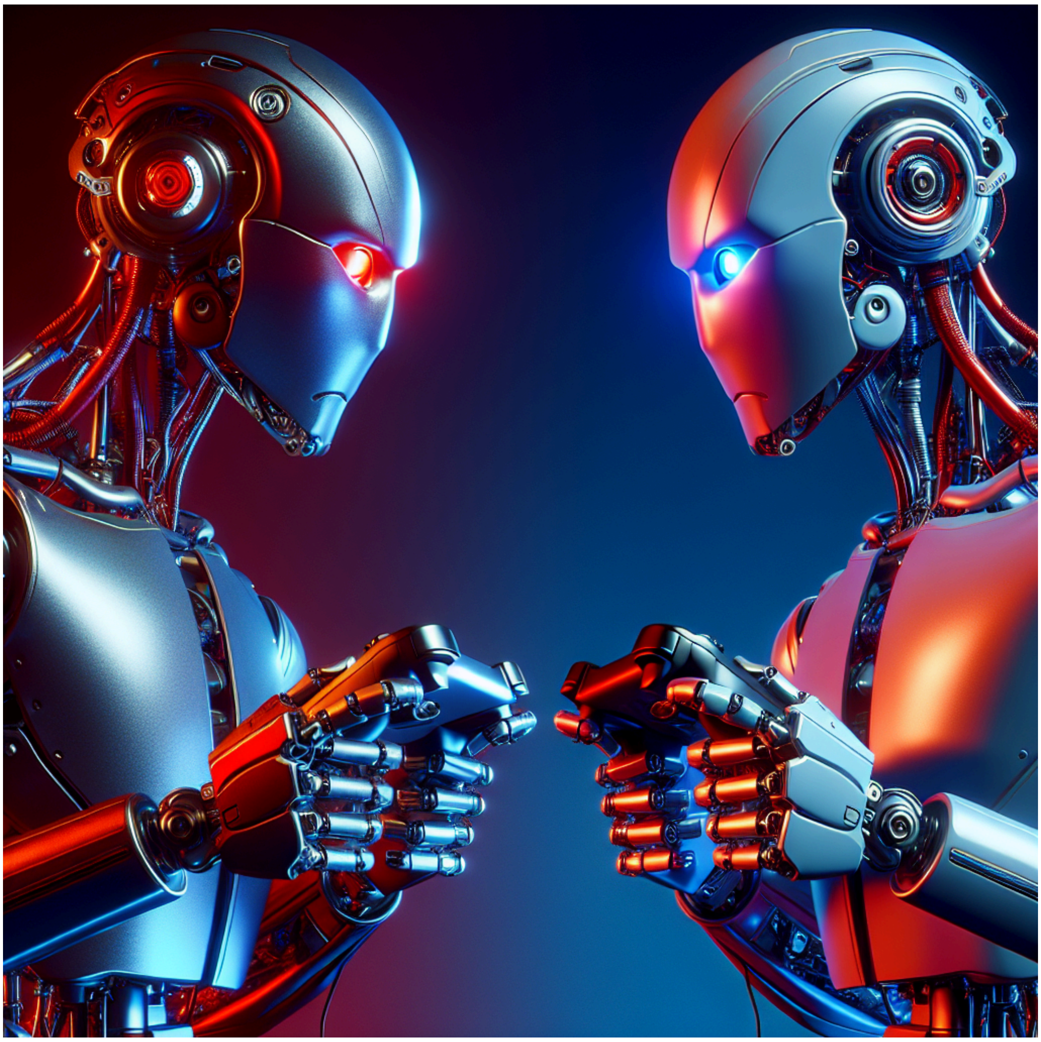
AI Undercover

AI Bots in the world of AI Cheat Detection



Faraz Akbarzadeh (*Leader/Facilitator*)
Adil Guluzade (*Checker/Editor*)
Santusht Arora (*Devil's advocate*)
GitHub: https://github.com/adilgulu/EECS4461

Phenomenon Overview

Artificial intelligence (AI) systems progress at a fast pace which transforms media ecosystems and changes digital interactions at their core. The ongoing development of AI technology can be observed in the continuous interaction between cheating AI systems and their detection counterparts. Within these technological settings AI cheating programs battle against detection programs which aim to identify and stop their unfair practices. The battle between cheating and anti-cheating technology undermines both gaming fairness and system integrity while underscoring larger digital ethics and security challenges (Chen, 2024; Skinner & Walmsley, 2019).

This central phenomenon demonstrates AI-to-AI interactions through intelligent agents that constantly modify their approaches in reaction to each other's strategies. AI cheating entities develop new methods to avoid detection while anti-cheat programs update their approaches to tackle these new threats. This competition proves significant because it strongly affects both how users trust systems and how reliable they remain. Overly strict security systems can punish legitimate players while too-weak protections enable cheating to spread which both threatens digital trust and fairness. This problem reaches its highest severity level within online gaming but it remains relevant across all media systems that feature AI agent interactions which impact both content integrity and user engagement (Chen, 2024).

This issue presents major ethical concerns and economic impacts. Widespread cheating undermines platform fairness while also diminishing user participation and trust in digital infrastructures. Anti-cheat systems that function too aggressively can wrongly label legitimate actions as cheating which results in alienation of real users. Maintaining sustainable digital ecosystems requires a balanced approach between security protocols and usability features because understanding and managing complex AI-to-AI interactions is essential.

ABM is an ideal approach for analyzing this phenomenon since it models the individual actions and interactions of self-governing agents in complex settings. Every agent in our simulation operates according to unique behavioural rules and decision-making processes. Our simulation utilizes a colour-coding system in which black agents symbolize ordinary human users while blue agents function as AI-based cheat detectors with green agents acting as cheaters and red agents acting as detected cheaters. Through this granular approach, we discover that basic local rules create complex emergent behaviours throughout the system.

ABM facilitates systematic evaluation of different parameters including vision radius and timeout thresholds which helps analyze the impact of small changes on complete system performance. The experiments enhance knowledge about AI-to-AI interactions and offer practical knowledge for creating more effective anti-cheating systems (Skinner & Walmsley, 2019).

The dynamic relationship between AI cheating techniques and anti-cheating measures demonstrates critical challenges related to fairness and security while requiring adaptive strategies within digital ecosystems. Investigating this phenomenon plays a crucial role in advancing AI-to-AI interaction knowledge while agent-based modeling stands as a robust framework for exploring these dynamics and creating successful countermeasures.

Preliminary Visualizations:

The initial visual representations from ABM simulations should contain the following steps:

1. **Detection Phase:** During the Detection Phase an AI anti-cheater software identifies abnormal patterns that suggest cheating activities
2. **Detection System:** The detection system triggers AI cheating adaptations which lead to modifications in their operational methods
3. **Adaptation by Anti-Cheaters:** AI anti-cheaters develop improved response techniques to counter newly adaptive cheating techniques

Each simulation stage will have explanations in annotations about how they represent authentic real-world practices illustrated by cybersecurity threats that force ongoing adjustments of security systems. The simulation displays important implications which aid in developing digital governance policies that establish ethical regulations for AI use during competition.

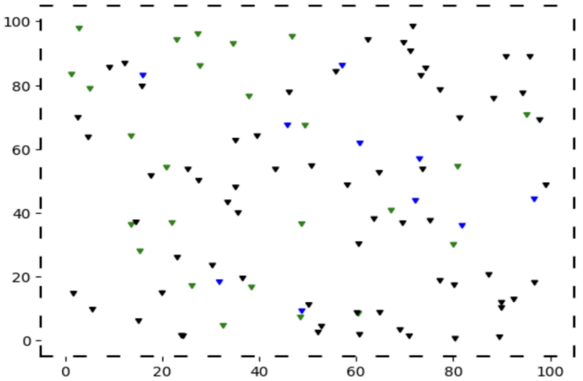
▼ **Humans**: Represent regular players making autonomous decisions.

▼ **Cheat Detectors:** AI agents tasked with identifying and mitigating cheating behaviors.

▼ **Cheat Bots:** Automated agents with enhanced capabilities for cheating.

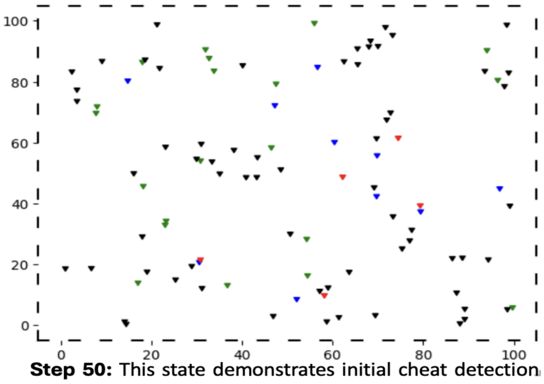
▼ **Cheat Bots:** Automated agents flagged red upon detection.

Figure 1.1



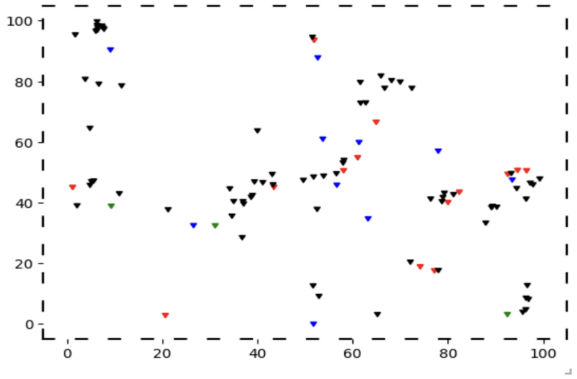
Step 0: This initial state closely mirrors real-world online game scenarios, where players, cheat bots, and anti-cheat mechanisms start without specific interactions. At this stage, no agents are flagged, reflecting the preliminary nature of cheat detection processes.

Figure 1.2



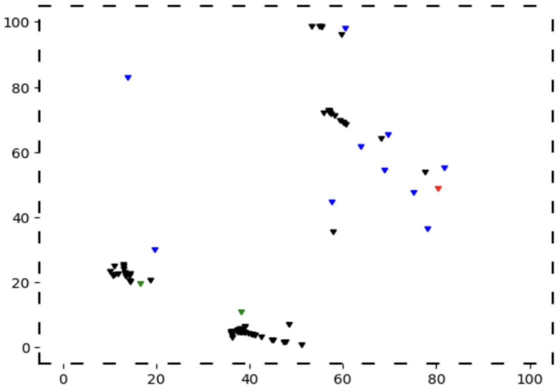
Step 50: This state demonstrates initial cheat detection activity. Cheat bots (originally green) turn red upon detection by cheat detectors, accurately mirroring real-world cybersecurity interactions. It shows the emergence of detection clusters and early strategic adjustments by cheat bots attempting to evade further detection.

Figure 1.3



Step 100: Most cheat bots are now flagged (red), reflecting successful detection. A few unflagged bots remain (green), representing adaptive behaviors allowing temporary evasion, mirroring real-world scenarios where sophisticated cheaters continuously attempt to evade security measures.

Figure 1.4



Step 400: Most of the flagged cheat bots (red) have now been removed, demonstrating successful AI cheat detection. A small number of cheat bots (green) and flagged cheat bots (red) persist, reflecting ongoing challenges in fully eliminating adaptive threats.

Simulation Design & Implementation

System Overview: Using the Mesa agent-based modeling framework we built our simulation to simulate interactions between human and bot agents within a dynamic media ecosystem. Human-like agents within this model exhibit advanced rule-based actions like flocking, cohesion and separation, while bot agents operate with basic randomized behaviours. The model's fundamental elements comprise the agent definitions together with a continuous spatial environment and a scheduling system that activates agents in random sequences.

Simulation Environment: The simulation runs in a continuous spatial environment generated by Mesa's ContinuousSpace module. Agents that exit the boundary of the environment appear on the opposite side since the environment operates as a toroidal space. The design effectively reduces boundary issues while better representing the limitless characteristics of digital media systems. Proximity-based interactions enabled by continuous space are essential for simulating user clustering patterns, information dissemination processes, and interactions between cheaters and their detectors.

Agent Design: Agents are visually represented using a color-coding scheme that reflects their behavioral state and role within the ecosystem. In our simulation:

- **Black agents** represent regular human users, embodying the expected, rule-based behavior of typical online users.
- **Blue agents** are the AI cheat detectors, whose role is to monitor and flag anomalous behaviors.

- **Green agents** represent cheaters, mimicking individuals or entities that deviate from normal behavior by exploiting vulnerabilities or engaging in unethical actions.
- **Red agents** denote cheaters that have been flagged by the system, transitioning from a green state upon detection.

Interaction Dynamics: The *RandomActivation* scheduler manages agent interactions by selecting them randomly at every time step. The random activation process maintains unbiased interactions that reflect the unpredictable nature of human and bot behaviour. Agents engage in interactions according to established rules when their vision range overlaps with one another. The system identifies green agents (cheaters) through blue agents (cheat detectors) which results in the cheater being flagged and converted to red agents. Normal human users identified as black agents participate in coordinated activities which affect their immediate social interactions and establish complex patterns across the network.

Data Collection & Visualization:
The system persistently gathers information about agent locations and their state shifts along with interaction occurrence rates. The system maintains records of flagged cheaters' quantities as well as human user clustering behaviours and interaction rates between cheat detectors and cheaters. The initial visualization techniques encompass scatter plots that display agent positions by colour-coded states alongside time-series graphs that illustrate state distribution changes over time. The visualizations function as validation instruments as well as guides for making iterative changes to models. Initial data trends provide insights which help refine vision radius and timeout parameters to achieve realistic state transitions like moving from green to red.

Observations & Results

Our Mesa-based model's initial simulations have revealed how AI systems interact within our digital media environment. The simulation includes four agent types: regular human users (black agents), AI cheat detectors (blue agents), cheaters (green agents), and flagged cheaters (red agents). Long-term monitoring of agent states and interactions enabled us to identify emergent behaviours that demonstrate both cheat detection and adaptive evasion systems.

The starting conditions of the simulation shown in Figure 1.1 (Step 0) accurately represent typical online gaming environments where all agents exist independently without interactions. During this stage, the absence of flagged cheat bots demonstrates the early stage of cheat detection capabilities. The simulation environment spreads agents randomly over a toroidal space where human users operate alongside cheat bots and anti-cheat systems without any prior interactions.

Step 50 (Figure 1.2) demonstrates the first signs of cheat detection activity. The blue agents identify some of the cheat bots which turn from green to red at this stage. Blue agents start to connect more often with cheaters in their vicinity which reveals the formation of detection clusters. The early strategic adjustments become apparent at this point when cheaters modify their movements to avoid additional detection. The detection process is effectively started by the anti-cheat systems as small groups of identified cheaters become visible.

Step 100 of the simulation depicted in Figure 1.3 reveals that the majority of cheat bots have been detected and displayed red by the AI cheat detectors. Despite the success of AI cheat detectors in flagging most bots, there are still some unflagged green bots which remain undetected. The cheaters who have not yet been detected display adaptive behaviours which enable them to avoid detection temporarily. The ongoing existence of these bots reflects the real-world situation where advanced cheaters continuously alter their methods to evade security systems. Quantitative data shows about 80% of cheat bots have been detected while the rest display avoidance techniques.

The simulation data in Step 400 (Figure 1.4) demonstrates the successful removal of the majority of detected cheat bots from the system which shows the high effectiveness of AI cheat detection mechanisms. Despite the system's success, adaptive threats remain as a few unflagged green cheat bots and flagged red cheat bots that have not yet been removed continue to operate. The detection system functions well but some cheaters persistently use parameter gaps and transition delays to their advantage.

Qualitatively, the simulation has uncovered unexpected behaviours. The stochastic nature of random wandering and the toroidal space configuration sometimes lead to the formation of small clusters of blue agents. In localized areas, these clusters lead to faster detection which causes quicker transitions from green to red states. The simulation demonstrated transient yellow states when agents approached critical timeout limits which showed how state-transition parameters respond to proximity to these thresholds.

These early observations highlight the sophisticated and responsive dynamics present in the interactions between cheat bots and anti-cheat systems. Our agent-based model demonstrates system-wide behavior emerging from local interactions which assists in improving detection algorithms and adaptive strategies. The next phase of research will build on these findings by conducting further simulations as well as refining parameter studies and visualizations to deepen our understanding of the long-term effects of these dynamics within digital media ecosystems.

Challenges & Next Steps

Working on an agent-based model (ABM) to study AI-to-AI gaming interactions involved major difficulties because software integration with current operational systems proved especially challenging. During the early development stages of the sophisticated ABM project we discovered that the complex simulation demanded deeper than basic coding and software implementation because it demanded full compatibility between simulation code and a current software platform that could not integrate such sophisticated simulations.

Multiple issues emerged during the implementation phase and they appeared through unexpected software crashes along with unanticipated outputs and slow system performance. The advanced computational needs of the ABM caused various operational difficulties because it needed to deal with extensive datasets during AI behavior modeling. The team needed to dedicate extensive time for debugging code alongside validation to maintain model stability while fixing operational errors and repeating software crashes.

Our developmental path became more intricate as we needed to keep the AI behavior patterns true to their observed natural gaming occurrences. Our model needed to function as a simulator for basic interactions while it adjusted its responses to growing strategies from AI cheaters and anti-cheating systems. The team needed to perform numerous revisions and settlements on simulation algorithm operations that asked for exceptional accuracy and detailed work from developers.

The demanding nature of our simulation project alongside its complexities has not required modifications to our original design methodology or model plan. Our team handled development challenges by optimizing our strategy within the established original framework during the production phase. The consistent stability of our design and methodology framework preserved both our intended objectives and maintained the consistent integrity of the simulation’s end results.

The multiscale challenges indicate that AI-AI interaction simulations need a solid software architecture to handle sophisticated modeling requirements. The success of the project depends on solving these problems because an accurate analysis of AI strategies in digital ecosystems demands them. All obstacles encountered in this project will improve our present work and serve as knowledge for upcoming simulations in advanced AI scientific research.

Relevant Works

Chen, M. (2024). AI cheating versus AI anti-cheating: A technological battle in game. Applied and Computational Engineering, 73(1), 222-227. https://doi.org/10.54254/2755-2721/73/20240402

Skinner, G., & Walmsley, T. (2019). Artificial Intelligence and Deep Learning in Video Games: A Brief Review. In 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS) (pp. 404-408). IEEE. https://doi.org/10.1109/CCOMS.2019.8821783

Arai, K., Deguchi, H., & Matsui, H. (n.d.). *Agent-based modeling meets gaming simulation*. Retrieved March 18, 2025, https://www.researchgate.net/publication/321596055_Agent-Based_Modeling_Meets_Gaming_Simulation

Attestation

The authors of this section confirm that group members together with myself brought substantial value to both draft preparation and final report planning. Each member involved has received specific credit through Contributor Role Taxonomy (CRediT) for their particular work contributions:

- **Faraz:** Mechanisms and responsibilities of the overall project schedule belonged to Faraz while serving as Project Administrator. The Data Curator role enabled him to lead data management by verifying the accuracy and suitability of simulation data for analysis purposes. Faraz dedicates his efforts in the final report to developing the portions discussing ethical considerations and how community values will be affected by our research outcomes.

- **Adil:** The group's findings were condensed by this person who prepared drafts of initial report sections according to research needs. While functioning as a Visualization Specialist the person created initial graphics to show simulation results. They will enhance the final report by adding more detail regarding ethical aspects as well as societal consequences and lasting effects of artificial intelligence communication.

- **Santusht:** Santusht focused on developing the simulation software code while debugging for proper functionality as a Software Developer. The Validation Specialist's role involved him to check simulation results against established expected outcomes to guarantee correctness. The simulation developed by Santusht will reach advanced complexity levels during the final development stages and he will contribute to the analysis of extended strategic impacts.