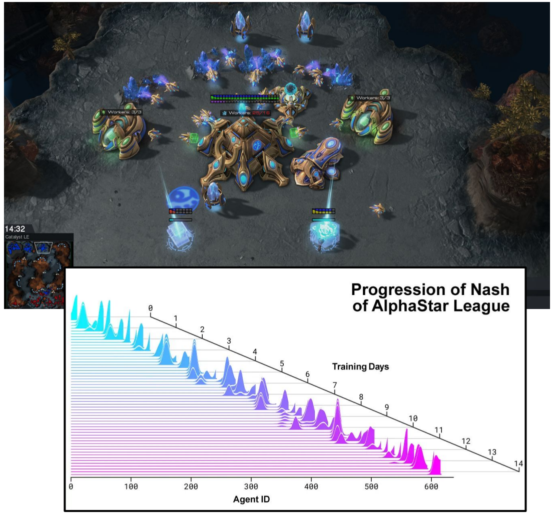
Transitive Games: self-play is enough

Cyclic Games: game-theoretic reasoning

UCL

[1] Balduzzi, David, et al. "Open-ended learning in symmetric zero-sum games." *International Conference on Machine Learning*. PMLR, 2019.
[2] Czarnecki, Wojciech M., et al. "Real World Games Look Like Spinning Tops." *Advances in Neural Information Processing Systems* 33 (2020).

Learning in “Real-World” Games



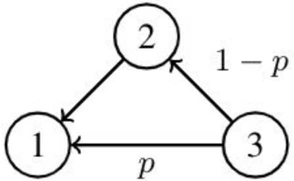
PSRO[1]: iterative learning of best-responses to mixture opponent strategies given by the meta-strategy solver (MSS) given expected returns between strategies.

- MSS = NE/Unif: convergence to NE.

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ p & 1-p & 0 \end{bmatrix}$$

$\sigma_2 = \Sigma_{[2, :]} =$ SOLVE-NE(Payoffs[:2, :2])

$$J_i = \mathbb{E}_{j \sim \Pr(\sigma_i)} \left[\mathbb{E}_{a \sim \pi_i, a' \sim \pi_j} \left[\sum_t r_t \gamma^t \right] \right]$$



PSRO-NASH

Limitations:

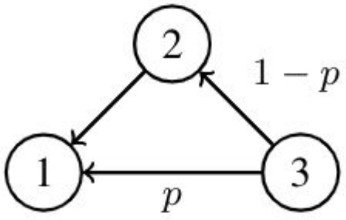
- Independent iterative learning of policies;
- “Good-”responses due to early stopping.

[1] Lanctot, Marc, et al. "A unified game-theoretic approach to multiagent reinforcement learning." Advances in neural information processing systems 30 (2017).
[2] Garnelo, Marta, et al. "Pick your battles: Interaction graphs as population-level objectives for strategic diversity." AAMAS (2021).

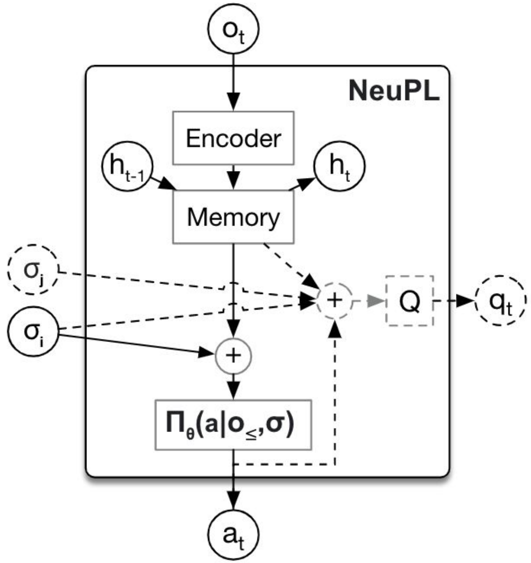
Neural Population Learning

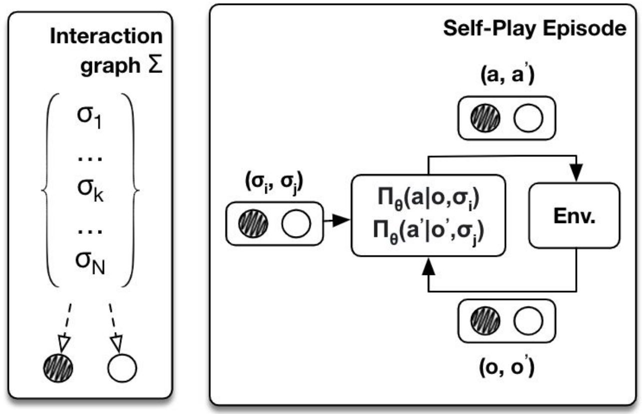
$$J_{\sigma_i} = \mathop{\mathbb{E}}_{\sigma_j \sim P(\sigma_i)} \left[\mathop{\mathbb{E}}_{a \sim \Pi_\theta(\cdot|o_{\leq t}, \sigma_i), a' \sim \Pi_\theta(\cdot|o'_{\leq t}, \sigma_j)} \left[\sum_t r_t \gamma^t \right] \right]$$

$$\begin{array}{cc} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ p & 1-p & 0 \end{bmatrix} & \begin{array}{c} \sigma_0 \\ \sigma_1 \\ \sigma_2 \end{array} \end{array}$$

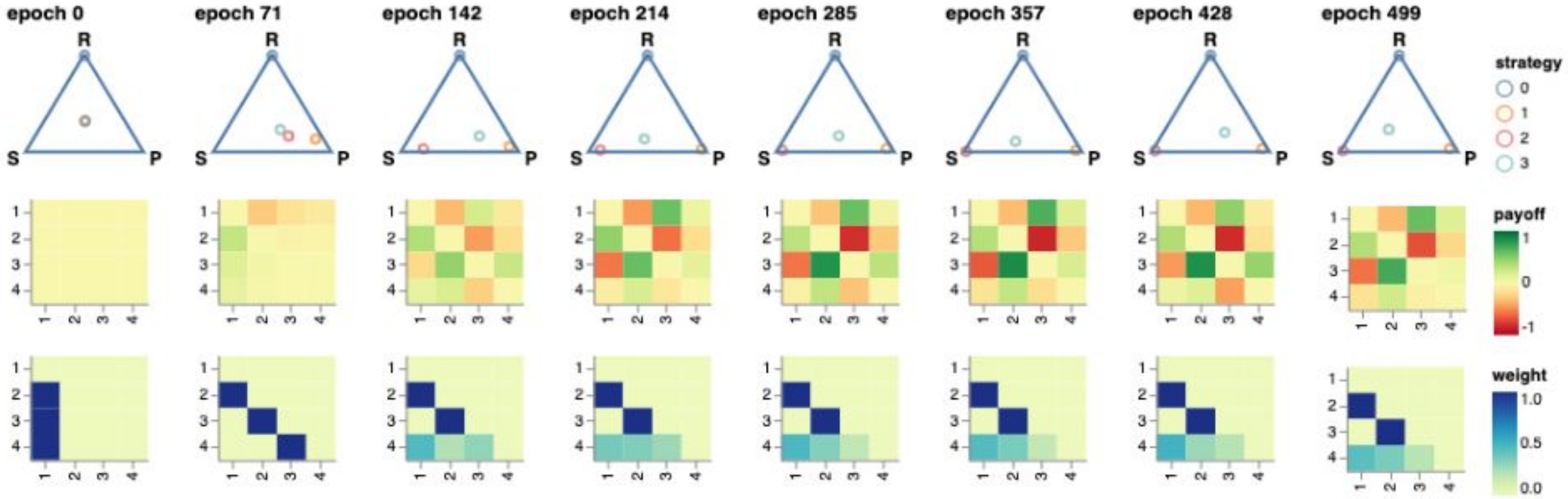


PSRO-NASH

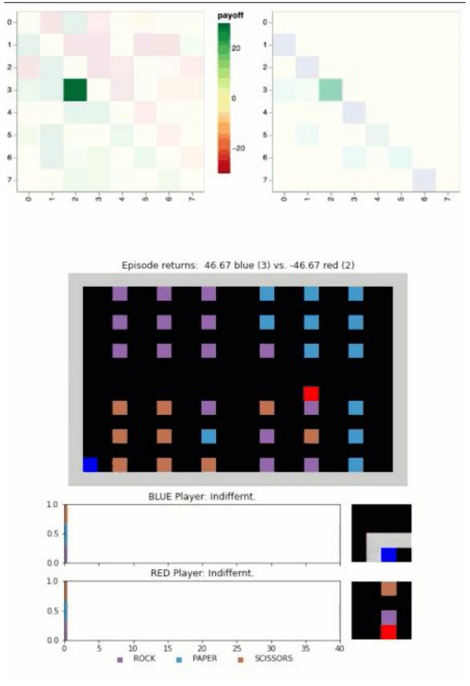


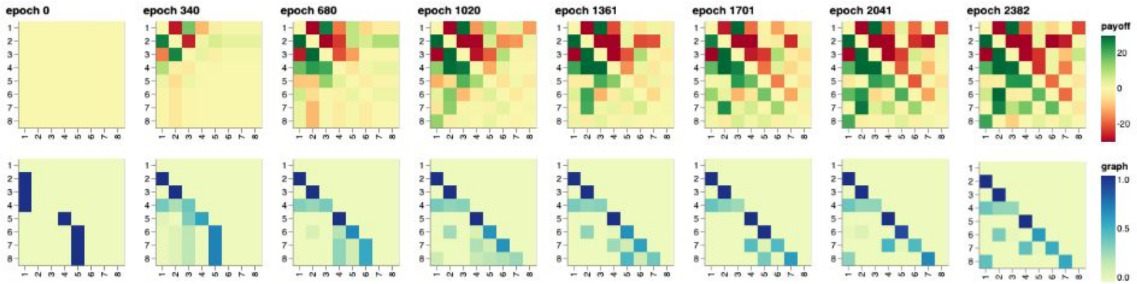


Rock-Paper-Scissors

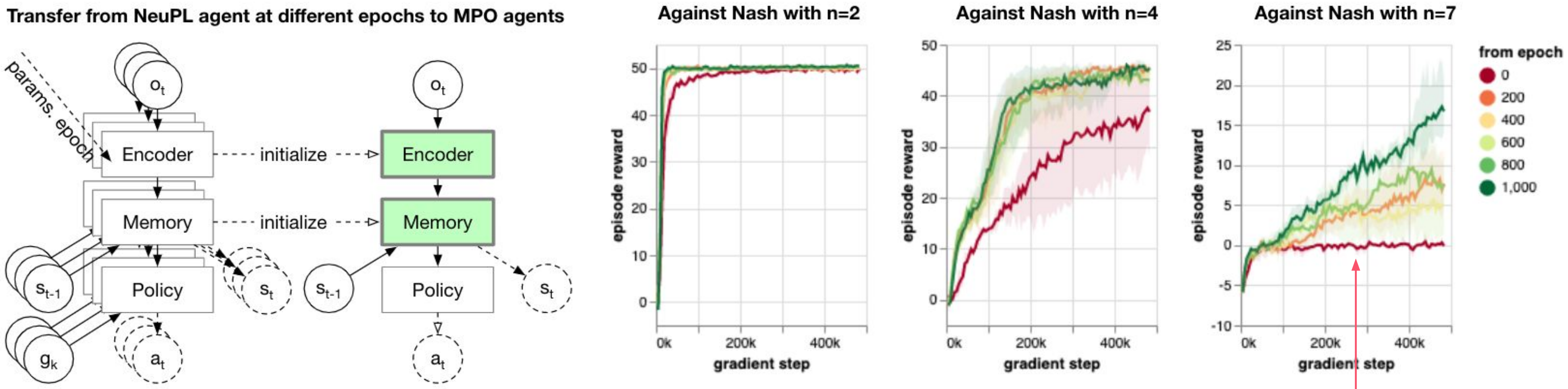


Running-with-Scissors





Running-with-Scissors



Without transfer learning, agent failed to exploit strong opponents!

Conclusion & Future Works

- **Game-Theoretic:** preserves convergence guarantees to NE (see Appendix C for proofs);
- **Transfer of skills** across strategies: learning the N+1$^{\text{th}}$ strategy becomes easier;
- **N-step Best-Response:** NeuPL yields a sequence of *N-step best-responses* instead of “good-”responses;
- **Efficient & Scalable:** represents a population of strategies within a single conditional network, using the compute resources of *self-play*.

- **Future Works:**
  - Beyond symmetric zero-sum games.

Algorithm 1 Neural Population Learning (Ours)

1: $\Pi_\theta(\cdot|s,\sigma)$ ▷ Conditional neural population net.
2: $\Sigma := \{\sigma_i\}_{i=1}^N$ ▷ Initial interaction graph.
3: $\mathcal{F} : \mathbb{R}^{N\times N} \rightarrow \mathbb{R}^{N\times N}$ ▷ Meta-graph solver.
4: **while** true **do**
5: $\quad \Pi_\theta^\Sigma \leftarrow \{\Pi_\theta(\cdot|s,\sigma_i)\}_{i=1}^N$ ▷ Neural population.
6: $\quad$ **for** $\sigma_i \in \textsc{Unique}(\Sigma)$ **do**
7: $\quad\quad \Pi_\theta^{\sigma_i} \leftarrow \Pi_\theta(\cdot|s,\sigma_i)$
8: $\quad\quad \Pi_\theta^{\sigma_i} \leftarrow \text{ABR}(\Pi_\theta^{\sigma_i}, \sigma_i, \Pi_\theta^\Sigma)$ ▷ Self-play.
9: $\quad \mathcal{U} \leftarrow \textsc{Eval}(\Pi_\theta^\Sigma)$ ▷ (Optional) if $\mathcal{F}$ adaptive.
10: $\quad \Sigma \leftarrow \mathcal{F}(\mathcal{U})$ ▷ (Optional) if $\mathcal{F}$ adaptive.
11: **return** Π_θ, Σ