# SMART TEAM PLAY: UTILITY OF POPULATION-BASED TRAINING FOR COOPERATIVE AI IN OVERCOOKED


Author
Janaína Moreira-Kanaley
J.M.Moreira-Kanaley@student.tudelft.nl

Supervisor
Robert Loftin
R.T.Loftin@tudelft.nl

Responsible Professor
Frans Oliehoek
F.A.Oliehoek@tudelft.nl

TU Delft


## 1. INTRODUCTION

**Context**

- Cooperative games provide an isolated and risk-free opportunity to evaluate human-AI cooperation

**Overcooked**

- Cooperative game which requires the collaboration of up to four players to complete cooking tasks

**Population-Based Training (PBT)**

- Evolutionary training algorithm [1]

**Proximal Policy Optimization (PPO)**

- Reinforcement learning algorithm, used in this case to update the policies during PBT [2]

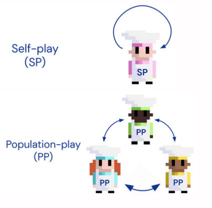



**Figure 1.** Depiction of self-play and population-based training methods. Illustrations belong to [3].

## 2. OBJECTIVE

The main objective of the research is to answer the following questions:

- **RQ1**: How does a PBT agent perform in a cooperative environment when paired with a human player?
- **RQ2**: What variations to PBT could improve an agent's performance with a human player?

## 3. METHOD

**Performance Measurements**

- Learning curves
- Final performance with human proxy

**Experiment Variations**

- Mutation iterations in the initial PBT population

**Baseline Agents**

Agents used to evaluate the performance of the PBT agent:

- **HProxy:** human model that simulates the human player
- **SP:** PPO agent trained using the self-play method
- **$PPO_{HProxy}$**: PPO agent trained with the human proxy
- **$PPO_{BC}$**: PPO agent trained with the human model BC
- **BC**: imitation agent based on a separate subset of the data used to train the human proxy

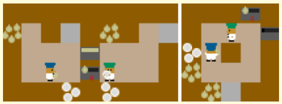
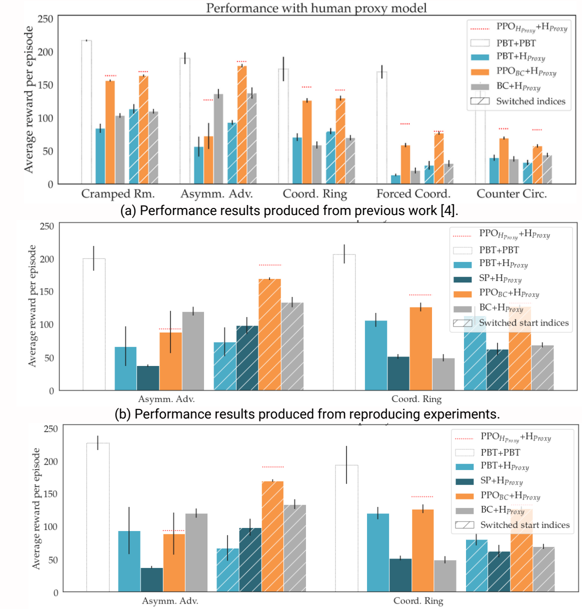


**Figure 2.** Room layouts used for the experiments. From left to right: *Asymmetric Advantages*, *Coordination Ring*. Image is taken from [4].

## 4. RESULTS

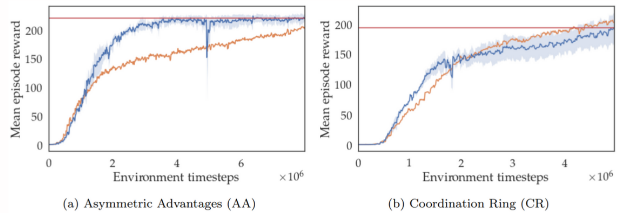



(a) Performance results produced from previous work [4].

(b) Performance results produced from reproducing experiments.

(c) Performance results produced from variations to the experiments.

**Figure 3.** Performance results of the mutated PBT agent when paired with the human proxy in comparison to performances of other agents matched with the proxy. For each layout, performance is given by the average sparse reward per episode (mean of 100 episodes) over 400 horizon timesteps. Hashed bars indicate results from switching the starting positions of the agents.




(a) Asymmetric Advantages (AA) (b) Coordination Ring (CR)

**Figure 4.** Learning curves of a PBT agent displaying the average sparse reward per episode (mean of 100 episodes) during training over 400 horizon timesteps. Orange indicates the unchanged agent while blue the mutated agent, with the shaded area indicating its standard error over three seeds. The red line shows the mutated agent's last performance reached.

## 5. ANALYSIS OF RESULTS

**Findings from Reproducing Experiments**

- PBT underperforms when paired with a human proxy and against agents trained on human data
- PBT outperforms self-play
- Results confirm the conclusions derived in the previous research [4]
- Poor sample efficiency for layouts with low risk of agent collison

**Findings from Variations to the Experiments**

- Improves sample efficiency for layouts with low risk of agent collison. Learning curves in other layouts stay about the same
- Very little effect on final performance

## 6. CONCLUSIONS

- **RQ1 Answer**: While PBT improves on self-play when paired with the human proxy, it underperforms against agents trained on human data.
- **RQ2 Answer**: Although introducing mutation iterations to the initial PBT population increases sample effiency for some layouts, more research is necessary to determine if it improves final performance.

**Limitations**

- Limited computational power & research time

**Future Work**

- Continued research into effects of increasing population diversity
- Investigate influence on final performance from incorporating BC agent into PBT population

## References


[1] M. Jaderberg, V. Dalibard, S. Osindero, et al., "Population based training of neural networks," CoRR, vol. abs/1711.09846, 2017. arXiv: 1711.09846. [Online]. Available: http://arxiv.org/ abs/1711.09846.

[2] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," CoRR, vol. abs/1707.06347, 2017. arXiv: 1707.06347. [Online]. Available: http:// arxiv.org/abs/1707.06347.

[3] D. Strouse, K. R. McKee, M. M. Botvinick, E. Hughes, and R. Everett, "Collaborating with humans without human data," CoRR, vol. abs/2110.08176, 2021. arXiv: 2110.08176. [Online]. Available: https://arxiv.org/abs/2110.08176.

[4] M. Carroll, R. Shah, M. K. Ho, et al., "On the utility of learning about humans for human-ai coordination," CoRR, vol. abs/1910.05789, 2019. arXiv: 1910.05789. [Online]. Available: http:// arxiv.org/abs/1910.05789