Abstract (200 words)-

Introduction(500 words what is the Iterated prisoner dilemma)-

Related Works(200 words)-

Background: Proximal Policy Optimisation (How they work) (600 words)-

Method (Results need to be obtainable from repeating what’s stated in method.) (What and how are you investigating) (600 words)-

Results (Graphs and results, Analysis, Comparison to others, Impact of results(present and future)) (1000 words)-

Conclusion-(200 words)-

Future Work(200 words)-

Legal, social, ethical, professional, and sustainable issues(400 words)-

Flaws in experiment

Not randomised quantity of rounds. This can lead to AI remembering how Many rounds there are and defect on last round.

AI is only told to get as many individual points as possible. This means that the AI isn’t facing a dilemma but rather a challenge or task. But current models and research for AI doesn’t allow for it to decided on dilemmas without being told which is the desired answer beforehand.

Simple IPD I could further this by adding random “mistakes” to the strategies and Agent’s choice. This could lead to the Ai getting rewarded for its mistakes.

The way the environment is set up the Agent is only battling against one AI at a time. It gets to find the strategy that works the best against each other strategy individually rather than sticking to a single strategy this could be changed by having it face off against random strategy and having to learn from the opponents’ actions rather than only on its own decisions.

References-

[1]

R. Axelrod, ‘Effective Choice in the Prisoner’s Dilemma’, *The Journal of Conflict Resolution*, vol. 24, no. 1, pp. 3–25, 1980.

[1]

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, ‘Proximal Policy Optimization Algorithms’. arXiv, Aug. 28, 2017. Accessed: Feb. 29, 2024. [Online]. Available: <http://arxiv.org/abs/1707.06347>

[1]

U. Technologies, ‘Build More Engaging Games with ML Agents | Unity’. Accessed: Feb. 29, 2024. [Online]. Available: <https://unity.com/products/machine-learning-agents>

[1]

‘Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine’, Unity. Accessed: Feb. 29, 2024. [Online]. Available: <https://unity.com>

[1]

M. Harper, V. Knight, M. Jones, G. Koutsovoulos, N. E. Glynatsi, and O. Campbell, ‘Reinforcement learning produces dominant strategies for the Iterated Prisoner’s Dilemma’, *PLOS ONE*, vol. 12, no. 12, p. e0188046, Dec. 2017, doi: [10.1371/journal.pone.0188046](https://doi.org/10.1371/journal.pone.0188046).

[1]

A. Dolgopolov, ‘Reinforcement Learning in a Prisoner’s Dilemma’.

[1]

‘Axelrod-Python · GitHub’. Accessed: Mar. 03, 2024. [Online]. Available: <https://github.com/Axelrod-Python>

[1]

‘Unity-Technologies/paper-ml-agents’. Unity Technologies, Dec. 14, 2023. Accessed: Mar. 03, 2024. [Online]. Available: <https://github.com/Unity-Technologies/paper-ml-agents>

Figure

[1]

‘FIG. 3. The prisoner’s dilemma, a game in normal form played by...’, ResearchGate. Accessed: Mar. 03, 2024. [Online]. Available: <https://www.researchgate.net/figure/The-prisoners-dilemma-a-game-in-normal-form-played-by-spacelike-separated-agents-The_fig8_334223540>