

# Support Rather Than Assault – Cooperative Agents in Minecraft

Danielle Potts  
Accenture UK  
Sunderland, SR68JP  
[danielle-19-93@live.com](mailto:danielle-19-93@live.com)

Kate MacFarlane  
University of Sunderland  
Sunderland, SR60DD  
[kate.macfarlane@sunderland.ac.uk](mailto:kate.macfarlane@sunderland.ac.uk)

Lynne Hall,  
University of Sunderland  
Sunderland, SR60DD  
[lynne.hall@sunderland.ac.uk](mailto:lynne.hall@sunderland.ac.uk)

With the dominant trope of the computer as adversary rather than enabler, reinforcement learning for games has mainly focused on the ability of agents to compete and win. Although cooperation is a product of learning, of understanding the player's requirements and applying agents' competences to fulfil them, there has been little investigation of reinforcement learning for cooperation in games. Reinforcement learning results in the agent adapting and changing, however, there are concerns that such adaptivity could alienate users if their cooperative agent outperforms them. To explore this, the paper outlines the development and training of cooperative agents reporting users' positive response to adaptive cooperation in games.

*Reinforcement Learning, Cooperative Interaction, Games Artificial Intelligence, Collaboration, Minecraft*

## 1. INTRODUCTION

This research explores how games technology and adaptable Artificial Intelligence can be fused to explore the gaps left by the lack of recent consideration of cooperative agents for players in non-competitive games worlds. It aimed to allow real-world users to interact with an adaptable agent in a game world. Unlike in most games, the agent would not be a competitor nor a danger. Rather, it would provide some sort of practical collaborative aid to the user. However, key challenges lie not only in identifying how agents can quickly learn to be useful and adapt, but also in their acceptability to users as cooperative agents.

Machine learning, in particular deep learning, has seen a surge of interest in recent years (Botvinick et al., 2019). It has been successfully implemented in previously unachievable tasks such as language translation and object detection (Arulkumaran, Deisenroth, Brundage and Bharath, 2017). Becoming cooperative, learning to be useful in response to a player, to assist rather than attack, is one such challenging task.

Machine learning is heavily used in games development (Kaliappan and Sundararajan, 2020) with games providing a powerful test bed for machine learning research (Dann, Zambetta and Thangarajah, 2018). Reinforcement learning has had particular success in games software (Arulkumaran, Deisenroth, Brundage and Bharath,

2017; Nair et al., 2018)) – with the goal of outdoing expert users and has been successfully applied to play games such as Chess, Atari, Doom and Starcraft (Botvinick et al., 2019, Xu and Chen, 2019). Reinforcement learning research in games has mainly focused on competitive agents, which compete with the player, they increase their skill level based on the progress of the player (Barros et al, 2020).

This paper discusses an alternative to competitive Non-Player Characters (NPCs), the development and training of NPCs that are adaptive agents designed to support a user in their game tasks. Section 2 introduces relevant related research. Section 3 explores the application of machine learning techniques to create cooperative agents. Section 4 reports a study of user responses to these cooperatively adaptable agents. The final section considers the results and future directions.

## 2. RELATED RESEARCH

AI is an intricate part of modern games for both world building and to increase challenge – a core piece of the experience (Xu and Chen, 2019). Most games use traditional techniques such as finite state machines (de Almeida Rocha and Cesar Duarte, 2019). However, with finite state machines, agents will always be limited to their routines and play styles. Velardo, (2019) discusses this in the context of the popular sandbox game Red Dead

Redemption. There are thousands of NPCs in this game with seemingly complex behaviour changing dependent on player actions, however, the behaviours are still limited to a finite set.

Machine learning driven adaptive AI provides unique features with new behaviours and routines developing overtime, increasing the realism of the environment and allowing agents to better adapt to player skill levels. Game difficulty increasing as a player's performance improves is common in games. Microsoft attempt to utilise this with their Forza series. Forza employs adaptive techniques to adapt agents to human skill (Walsh, 2018) as the users play. Machine learning is used to estimate players' skill levels and to personalise bot skill levels for the player to compete against (Orland, 2013).

In *The Last of Us Part II*, developers Naughty Dog pushed the AI of their NPCs to achieve a higher level of realism for its players (Hara, 2020: Online). These characters would call out to nearby NPCs for help and clutch the area on their body where they had been wounded by the human player. They also appeared to have relationships with other NPCs, which manifested in NPCs appearing to be getting more aggressive if they saw their 'friends' under attack. While this gives the player another level of immersion into the environment, it is an increase to the threat level and game difficulty for the player.

Kahn, (2017) noted that competitive adaptable agents can appear threatening when they out compete people. Often, their purpose is to be better than expert players. This itself poses an issue with adaptable agents, especially in competitive games. The experience of agents that can endlessly adapt may appear unfair and frustrate users. A proposed solution to this is to avoid using competitive style games to introduce these agents, but rather instead introduce them in games that promote cooperation.

There are some examples of cooperative NPCs providing help and support to players in competitive gaming experiences. A notable example is *Star Wars: The Old Republic* (2021: Online). In this game the player has a choice of companion characters to travel alongside them as they level up and complete missions in this massively multiplayer online game from BioWare. These companion characters can provide a range of support roles (act as a healer, help with combat, or take damage) to the player character. This proved to be very popular with the player base for the game, with the relationships with the various companion characters being integral to their overall gaming experience.

There remains a gap, a continuing lack of emphasis on experiences that are non-competitive. However, in many other application areas of machine and reinforcement learning, the emphasis has not been on competition but rather on improving work processes (Daugherty and Euchner, 2020).

### 3. COOPERATIVE AGENTS IN MINECRAFT

The project aimed to develop and assess an adaptable agent who would, eventually through a period of training provide some sort of practical collaborative use to the user within a game world. The goal was not to produce a competitive agent, as much of the reinforcement learning research does, rather an agent who could add value for the user.

#### Game World - Minecraft

The game world selected for the user experience was Minecraft, currently one of the most successful computer games, with over 140 million registered user accounts, (Clement, 2021: Online). Minecraft was selected because it is an open world sandbox game, which gives players total freedom over what they do and how they play. They can choose to build, explore, or fight enemies – there is no single goal. And further, Minecraft is not competitive, with players often working together to achieve goals.

Minecraft as a deep learning task is well researched but poses some challenges such as the amount of time required just to achieve a single goal (Scheller, 2020). Further, because of its hierarchical nature, how best to tackle this issue is contested. Reynard, Kamper, Engelbrecht and Rosman, (2020) explored tackling the issue as a steppingstone style issue, creating many small tasks for the agent to solve before introducing it to the fully complex game world. They compared this to throwing the agent straight into the world, a sink or swim approach, discovering that building up from smaller to larger tasks was a more promising approach. The types of small problems given to the agent, and the data and rewards impact the learning outcomes for the agent.

Jaderberg et al., (2016) explore how this level of general intelligence requires the agent to solve many smaller auxiliary goals along the way. The issue with this is that the reward can become skewed, in that, the agent can become distracted from the ultimate goal. Moreover, the task of deciding what should be a reward and what shouldn't be is considered the sparse reward problem common in other environments besides Minecraft (Nair et al., 2018). Perez et al., (2019) addresses achieving the many subtasks in Minecraft using multi-agents, whereby one agent may be very good at navigating, and another may be very good at mining wood. Despite the challenges, Minecraft was chosen because it is not competitive.

#### Goals of the Agent and User

Minecraft requires a player to mine materials before they build structures. Mining resources can take some time and can become repetitive. Thus, an agent to help the player with this was the goal. The Minecraft agent was to be trained to do two things:

1. To navigate around the world logically
2. To mine trees. Mining trees produces wood, a necessary resource for creating tools and weapons in the game.

A user would be required to interact with the game as normal - simply to play the game as they wish. The agent could then be used as helper to gather a particular resource, in this case, wood. This would help to speed up their progress and remove a lot of the repetitive nature of the game.

### Cooperative Agent Development

Reinforcement learning (RL) was the method selected to train the cooperative agent due to its effectiveness in game environments. It has been applied to Minecraft before (Milani et al., 2020). The solution to training the agent would use a deep learning model due to the high complexity of Minecraft and the size of the dataset.

RL focuses on the reward, but how to get to that reward is not always clear. Most methods will give the agent no prior knowledge about the environment it is expected to work within. In many cases, this solution works because the environment is of low dimensionality or complexity. As the environment complexity increases, so too does the difficulty.

When humans approach a problem, they rarely have zero knowledge about the domain or how to solve it. They consider a solution with existing knowledge, even if the new problem is not one within known experiences, able to make assumptions based on an accumulation of experiences (Efros et al., 2018). Replicating this, before entering an environment the agent is given some domain information. For example, in (Hester et al., 2017), the agent first learns about the problem via a deep neural network before interacting with the environment.

Q-Learning extends reinforcement learning by implementing an estimation technique, which aims to satisfy the Bellman Equation used to estimate future rewards. Given a current state and available actions it will perform in subsequent states based on the decision at the current point. This method is used to assess the goodness of a particular policy given the following state action pairs observable at future states (Sutton et al., 2018).

Deep Q-Learning subsequently implements deep neural networks into the equation and uses two networks. One network is the network being trained; the other is the 'target' network. When the network evaluates its actions, it measures the loss against the target network. Every so many episodes during training, the target network is updated by copying the parameters of the trained network promoting continuous learning (Silver et al., 2016). The output is an action along with a prediction, which is a

percentage that represents the likelihood that each action will lead to reward maximisation.

To train the agent to play Minecraft, Hester et al.'s, (2017) work was the primary inspiration for the final model type – Deep-Q Learning from Demonstrations (DQfD). This is an imitation learning style approach to Deep Q-Learning. It utilises a Deep Neural Network for policy optimisation and a standard Convolutional Neural Network for image recognition (Kunanusont et al., 2017). The Python coded cooperative agent used Tensorflow and Keras.

### Using MineRL to train the agent.

To train the agent the MineRL dataset was used. This is composed of 500 hours of image-based demonstrations broken down into different tasks and goals such as navigation and tree chopping, the two selected tasks for the agent. It has over 60 million state action pairs of human demonstration from the game. Observations are made as arrays of low-resolution images making them easier to compute. These were then pre-processed as standard for convolutional neural networks. Google Collab was used to speed up data pre-processing and parsing.

Two MineRL environments and datasets were used for training: Navigate and Treechop. The goal of navigate was to learn how to logically navigate the environment. This includes learning to walk, jumping in certain places and placing dirt to reach high places. In Treechop, the agent had to learn two things: how to walk and how to cut down trees.

### Cooperative Agent Outcomes

Whilst the model did not entirely solve the issues it faced; it did make some progress. In Navigate, the agent was able to navigate the environment. In Treechop, the agent often did attempt to mine things. It struggled to successfully locate and target trees. There was clear evidence to show the agent was learning, for example with Treechop the agent's most common action became attack with rewards increasing over time. However, to be able to find the trees the agent also needs improve navigation, something which proved a challenge for the agent.

## 4. USER STUDY

The purpose of the user study was to explore reactions to the cooperative agent and to investigate whether this engendered trust in cooperative intelligent agents in other environments.

### Method

The original goal for the demonstration application was to allow users to try it in person. However, due to the Covid-19 pandemic this was not possible. With Minecraft being a paid application – the

logistics of expecting people to set the application up on their machines was limited. To get around this challenge, a video of the agent was recorded which was shared with a survey to gather some insight about people's feelings towards AI and agents.

The users were given the context around the study, that this was an experimental application of AI in Minecraft looking at cooperative agents. They were then requested to watch a video containing a Minecraft agent attempting to mine some trees. After this, the users would be requested to answer a survey to record their reception and opinions around the video and to explore opinions around this kind of implementation and use of AI.

31 participants engaged in the study, a mixture of people from those who worked within tech and were at least familiar with machine learning and AI, and to those who had no experience or knowledge of it. Participants were recruited through personal contacts and opportunistic sampling due to the pandemic context.

### Results & Interpretation

61% of respondents said that they somewhat trusted artificial agents, 31% did not trust them at all and only 8% had a positive level of trust in them. This lack of trust is also seen other with AI-based technologies, for example some people distrust voice assistants believing that they are continually and cleverly listening to them. The survey results indicated that participants were wary of intelligent agents suggesting they might be predisposed to have a negative view of the helper agent.

However, just as the lack of trust in Alexa does not deter use, neither does the lack of trust in agents impact on the use or intention to use agents. 76% of participants indicated that they believed that they regularly interacted with artificial agents on a daily basis. Participants indicated that they were willing to work with intelligent agents, 39% stated that they would be willing to work with an agent with general intelligence, 44% might be willing to and 17% said that they would not be willing to work with them.

Participants were keener to interact with agents in their personal life with 48% happy to interact with an agent out of the workplace and a further 32% who might be. There was even more willingness to engage when agent usefulness was considered, like the helper agent in Minecraft. When asked if they would find it useful to have a helper agent in Minecraft to carry out basic tasks, 64% of the respondents said that they would, 22% thought that it might be a useful addition and 14% said they would not find it useful. Only 1 participant would still feel threatened by agents like the one in the demonstration video, 14% said that they might feel threatened, but the majority (83%) said that they would not feel threatened.

However, participants identified they would be less comfortable with a helper agent in the workplace. 23% of the sample believed that AI would impact the workplace in a negative way. However, mainly the view was positive with 57% of participants believing that AI would enhance the workplace and create new types of jobs. In comparison, many of the participants still felt that AI/agents were somehow threatening with 54% believing that AI is threatening to some degree.

Generally, people found the idea of having an agent in a game as a helper to be positive. Participants also suggested other types of tasks in Minecraft, as well as other games in which this type of AI could be useful to them, like defending against enemies and to help those with increased accessibility needs.

## 5. CONCLUSION AND FUTURE WORK

Games provide a powerful test environment for reinforcement learning research. Such gamification provides a powerful tool for training algorithms before integrating them into real world systems (Riedmiller et al., 2018). Moving towards more cooperative gameplay has significant potential for some player experiences. Using adaptable helper agents could lead to greater game longevity and more immersive gaming in both competitive and non-competitive gaming environments.

This study has highlighted the potential of cooperative agents in game worlds. A key issue that this study identified is the lack of trust in AI-based technologies. Cooperative helper agents offer a clear route to engender higher levels of trust in AI systems, particularly, if this was the guise in which agents were first introduced to users.

However, there are significant limitations to the study at this point, primarily that user engagement involved watching a video rather than interacting with the agent in Minecraft.

Future work focuses on replicating the experiment with the user working on activities with a need for wood, with the agent cooperating by chopping that wood. This will benefit from a more refined agent architecture where the agent is provided with the underpinning skills necessary to achieve the high-level task. The Minecraft helper agent could be further developed to provide other support services like gathering food and build tasks.

## REFERENCES

Arulkumaran, K., Deisenroth, M., Brundage, M. and Bharath, A, "Deep Reinforcement Learning: A Brief Survey". *IEEE Signal Processing Magazine*, 34(6), 2017, pp.26-38, Doi: 10.1109/MSP.2017.2743240

- Barros, P., Tanevska, A. and Sciutti, A., 2020. "Learning from Learners: Adapting Reinforcement Learning Agents to be Competitive in a Card Game", viewed 10 May 2021 <https://arxiv.org/abs/2004.04000>
- Clement, J., 2021, *Minecraft number of players worldwide*. Statista. Available at: <https://www.statista.com/statistics/680139/minecraft-active-players-worldwide/> (accessed 5.10.21).
- Dann, M., Zambetta, F. and Thangarajah, J., 2018. "Integrating Skills and Simulation to Solve Complex Navigation Tasks in Infinite Mario", *IEEE Transactions on Games*, 10(1), 2018 ,pp.101-106, doi: 10.1109/TCIAIG.2017.2696045
- Daugherty, P. and Euchner, J., 2020. "Human + Machine: Collaboration in the Age of AI", *Research-Technology Management*, 63(2), 2020, pp.12-17, doi: <https://doi.org/10.1080/08956308.2020.1707001>
- de Almeida Rocha, D. and Cesar Duarte, J., 2019. "Simulating Human Behaviour in Games using Machine Learning", *2019 18th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, 2019, doi: 10.1109/SBGames.2019.00030
- Efros, A., Griffiths, T., Pathak, D., Agrawal, P. and Dubey, R., 2018. "Investigating Human Priors for Playing Video Games", *International Conference on Machine Learning*, viewed 19 December 2020, <https://arxiv.org/abs/1802.10217>
- Gough, C., 2020. *Minecraft Unit Sales Worldwide 2020* | Statista. Available at: <https://www.statista.com/statistics/680124/minecraft-unit-sales-worldwide/#:~:text=Minecraft%20unit%20sales%20worldwide%202016%2D2020&text=Since%20its%20release%20in%202011,and%20Grand%20Theft%20Auto%20V> (Accessed: 19 December 2020)
- Hara, R., 2020. The Last of Us 2 Dev Reveals Major AI Improvements. *Game Rant*. Available at: <https://gamerant.com/last-of-us-2-ai-improvements/> (Accessed: 10 May 2021).
- Hester, T., Vecerik, M., Pietquin, O., Lanctot, M., Schaul, T., Piot, B., Horgan, D., Quan, J., Sendonaris, A., Dulac-Arnold, G., Osband, I., Agapiou, J., Leibo, J. and Gruslys, A., 2017. "Deep Q-learning from Demonstrations", *Association for the Advancement of Artificial Intelligence*, viewed 19 December 2020, <<https://arxiv.org/abs/1704.03732>>
- Jaderberg, M., Mnih, V., Czarnnecki, W., Schaul, T., Leibo, J., Silver, D. and Kavukcuoglu, K., 2016. "Reinforcement Learning with Unsupervised Auxiliary Tasks", *DeepMind*, viewed 19 December 2020, <https://arxiv.org/abs/1611.05397>
- Home, 2021. *The Elder Scrolls Online*. Available at: <https://www.elderscrollsonline.com/en-us/home> (accessed 5.10.21).
- Kahn, J., 2017. *Bloomberg - Are You A Robot?* Bloomberg.com Available at: <https://www.bloomberg.com/news/articles/2017-03-31/video-games-without-the-players> (Accessed 19 December 2020).
- Kaliappan, J. and Sundararajan, K., 2020. "Machine Learning in Video Games. Handbook of Research on Emerging Trends and Applications of Machine Learning", 2020, pp.425-443, doi: 10.4018/978-1-5225-9643-1
- Kunanusont K, Lucas SM, Perez-Liebana D (2017). General Video Game AI: Learning from screen capture. DOI: 10.1109/CEC.2017.7969556
- Milani, S., Topin, N., Houghton, B., Guss, W., Mohanty, S., Nakata, K., Vinyals, O. and Kuno, N., 2020. "Retrospective Analysis of the 2019 MineRL Competition on Sample Efficient Reinforcement Learning", viewed 19 December 2020 <<https://arxiv.org/abs/2003.05012>>
- Nair, A., McGrew, B., Andrychowicz, M., Zaremba, W. and Abbeel, P., 2018." Overcoming Exploration in Reinforcement Learning with Demonstrations", *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, doi: 10.1109/ICRA.2018.8463162
- Orland, K., 2013. How Forza 5 and the Xbox One Use the Cloud to Drive Machine-Learning AI. *Ars Technica*. Available at: <https://arstechnica.com/gaming/2013/10/how-forza-5-and-the-xbox-one-use-the-cloud-to-drive-machine-learning-ai/> (Accessed 19 December 2020)
- Perez D, Liu J, Abdel Samea Khalifa A et al. (2019). General Video Game AI: a Multi-Track Framework for Evaluating Agents, Games and Content Generation Algorithms. DOI: 10.1109/tg.2019.2901021
- Reynard, M., Kamper, H., Engelbrecht, H. and Rosman, B., 2020. "Combining primitive DQNs for improved reinforcement learning in Minecraft". *2020 International SAUPEC/RobMech/PRASA Conference, 2020*, doi: 10.1109/SAUPEC/RobMech/PRASA48453.2020.9041025
- Riedmiller, M., Hafner, R., Lampe, T., Neunert, M., Degraeve, J., Van de Wiele, T., Mnih, V., Heess, N. and Springenberg, T., 2018. "Learning by Playing – Solving Sparse Reward Tasks from Scratch", *Google Deepmind*, viewed 21 May 2020, <https://arxiv.org/abs/1802.10567>
- Scheller, C., Schraner, Y. and Vogel, M., 2020. "Sample Efficient Reinforcement Learning through Learning from Demonstrations in Minecraft". *NeurIPS2019 Competition & Demonstration Track*,

viewed 19 December 2020  
<<https://arxiv.org/abs/2003.06066>>

Silver, D., Guez, A. and Hasselt, H., 2016. "Deep Reinforcement Learning with Double Q- learning", *Association for the Advancement of Artificial Intelligence*, viewed 19 December 2020, <<https://arxiv.org/abs/1509.06461>>

Star Wars: *The Old Republic* Available at: <https://www.swtor.com/> (accessed 5.10.21)

Sutton, R., Bach, F. and Barto, A., 2018. *Reinforcement Learning*. Massachusetts: MIT Press Ltd, ISBN: 978026203924

Velardo, V., 2019. *The Reality of Red Dead Redemption 2'S AI (Part 1)*. Medium. Available at: <https://medium.com/the-sound-of-ai/the-reality-of-red-dead-redemption-2s-ai-part-1-c276e9da2763> (Accessed 20 December 2020).

Walsh, A., 2018 *Forza Motorsport 7 Drivatars Being Improved, Race Adjudication Pushed To 2019 – Fullthrottle Media*. Available at: [Fullthrottlemedia.co.uk](http://Fullthrottlemedia.co.uk) (Accessed 19 March 2020)

Xu, L. and Chen, Y., 2019. A Hierarchical Approach for MARLÖ Challenge. *2019 IEEE Conference on Games (CoG)*, 2019, doi: 10.1109/CIG.2019.8847943