Human-like AI in Strategy Games: Guided by Playstyle Profiling and Player Perception

by

Torsten van Ham

In partial fulfilment of the requirements for the degree of
Master of Science
at Delft University of Technology,
to be defended publicly on the 19th of May 2025

Faculty: Mechanical Engineering
Department: Cognitive Robotics
Programme: MSc Robotics

Graduation committee: Dr. A. Zgonnikov
Dr. C. Hao
Dr. M. Mozgovoy
Dr. J.M. Prendergast

An electronic version of this thesis is available at http://repository.tudelft.nl

TU Delft

Keywords:
Human-like AI
Hierarchical Reinforcement Learning
HRL
Playstyle
Believable AI
Strategy Game

Abstract

Strategy games provide a compelling testbed for developing human-like computer agents, with applications that extend beyond gaming into fields requiring adaptive and socially intelligent AI. In these games, players tend to enjoy and engage more deeply with AI opponents that not only provide a challenge but also behave in ways that resemble human thinking and decision-making. However, despite progress in developing such agents, there is still no standard approach for evaluating how human-like these opponents truly are—making it difficult to assess and improve their design. Here I show that strategy game opponents having more human-like game-level playstyles does not necessarily lead to them being more believable (perceived as human-like by human players).

By developing a turn-based strategy game and evaluating Hierarchical Reinforcement Learning (HRL) agents of varying complexity, I assessed both their behavioural similarity to human players and how believable they were perceived to be by human players. This research introduces a new approach for understanding player behaviour using behaviour vectors composed of three high-level metrics—Aggressiveness, Management, and Exploration—consistent with existing literature. These metrics are designed to be broadly applicable across strategy games, enabling consistent comparison between human and AI opponents, as well as across different games and agents. The findings demonstrate that while HRL agents can replicate human-like playstyles without using human training data, players judge human-likeness more on perceived intelligence and fairness. This suggests that creating truly human-like AI opponents requires not just replicating human game-level playstyles, but designing agents that align with players' expectations for intelligent and fair decision-making.

Human-like AI in Strategy Games: Guided by Playstyle Profiling and Player Perception

Torsten van Ham
Delft University of Technology

***Abstract*—Strategy games provide a compelling testbed for developing human-like computer agents, with applications that extend beyond gaming into fields requiring adaptive and socially intelligent AI. In these games, players tend to enjoy and engage more deeply with AI opponents that not only provide a challenge but also behave in ways that resemble human thinking and decision-making. However, despite progress in developing such agents, there is still no standard approach for evaluating how human-like these opponents truly are—making it difficult to assess and improve their design. Here I show that strategy game opponents having more human-like game-level playstyles does not necessarily lead to them being more believable (perceived as human-like by human players).**

By developing a turn-based strategy game and evaluating Hierarchical Reinforcement Learning (HRL) agents of varying complexity, I assessed both their behavioural similarity to human players and how believable they were perceived to be by human players. This research introduces a new approach for understanding player behaviour using behaviour vectors composed of three high-level metrics—Aggressiveness, Management, and Exploration—consistent with existing literature. These metrics are designed to be broadly applicable across strategy games, enabling consistent comparison between human and AI opponents, as well as across different games and agents. The findings demonstrate that while HRL agents can replicate human-like playstyles without using human training data, players judge human-likeness more on perceived intelligence and fairness. This suggests that creating truly human-like AI opponents requires not just replicating human game-level playstyles, but designing agents that align with players' expectations for intelligent and fair decision-making.

I Introduction

Research into creating human-like computer agents in strategy games (e.g. StarCraft II, Sid Meier's Civilization, Chess) aims to enhance player engagement while also advancing artificial intelligence by developing techniques applicable to other domains that require human-like decision-making and adaptability. By understanding and replicating human behavior in virtual environments, this line of research can provide valuable insights into human psychology and social interactions. The findings not only contribute to the gaming industry but also have broader implications for fields that benefit from human-like AI interactions.

Playing a strategy game against a human opponent is often more enjoyable than facing an AI opponent. In strategy video games, AI opponents are often used as substitutes for human opponents, who may not always be available. Players find the experience more engaging when an AI opponent is not only competitive but also exhibits a wide range of human behaviors [1, 2]. The ability of these AI opponents to display human-like behaviors allows players to empathize with them [3], enhancing the overall engagement [4]. Therefore, an ideal strategy game opponent should be both competent enough to provide a challenge and human-like enough for players to understand and relate to the opponent's perspective.

Although significant effort has gone into developing AI opponents that appear human-like, there remains a lack of consistent and reliable methods to evaluate whether these agents actually exhibit human-like behavior. Without a proper understanding of human behaviour, it is unclear whether the advances in architecture and design truly contribute to more human-like agents. A well-defined evaluation framework would help future researchers with iterating their agents' design for increased human-likeness. A consistent method of evaluation will also make it easier to compare different agents across different games. In order to make such an evaluation framework, the following questions have to be answered.

How should player behaviour be quantified and validated to guide the design of human-like agents in strategy games?

1) ***How should a strategy game and an Hierarchical Reinforcement Learning (HRL) agent opponent in that game be designed to allow for human-like behaviour to be easily observed?***

2) ***How can player behaviour in a strategy game be quantified to allow for comparison between human and AI opponents?***

3) ***How does the degree of human-likeness and perceived opponent traits affect believability in an AI opponent and how does the believability affect player engagement?***

I attempted to solve these questions in this research by:

- Developing a simplified strategy game, that is easy to learn and allows for diverse strategies, with two types of AI opponents (based on Behaviour Trees and Hierarchical Reinforcement Learning).
- Developing a quantitative metric for the game-level

playstyle of a player, consistent with existing literature on human playstyles, for measuring the difference in behaviour between two populations, in this case humans and bots.
- Collecting human play data and making participants evaluate their opponents with surveys to find how the game-level playstyle and qualitative metrics influence the believability of an agent, and how that believability affects engagement.

A. Related Work

To support the design choices and evaluation methods used in this research, this section reviews previous research on how human-likeness is evaluated, how believable agents are made, and how player engagement is measured. It covers existing approaches for evaluating human-likeness in agents, including Turing tests, action-level comparisons, and behaviour metrics. It also discusses common agent architectures like Behaviour Trees and Hierarchical Reinforcement Learning, and examines how engagement and opponent perception are measured.

1) Evaluating Human-Likeness

Most papers that create agents for games do not evaluate the human-likeness of the created agents and instead only evaluate the agents on their task performance (e.g. win rate), which is sometimes compared to the task performance of humans, for instance in [5].

A common approach to measuring human-likeness is move match accuracy, which compares the decisions made by humans and agents in the same game states. This method requires a substantial number of humans and agents to have encountered and acted from the same states. While this is feasible in simple games with a limited number of distinct states, it becomes particularly challenging in complex games, where the vast action and state spaces drastically reduce the likelihood of overlap between human and agent decisions. As a result, there are often too few shared states to perform meaningful comparisons. An exception to this is Chess (and Go), where the abundance of human gameplay data means that even positions several moves into a game may have been encountered thousands of times. This makes such games uniquely well-suited for evaluating the human-likeness of agents using move match accuracy, as demonstrated in papers like [6] and [7].

Aside from the enormous amount of human data required in complex (strategy) games for this approach, this approach evaluates decisions made on the lowest level (action-level) and thus ignores higher level (strategic) decisions/behaviours. Comparing decisions on the action-level could result in false negatives, because two actions although different could belong to the same strategy. Also, it could result in false positives, when two actions although the same belong to different strategies.

A different approach is to calculate metrics over the whole game and compare the metrics of humans with those of the agent. These behaviour metrics when concatenated into a vector will be referred to as a behaviour vector. This approach of comparing these behaviour metrics or behaviour vectors seems to be quite common when evaluating the human-likeness of AI opponents. I summarized existing research using behaviour vectors for evaluating the human-likeness of AI opponents in Table 1). This table shows that there is no consistent naming for the behaviour metrics, no consistent comparison distance metric and no consistent visualization method. The only thing that a few of the papers do share, is calling clusters of behaviour vectors playstyles.

Paper	Name for the scalar values	Comparison method	Visualization method
[8]	Features	Predictive model	N/A
[9]	Play logs	K-means clustering	PCA plot
[10]	Phenotypes	N/A	Heatmap
[11]	Metrics	KL and JS divergence	PCA plot (only two metrics)
[12]	Statistics	Euclidean distance	N/A
[13]	Summary statistics	Visual	Each metric separately vs. turn number
[14]	N/A (Referred to directly by name)	L1 distance	N/A

TABLE I
COMPARISON OF METHODS USED IN LITERATURE THAT ANALYZE THE HUMAN-LIKENESS OF AGENTS BASED ON SCALAR VALUES THAT SUMMARIZE PLAYER BEHAVIOUR

Alternatively, AI opponents can be compared qualitatively using Turing tests. A common method for comparing agents with humans is through Turing tests. For instance in [15] participants compare human gameplay with gameplay from agents by looking at side-by-side videos. A limitation of this approach is that iterating the design of the agent based on these results is costly, because it requires participants for each comparison. Another limitation is that the believability found using the Turing tests is not transparant, it is not clear what part of the behaviour is responsible for the participant's decision to judge a player as human-like or not human-like. However, believability does offer insight into how closely an AI opponent's behaviour aligns with human expectations and perceptions of natural gameplay. Therefore in this research, believability is used to validate the playstyle profiling used to determine human-likeness.

2) Agent Architecture

In order to make a proper comparison between the human participants and the agents, the agent should be capable of solving the game with some degree of success. Furthermore, the agent should give the participant some challenge as an opponent, to be able to see diverse human behaviour. Currently, games often use simplistic Behaviour Tree [16] or Finite State Machine [17] agents due to development time and computational constraints. Finite State Machines "scale poorly and are difficult to extend, adapt and reuse" [18]. Behaviour trees are more adaptable, but implementing certain behaviours, such as those that respond to external events (like the player), can be cumbersome [19].

Alternatively, computational cognitive models can be utilized, whose main goal often is to accurately model human cognition [20]. Therefore computational cognitive models are more likely to provide more human-like behaviour than Behaviour Trees and Finite State Machines, which ”describe predefined missions with limited decision-making” [21]. Furthermore, computational cognitive models describe the underlying processes of cognition [20], instead of explicitly programming behaviours like in Behaviour Trees and Finite State Machines [19].

From the many potential computational cognitive model architectures, I chose the Hierarchical Reinforcement Learning (HRL) architecture, since it has a high potential to create a competitive and human-like agent.

3) Engagement evaluation

A human opponent is more fun to play against, but does this also make a human-like AI agent more fun to play against? Measuring the level of engagement can be done with surveys to directly ask participants how fun/engaging they found the game to be. A study [22] surveyed players by both asking whether they would like to continue playing and asking them to pick cards with specific words on them to see why they were engaged. However, the evaluation of engagement and player perception of their opponent alongside a quantitative human-likeness analysis is underrepresented in the research into creating human-like opponents.

II Methodology

A. Game Features

The first objective of this research is to create a strategy game that allows for human-like behaviour to be easily observed. Firstly, why is a strategy game specifically suited for this? Many game genres exist, but strategy games are among the most cognitively demanding, requiring players to make both short-term tactical decisions and long-term strategic plans [23], making them a good testing ground for analyzing human decision-making. Strategy games for research can differ in a range of mechanics, such as cooperative [24] or competitive [25], real-time [26] or turn-based gameplay [27], continuous [5] or tile-based maps [28], and varying numbers of opponents, among other design differences. In this research, a competitive two-player turn-based tile-based strategy game was used, to reduce the complexity for both the player and the AI opponent.

A strategy game that allows for human-like behaviour to be easily observed in both the human and AI players has several requirements. The game should be easy to learn quickly, as the participants will only play a few games and will have varying levels of experience with strategy games. The game should also enable varied behaviour to ensures a fair comparison between the AI and the player. In order to meet these requirements, I made a game specifically for this research, a screenshot for which can be seen in Figure 1.

When designing the game, it’s important to balance diverse player behaviour with ease of learning. Complex mechanics can provide many strategies but may be hard to learn quickly. Simple mechanics, familiar from other strategy games, can be easy to understand and still offer multiple possible strategies. If the participants can learn the game quickly, they are more likely to start playing the actual game, and they will have more time for the actual experiment. Even if there are many possible strategies, they should be balanced such that no single strategy is too strong (degenerate strategy). If one strategy is clearly better, both the AI and the player are likely to always use it, which would undermine the validity of the comparison.

The primary objective of the game is to destroy the opponent’s tower. Victory is achieved by being the first to destroy the enemy’s tower. If neither tower is destroyed within 45 turns, the player with the tower having the most health is declared the winner, with player 2 winning in the event of a tie.

At the beginning of a game, players can select from four distinct units to recruit, each with unique costs, strengths, and weaknesses. The available units are the pikeman, archer, knight, and battering ram. These units are part of a combat triangle: pikemen are strong against knights, knights are strong against archers, and archers are strong against pikemen. Additionally, all units are effective against the battering ram, which in turn is particularly effective against towers. Players can have a maximum of five units on the board simultaneously. Units require gold for recruitment, which players earn each turn. Additional gold can be acquired by controlling gold mines.

The towers are positioned on the left and right sides of the board, with recruited units appearing in the player’s training yards located behind the towers. The path between the two towers is divided by a mountain range, creating two separate routes to the towers. There is a gold collection point located at both the top and bottom of the board.

B. Game Agent Design

For this research, three different agents were created to represent different approaches to the design of AI agents. The first agent is used as a baseline and uses a behaviour tree like structure to make decisions to be representative of how AI opponents are traditionally made in games. The second agent is an Hierarchical Reinforcement Learning (HRL) agent with three layers. The third agent is an HRL agent with two layers.

In order to provide a more engaging experience I created variants for each AI opponent type. The Baseline bot uses a different seed in each game resulting in its attacks being timed slightly differently and it using a different composition of units. The HRL agents each use a different agent for each

Fig. 1. A screenshot of the game

experiment game. These agents are trained in the same way, but since the training process does not always converge to the same solution, agents with different playstyles can be created. The order of the different versions of the AI opponents were chosen such that they are similar in behaviour, for instance the variant participants play against in the first game is an aggressive version for all three AI opponents. Since the variants differ in the same way for the three AI opponents, the AI opponents can be compared as a whole, without needing to compare the individual variants with each other.

1) Baseline Bot

The Baseline Bot uses a fixed set of rules to make decisions. Newly recruited units start with the defensive strategy. The strategy of all units changes every 5-10 turns to a random new strategy that is different from the old one. The potential strategies are the same as for the HRL agent: defend, attack, control the top gold collection point and control the bottom gold collection point. Each turn each unit tries to move towards their assigned strategy center point. However, if a unit is close to an enemy unit or the enemy tower, it will attack the enemy unit or tower that it will deal the most damage to.

2) HRL Agent with Three Layers

Hierarchical Reinforcement Learning (HRL) extends Reinforcement Learning (RL) and uses temporal and spatial abstractions to allow for more efficient training in complex environments [29]. HRL has been shown to potentially be a good model of human decision-making since humans also use an hierarchical approach to break down complex problems into smaller problems they have solved before [30, 31]. HRL has been used to solve complex (strategy) games like a version of the game OverCooked [24], a MOBA [5], StarCraft [32] and Montezuma's Revenge [33] to name a few.

An RL agent optimizes a single policy to select primitive actions, whereas HRL agents optimize multiple policies, each corresponding to a different level of abstraction [31]. For the terminology in this paper, the HRL agent is split into multiple layers, where each layer can then also contain multiple agents, one for each unit the player has. Since there are multiple agents in each layer, this is a Multi Agent HRL (MAHRL) agent [34]. The layers are organized in a hierarchy, where the higher level agents set objectives for the lower level agents. In this research, some of the layers are more loosely coupled than is typical for HRL.

HRL Adjustments In order to make the HRL agent perform well on the game made for this research, several adjustments were made to the typical HRL architecture. Firstly the abstractions that were made by each layer are manually set. The abstractions made in each layer of the HRL agent can be seen in Table II. Where the power level reward is defined by equation 1.

$$\text{power level} = \sum_{i=0}^{n} \text{unit gold cost}_i \frac{\text{unit health}_i}{\text{unit max health}_i} + \text{player gold} + 50 \frac{\text{player tower health} - \text{opponent tower health}}{\text{tower max health}} \quad (1)$$

Layer	Number of agents	Responsibility	Observation	Reward
0	one for each unit	unit strategies	unit health and type, distances to strategy center points	power level
1	1 when a unit can be recruited	unit recruitment	enemy unit types and own strategy distribution	power level
2	one for each unit	unit targets	current strategy and target type, distance and effectiveness	damage to target - damage to self

TABLE II
ABSTRACTIONS MADE FOR EACH LAYER OF THE HRL AGENT

A standard HRL agent has higher level agents make decisions less frequently, so that the lower level agents can follow through on the set objective. However, in this research, each layer related to a unit decides a new action every turn. Except for the layer that recruits the units, since this layer decides a new action every time a unit can be recruited, so this can be multiple times in a single turn, but also not at all in a certain turn when there are not enough resources to recruit a unit, or the maximum number of units is already recruited.

By having the higher level agents rethink their objective every turn, the agent is more responsive to the opponents actions. To prevent the higher layers from adding too much noise in their decision making resulting in a lower level layer not being able to follow through on the set objective since it changes too quickly, the top layers only change their objective when the confidence level for the new objective is significantly higher than the confidence level for the old objective.

HRL Training Process The HRL agent is trained against the Baseline bot using the Proximal Policy Optimization (PPO) algorithm [35]. The HRL agent was implemented in Python using the Ray library [36]. The learning rate was set at 1E^{-4}, the batch size was set at 256 and the neural network for each layer consisted of two hidden layers of 128 nodes.

Each layer of the HRL agent needs the other layers to work well in order to perform well. To start up the training process, the layers are first trained bottom up on scenarios. In these scenarios, the decisions made by the layers above the currently trained layer are fixed to a randomized action. In the scenarios for the layer 2 training, the game starts with a random number of units at random positions on the board. For the layer 1 and 0 training, the game starts as it normally would (with no units on the board).

After the bottom up training using the scenarios, the layers are not necessarily aligned with each other. The bottom layers have been trained on random scenarios, but have not been trained on the actual situations that the higher layers will put them in. So for the second training phase, these layers are trained again bottom up, but without using scenarios, playing the actual game.

3) HRL Agent with Two Layers

This agent is similar to the HRL agent with three layers, but with the top layer removed. This agent is included to investigate the impact of the number of layers in the agent architecture on the human-likeness.

Since the top layer (layer 0) is removed, layer 2 (Table II) now can choose between targets for all possible strategies, instead of just the targets for the strategy set by layer 0. This means that the agent is still capable of making the same exact decisions, but the action space of two layers is combined into one.

This agent uses the same training process as the one for the HRL agent with three layers, but with the top layer removed. The agent is trained bottom up on scenarios and then trained on the actual game.

C. Human Behaviour and Engagement Analysis

Similar to [9, 10, 11, 13, 14], I use gameplay summary statistics (behaviour metrics) in order to compare the playstyle of humans and AI players. In contrast with these papers, the behaviour metrics I chose are consistent with literature [37]. I then use these metrics to find differences in playstyle instead of categorizing gameplay into predefined task-specific playstyles. Furthermore, I also validated the chosen submetrics for the behaviour metrics by using the fact that humans have individually consistent behaviour. Finally, I explain what questions participants were asked to determine player engagement, AI opponent traits and believability.

1) Behaviour Vector Construction

As stated in the introduction, one goal of the human-likeness evaluation is to make it easier to compare agents across different games. To facilitate this, the chosen behaviour metrics were drawn from literature, which states that the player types most likely to play strategy games are characterized by the behaviour metrics of aggressiveness, management, and exploration [37]. Where aggressiveness measures a player's tendency to attack and pursue offensive strategies, management measures a player's focus on defensive and resource management strategies and exploration measures a player's time spent on exploring the game map.

The aggressiveness, management and exploration metrics are used to construct a behaviour vector, which is representative of the game-level playstyle of the player. These metrics are made by combining several submetrics that each summarize a different part of the gameplay. The composition of the behaviour metrics into submetrics is described in table III and the way these are calculated is explained in the next section.

Aggressiveness	
front_movement	Proportion of unit movements in the forward direction (-30° to 30°)
fraction_attacking	Proportion of turns with at least one unit has the attack strategy
1 - avg_turns_between_attacks	Average number of turns between attack periods
Management	
back_movement	Proportion of movements in the backward direction (150° to 210°)
idle_time	Proportion of turns with no movement
n_units_defending	Average number of units with the defend strategy per turn
Exploration	
side_movement	Proportion of movements in lateral directions (all other angles)
n_unique_visited_tiles	Count of distinct tiles visited by a player's units
n_units_gcp	Average number of units with GCP-related strategies per turn

TABLE III
BEHAVIOUR METRIC COMPOSITION FROM SUBMETRICS THAT SUMMARIZE GAMEPLAY ELEMENTS

2) Submetric Calculation

The calculation of these submetrics involves several data processing steps:

Unit Path and Strategy Assignment Unit strategies are inferred from their positions on the map:

1) Each unit's full movement path is recorded from creation to destruction.
2) The map is divided into defend (own Tower/Yard), attack (enemy Tower), and GCP (top/bottom gold points) zones.
3) Each turn, units are assigned a strategy based on their current zone. If outside all zones, they retain their previous strategy.

Afterwards, for each turn, the number of units following each strategy are summed up to get the following metrics:

- `n_units_defending`: Average number of units per turn in defend zones
- `n_units_attacking`: Average number of units per turn in attack zones (used for the calculation of `fraction_attacking` and `avg_turns_between_attacks`)
- `n_units_gcp`: Average number of units per turn in GCP zones

Movement Direction Analysis The movement direction metrics track how units move across the game map:

1) For each unit movement, the angle is calculated between the previous and current position
2) Movements are categorized and counted into the metrics shown below.

- `front_movement`: Proportion of movements in the forward direction (-30° to 30°)
- `back_movement`: Proportion of movements in the backward direction (150° to 210°)
- `side_movement`: Proportion of movements in lateral directions (all other angles)
- `idle_time`: Proportion of turns with no movement

Spatial and Attack Pattern Analysis Additional metrics capture spatial exploration and attack patterns:

- `n_unique_visited_tiles`: Count of distinct tiles visited by a player's units
- `fraction_attacking`: Proportion of turns with at least one unit has the attack strategy
- `avg_turns_between_attacks`: Average number of turns between attack periods

3) Normalization Process

The behaviour vector computation involves a multi-step normalization process:

1) Individual submetric normalization: Each submetric is normalized to a [0,1] range based on the observed distribution in human player data. For inverted metrics (like `avg_turns_between_attacks` where lower values indicate higher aggressiveness), the normalized value is subtracted from 1.
2) Composite metric calculation: Each primary metric is calculated as the average of its normalized submetrics.
3) Final normalization: The resulting behaviour vector is normalized so that the sum of its elements equals 1, ensuring that all vectors are directly comparable in terms of the relative proportions of behaviours.

4) Vector Comparison Methods

Several methods were employed to compare behaviour vectors across game sessions and between players. These include similarity metrics, consistency measures, and multivariate distance calculations. Before further analysis, first for each game played by the participants of the experiment a behaviour vector was calculated. How the participants were recruited and what there demographics are is described in section 7).

a) Cosine Similarity

Cosine similarity (equation 2) was used as the primary method to compare individual behaviour vectors.

$$\text{cosine\_similarity}(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|} \tag{2}$$

This metric measures the angle between two vectors and is unaffected by their magnitudes. Since behaviour vectors may differ slightly in scale even after normalization, cosine similarity emphasizes the relative proportions of behaviour components rather than their absolute values. It is used instead of the Euclidean distance in for instance the standard deviation calculation shown in the next paragraph.

b) Behavioural Consistency

Players have neither perfectly consistent, nor perfectly random behaviour, some will change the way they play, while others keep playing in the same way [38]. Using this insight, the previously described method of determining playstyles through the behaviour vectors can be validated with human data by comparing the consistency of the human behaviour vectors throughout the game with random behaviour vectors. To evaluate the consistency of behaviour across multiple games for each participant, a behavioural consistency score was defined, which was not taken from prior research. It quantifies how stable a participant's behaviour is over time relative to the full group of participants.

Let each behaviour vector have d dimensions. Given a set of N behaviour vectors $\{\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_N\}$, the standard deviation of behaviour across those vectors is calculated as:

$$\sigma = \frac{1}{d} \sum_{i=1}^{d} \text{std}\left(v_1^i, v_2^i, \ldots, v_N^i\right) \tag{3}$$

This formula is applied once for each player to compute σ_{player} using all behaviour vectors of a single participant, and once to compute σ_{group} using the behaviour vectors from all participants.

The behavioural consistency score for a certain player is then defined as:

$$\text{consistency} = 1 - \frac{\sigma_{\text{player}}}{\sigma_{\text{group}}} \tag{4}$$

A higher score indicates that the participant's behaviour is more consistent.

c) Behaviour Group Differences

To compare sets of behaviour vectors across different groups, the Mahalanobis distance (d_M) [39] was used, shown in equation 5. This multivariate distance metric accounts for correlations between behaviour dimensions and provides a scale-invariant measure of dissimilarity between distributions.

$$d_M = \sqrt{(\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu})} \tag{5}$$

This distance was used to compare the distribution of behaviour vectors from the human participant group with those from each of the three AI opponents, when they were playing against the Baseline bot. For each group, a mean behaviour vector $\vec{\mu}$ and a covariance matrix Σ were computed, and the Mahalanobis distance between group distributions was calculated to determine how similar the playstyles of the AI opponents were to those of the participants.

5) Responsiveness and Challenge Metrics

The previously described behaviour vector defines the game-level playstyle used by the participant or AI opponent in a single game. However, in order to reflect a part of the lower-level behaviour, the responsiveness metric can be used, seen in equation 6.

$$\text{responsiveness} = \frac{\text{\# actual attacks}}{\text{\# potential attacks}} \tag{6}$$

Where the number of actual attacks is the number of attacks on enemy units and the number of potential attacks is the maximum number of attacks on enemy units that was possible. The number of actual attacks and potential attacks are determined for each turn and then added up for a whole game, resulting in a responsiveness value for the whole game. This metric is interesting for analyzing behaviour because it measures part of how the player interacts with their opponent, which is not measured by the behaviour vector.

Another element that is not measured with the behaviour vector is how well the players in the game are doing. The degree of challenge is dividing the final power level by the number of turns in the game, seen in equation 7.

$$\text{challenge} = -\frac{P_T}{T} \tag{7}$$

Where T is the total number of turns, P_T is the player's power level (equation 1) at the final turn T. If the player's power level at the end of the game is positive, the player has won and so the challenge value will be negative. The faster the player has won and the larger the margin of winning is, the lower the challenge value. If the player has lost, the challenge value will be positive. The faster the player has lost and the larger the margin of losing is, the higher the challenge value.

6) Believability, Opponent Traits and Engagement Evaluation

Current research evaluates believability with either a first or third person assessment, which is either binary or uses a scale [40]. In this research, I opted for a first person assessment where the player rates the believability of their opponent using a Likert-scale. An extensive review of the research using Likert-scale for measuring the believability of virtual agents is performed in [41]. The first person assessment is more representative of the end goal, making players believe they are playing against a human. The Likert-scale allows for the degree of believability to be measured for individual games, whereas the binary assessment needs to be averaged over multiple games to get a degree of believability for a certain

AI opponent. The degree of believability is used to validate that the previously described behaviour vectors capture a part of human behaviour that is noticeable to humans themselves.

In order to evaluate the engagement of the participants and where that engagement comes from, a survey was used. The survey was designed to be answered after each game, so that the change in engagement over time could be observed. The survey consisted of the following statements, which the participants were asked to rate on a 7-point Likert scale from 1 (strongly disagree) to 7 (strongly agree):

- "I felt fully into the game and lost track of time"
- "I played naturally without overthinking"
- "The game made me feel excited or angry"
- "The opponent played in a human-like way"
- "The opponent played in a predictable way"
- "The opponent was challenging to play against"
- "The opponent's moves felt smart and planned"
- "The opponent changed its strategy based on how I played"
- "The opponent played in a fair way"

To compare the different AI opponents, the differences between the survey responses were analyzed using a mixed effects model, shown in equation 8.

$$\text{SurveyQuestion}_i \sim \text{Opponent} + (1|\text{Participant\_id}) \quad (8)$$

In this model, SurveyQuestion_i represents the Likert-scale response to the i-th survey statement. The variable Opponent is a fixed effect indicating which AI opponent the participant faced in that game (Baseline, HRL-2-layer, or HRL-3-layer). The term $(1|\text{Participant\_id})$ specifies a random intercept for each participant, which takes into account variability in measurements for each participant. This mixed effects approach allows the effects of different AI opponents to be assessed while accounting for individual differences in survey responses.

7) Experiment Setup

Participants were recruited through social media with a small description of the experiment and a link to the game, which could be played in a browser. Participants were offered no form of remuneration. When the participant first opens the game, they are shown an interactive tutorial to teach them the previously described mechanics. The tutorial should take less than 5 minutes to complete, so they can quickly start with the actual experiment. After completing the tutorial, participants were asked to agree to the consent form and provide their age, gender and game playing experience, which was evaluated on a Likert scale from 1 to 7. All data was collected through Google Firebase Realtime Database, allowing both for easy data collection and the ability to host the AI opponent on a server. This offloads the computation from the client and is convenient because the AI opponent was written in Python, whereas the game was made in Godot.

Each participant played 3 games, during which the moves of both the participant and the AI opponent were recorded. After completing each game, participants were required to fill out a survey (section 6). The participants could face one of three different AI opponents: the Baseline bot, the HRL agent with three layers, or the HRL agent without the top layer (so two layers). Each participant was randomly assigned to face the same AI opponent for all 3 games.

In total, around 150 people opened the game, 54 people started the first game, 45 people completed the first game and 27 people completed the full 3 games. Of the 45 participants that completed the first game, 30 were men, 12 were women and 3 did not report their gender. The participants were between 18 and 68 years old, with a mean age of 35 years and a standard deviation of 12.6. The average self-reported game playing experience was 3.7 out of 7, with a standard deviation of 1.6.

All analyses in this study include only data from participants who completed all three games in the experiment, this was necessary to make fair comparisons between groups. Each participant played three games against the same AI opponent type, but the AI opponent's behaviour varied in a consistent way from game to game. For instance, participants always played the first game against an aggressive variant of their assigned AI opponent type. If participants who dropped out early were included, certain AI opponent types might be overrepresented in specific game stages—such as the first, second, or third game. This would introduce bias, as comparisons between opponent types would reflect uneven exposure to their varying behaviors rather than genuine differences in player responses.

Ethical approval for this study was granted by the Human Research Ethics Committee (HREC) at Delft University of Technology (Application Number 4774).

III Results

A. Game and Agent Design

In order to determine whether the HRL agent is capable of learning to play the game, the win rate against the Baseline bot is plotted against the iteration number (Figure 2), showing that the HRL agent eventually learns to play the game well, despite some setbacks. In Figure 2 the training results for the three layered HRL agent that is used in the first game of the experiment can be seen. The win rate is used instead of the policy rewards, because the rewards for the layers differ.

The total training time for this agent was 1 hour and 27 minutes, which is quite fast, considering it was trained on a mid-range processor (AMD Ryzen 7 5800X). However, I had to train dozens of agents to get the 3 viable agents for

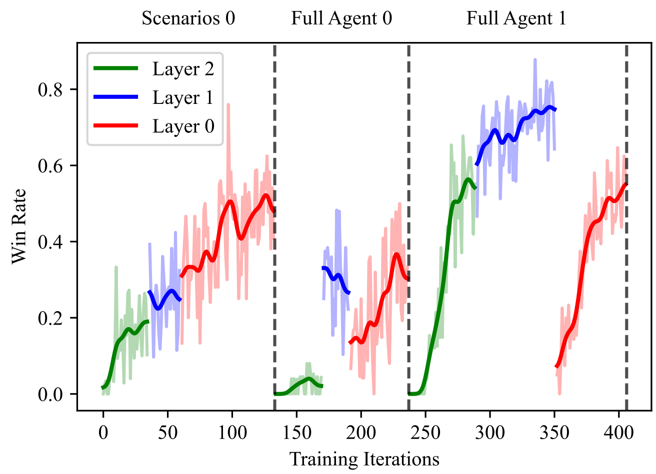

Fig. 2. HRL-3-layer first game variant training results

HRL-2-layer and 3 viable agents for HRL-3-layer, since the training process does not always converge to an agent that will be competitive against human players. The HRL agents each have a different variant for each experiment game, to match the Baseline bot, which uses a different seed for each game.

Using the agent without using the last layer 0 training step would have resulted in a better agent in hindsight.

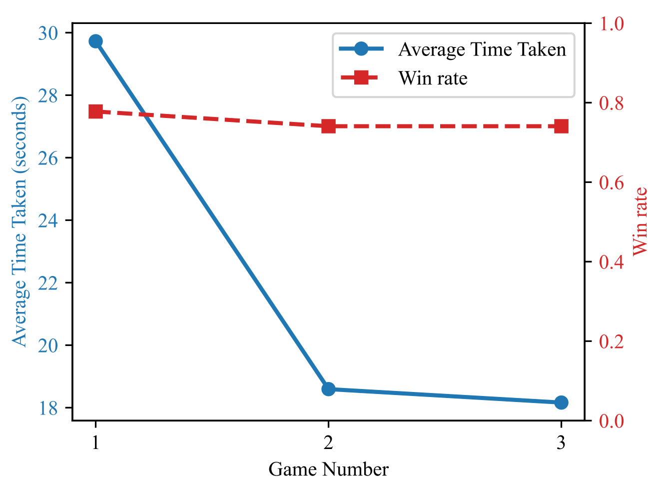

Fig. 3. Player learning across the three experiment games

In order to determine that participants learn to play the game well, the average time taken per turn and win rate was compared for each experiment game (Figure 3), showing that players quickly learn to play the game. The initial sharp decline in average time taken per turn after the first game and the stabilization of it from the second game to the third and last game suggests that the players still had to spend a lot of time going through all the options they have each turn in the first game. Whereas in the other games, the player quickly knows what they can do according to the game rules and spend most of their time moving their units and devising their strategy.

In order to validate the chosen behaviour metrics, the participants' behavioural consistency was compared to that of a set of random behaviour vectors, showing that the behaviour vector is capable of capturing enough of the behaviour that behavioural consistency is observed. The random set of behaviour vectors was generated by setting each metric in each behaviour vector to a random value between 0 and 1. This random set of behaviour vectors had an average consistency of 0.28, whereas the participants' behaviour vector set had an average consistency of 0.45, with a 95% confidence interval of [0.34-0.55]. This shows that the behaviour metrics are free enough of noise. It however does not prove that the behaviour metrics encompass all of human behaviour variation.

It is possible that the player learning in the first game (Figure 3) increases the behavioural diversity and thus increases the consistency value, but in spite of that, the participants are still more consistent than the random case. The random case has its consistency value significantly above 0 (0.28), this is a result of there only being 3 games. For instance if the analysis was done for 5 games, the consistency for the random case would be 0.16. So for 3 games it is more difficult for consistent behaviour to be significantly below the random baseline

Now that the behaviour vector methodology is validated to capture a part of human behaviour, it can be used to compare the behaviour of the three different agents the participants played against.

B. Quantitative Human-likeness Analysis

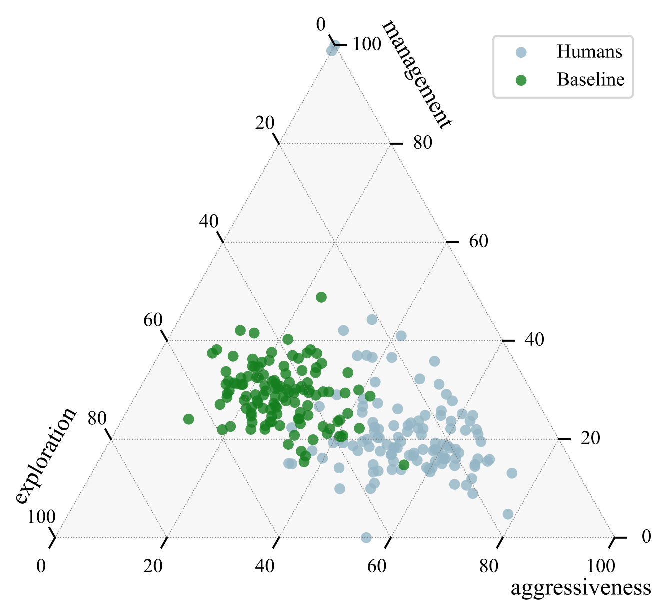

Fig. 4. Ternary plot comparing human participants with the Baseline bot

In order to determine whether the Baseline bot or the HRL agent variants show more human-like behaviour, their

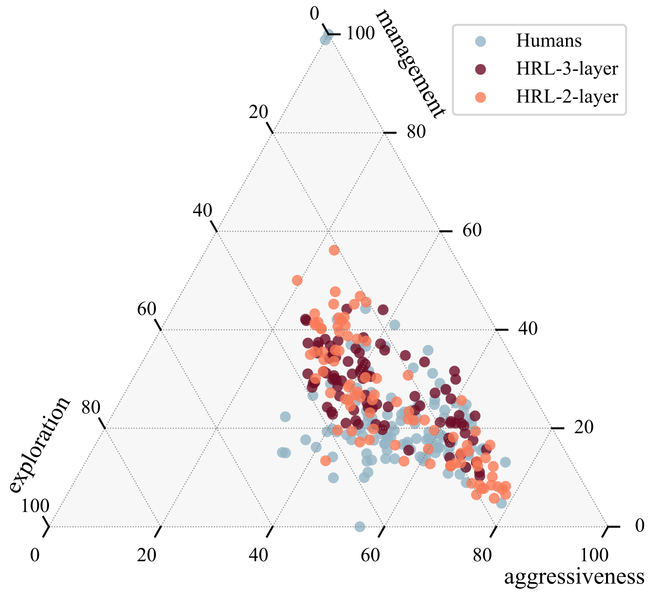

Fig. 5. Ternary plot comparing human participants with the HRL-2-layer agent and the HRL-3-layer agent

	Baseline	HRL-2-layer	HRL-3-layer
Mahalanobis distance to human data	3.001	0.635	0.915

TABLE IV
MAHALANOBIS DISTANCE BETWEEN AI OPPONENT BEHAVIOUR VECTOR DISTRIBUTIONS

playstyle distributions were visually compared in Figures 4, 5 and quantitatively compared in Table IV using the Mahalanobis metric, showing that the HRL agents follow the human behaviour distribution more closely than the Baseline bot does.

The metrics in the behaviour vector are summarized across the whole game, so they are only capable of showing game-level behaviours, like being more aggressive. Now I will look at two other quantitative metrics, an action-level metric for how the players respond to their opponents and a metric for how challenging the game was, which are compared for the three AI opponents in Table V.

The responsiveness and challenge of each bot is recorded when it was playing against humans, with the standard deviation after the average value. The Baseline and HRL-3-layer bots are both statistically significantly more responsive than the participants ($p < 0.001$), whereas the HRL-2-layer is not. The Baseline bot is programmed to attack each time it can, so it having the highest responsiveness is natural. The difference in responsiveness of the HRL agents can be explained by the reduction in potential targets that the three layered HRL agent has due to the additional level of abstraction. The degree of challenge (quantitative) experienced by the participants was not significantly different between the three different AI opponents.

	Humans	Baseline	HRL-2-layer	HRL-3-layer
Responsiveness	0.78 ± 0.13	0.91 ± 0.06	0.78 ± 0.14	0.87 ± 0.07
Challenge	-	-0.44 ± 1.04	-0.62 ± 0.53	-0.29 ± 0.93

TABLE V
RESPONSIVENESS AND CHALLENGE QUANTITATIVE METRICS

C. Qualitative Believability Analysis

The answers to the survey questions were compared between the AI opponents using the Mixed Effects model described in equation 8. The statistically significant differences between opponents are as follows: the HRL-2-layer agent is perceived as more predictable than the HRL-3-layer agent ($p<0.05$), with the average values for perceived predictability being 4.7 and 3.6 out of 7 for the HRL-2-layer and HRL-3-layer agent, respectively. The HRL-2-layer agent is perceived as more fair than the Baseline bot ($p<0.05$), with the average values for perceived fairness being 4.1 and 5.4 out of 7 for the Baseline and HRL-2-layer agent, respectively. The answers to the predictable, challenge, strategic and adaptive questions, where predictability is first transformed to 7-predictability, can be combined into one intelligence value, since they are strongly correlated with Pearson correlation coefficient values between 0.5 and 0.8. When this perceived intelligence value is compared for both HRL agents, the HRL-3-layer and HRL-2-layer agents have average values of 4.1 and 3.3 out of 7 respectively. This difference is not statistically significant with $p = 0.1$. The differences between HRL agents on the predictable, challenge, strategic and adaptive questions are always in favour of the three layered HRL agent, but also not statistically significant.

In order to determine whether either HRL agent is more engaging to play against than the Baseline bot, the average of the answers to the engagement questions were compared (the 3 left most questions in Figure 6), showing there is no statistically significant difference between the three AI opponents in how engaged the participants were.

One problem of determining the engagement this way is that there is a large selection bias for participants that are already engaged by the game. The data used for this comparison includes only participants that have completed at least one game. In order to reduce the selection bias, I used the first game completion rates between the three AI opponents (Figure 7) as a measure of engagement, which means all participants that have completed the tutorial are included, to show that there is still not a significant difference in engagement between the players that played against the Baseline bot and either of the HRL agents. The hypothesis that the first game completion rate of the combined group of players that played against either of the HRL agents is greater than that of the group of players that played against the Baseline bot is not quite significant at

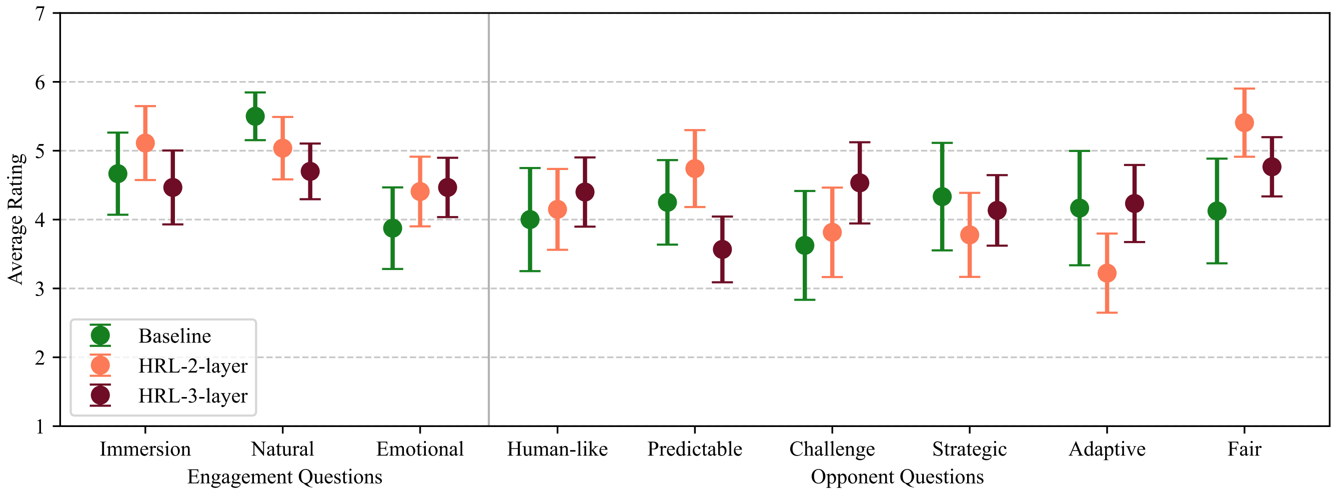

Fig. 6. Survey results by opponent with 95% confidence intervals

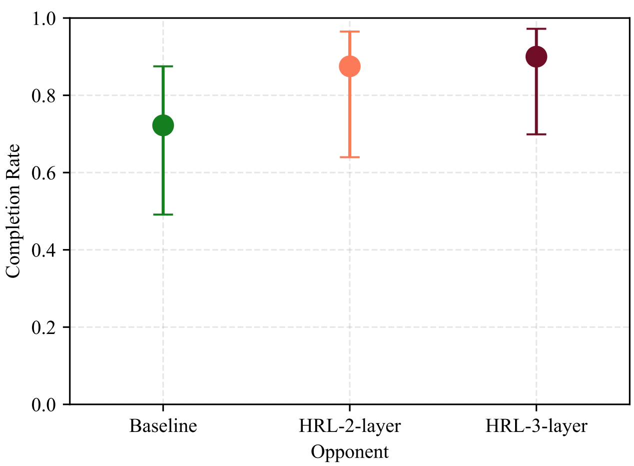

Fig. 7. Experiment completion rates by opponent with 95% confidence intervals

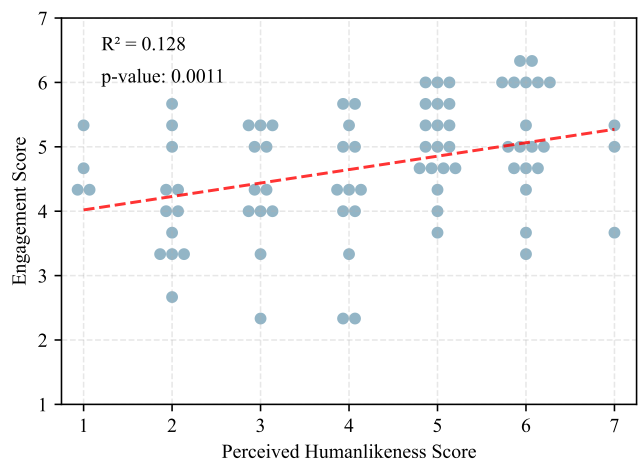

Fig. 8. Engagement versus perceived human-likeness (believability)

p=0.06, even though the first game completion rate combined for both HRL agents is 0.89 and 0.72 for the Baseline bot.

In order to determine whether a more believable agent leads to a more engaging experience, a linear regression was done between the surveyed believability and the average of the three surveyed engagement questions (Figure 8), showing a small, but significant correlation between believability and engagement. Although there is a statistically significant (p $<$0.01) correlation between the believability and engagement, this correlation is weak given the R^2 value of 0.128. This weak correlation suggests that other factors are more important for engagement.

I could not make any strong observations on the engagement of the participants using the survey question answers on believability. Observing a part of the participants playing the game, I saw that the most significant blockers for engagement were not understanding the rules of the game and not having a preference for the strategy game genre. These things could be mitigated by letting only experienced players play the game, but that would reduce the pool potential of participants significantly.

In order to determine to what effect the perceived opponent traits (strategic, challenge, adaptiveness, predictability and fairness) correlate with believability, a multiple linear regression was performed, showing that there is a correlation (adjusted R^2 = 0.33) between them. First the answers for the questions to the strategic, challenge, predictable and adaptive were averaged into combined intelligence evaluation since they were all significantly correlated with Pearson coefficient values ranging from 0.5 to 0.8, resulting in multicollinearity during the multiple linear regression if they were not combined. The fairness value was kept separate because it was not correlated with the values making up the intelligence value. This intelligence value combined with the value from the fairness question was used to predict the believability of the AI opponents (answer to the human-likeness question) using multiple linear regression. This resulted in a moderate correlation with an adjusted R^2 of 0.33 ($p < 0.001$) and the coefficients for intelligence and fairness being 0.594 and 0.361 respectively. These coefficients suggest that intelligence is more important than fairness for believability, but both play a role.

D. Correlating Quantitative and Qualitative Metrics

In order to determine if the players judge the human-likeness of the agents on their game-level playstyle, the standardized distance from the opponent behaviour vector to the human behaviour vector distribution was used to try to predict the player's answer on the human-likeness question for that opponent, but this failed, showing that there is no correlation between the two. The behaviour vector defines the game-level playstyle of the AI opponent and this does not influence the believability of the AI opponents. In the Methodology I introduced two other quantitative metrics: responsiveness (equation 6) and challenge (equation 7. In the next paragraph I will show the results for the correlation between these metrics and the participant survey responses.

Responsiveness is correlated with the perceived fairness (R^2 = 0.11, $p<0.01$). All AI opponents that were rated below a 4 out of 7 for fairness had a responsiveness of more than the human average of 0.78 (Table V). This suggests that for these players, the AI opponent responding to their moves too often was unfair. However, most of the AI opponents that had a responsiveness higher than 0.78 were rated at or above a 4 out of 7 for fairness. Responsiveness is also correlated with the perceived adaptiveness (R^2 = 0.08, $p<0.05$). Both correlations are weak, potentially because the responsiveness only captures a small part of how the AI opponent interacts with the human player, since it only tracks the number of attacks and not how those attacks were performed. The quantitative challenge metric is strongly correlated with perceived challenge (R^2 = 0.42, $p<0.0001$), making it a good predictor for the perceived challenge the human players experienced.

IV Discussion

In this research, I have created a 2-player strategy game and three different AI opponents which human players played against. I compared the human and AI players quantitatively and qualitatively, showing that AI players that followed the human game-level playstyles more closely were not perceived by players as more believable. Instead, the believability of the AI players was correlated with the perceived intelligence and fairness of the agent.

The HRL agents are capable of producing human-like game-level playstyles without training on human data. This emergence of human-like behaviour is useful in the development of AI opponents for games in development, because those games have little to no human play data available. This result sets this research apart from prior approaches that rely heavily on imitation learning or behavior cloning [11, 12, 42].

Adaptations to the HRL architecture resulted in efficient training and reasonably strong task-performance, although human players still win against it 80% of the time. The key adaptations are: re-evaluating decisions at every layer on each turn, assigning a dedicated agent to each unit, and allowing the recruitment agent to act only when recruitment is possible. These adjustments distinguish my implementation from standard HRL approaches, like [24], where only the bottom layer makes a decision at every turn and the middle and top layers make decisions every 4 and 48 turns respectively. Using a dedicated agent for each unit was also done in [25].

I developed a behaviour vector based on three metrics—Aggressiveness, Management and Exploration—consistent with existing literature on player behaviour in games [37]. This provides a consistent framework for analyzing and comparing playstyles across different games. By selecting appropriate submetrics for each metric, researchers can adapt the behaviour vector to suit their own game environments while maintaining compatibility with this common structure. The validity of this behaviour vector can be assessed using human player data, by comparing player consistency against a random baseline, as outlined in Section 4). Prior research into playstyles did not use a consistent framework, as can be seen in Table 1), and this prior research also did not perform this validation.

Research combining both playstyle profiling and believability analysis is currently underrepresented. Both fields of research would benefit from including the other. Playstyle profiling with the goal of having AI agents produce human-like behaviour can use the believability analysis to confirm or reject the assumption that more human-like playstyles results in more believable agents. In this research, I showed that the participants in the experiment did not rate HRL agents, whose playstyle were more human-like, as more believable than the Baseline bot. The validation of the playstyle comparison between humans and bots can be used to guide the determination of playstyle. For instance in this research, the game level playstyles were not validated, so I would focus on tactical and action-level playstyles next. Believability analysis research can benefit from playstyle profiling since it will allow for narrowing down what parts of human behaviour in games are most responsible for believability.

A. Limitations

The number of participants that completed the full experiment (27) was limited, as was the number of games each participant played during the full experiment (3). Testing with more participants for each AI opponent type could increase the statistical power so that more conclusions can be drawn. The completion rate comparison between the AI opponents were quite different, but still not statistically significant. Some questions on the traits of the opponent were answered quite differently depending on the opponent, but also not statistically significant.

Testing with more games per participant could reduce some of learning effects. Moreover, the player consistency comparison is stronger from around 5 games for each participant, since then the random case will have a consistency score closer to 0. Furthermore, there is just one engagement evaluation point after each game, so due to recency bias, this engagement evaluation will only say something about the last part of that game.

The ternary diagram (Figures 4 and 5) shows game-level strategies well, but does not have the capacity to show specific tactical and action-level strategies, like for instance if the player always attacks from a certain direction, or if a player always has a very specific build order of units. In casual play, as is the case for this experiment, the range of game-level playstyles is diverse, but this could be different for competitive play, where playstyles could be different in a more subtle way.

B. Future Work

The HRL agent designed in this research could be improved by including an opponent model in the architecture to improve the perceived adaptiveness of the agent by the players. For the HRL agents designed in this research, the perceived adaptiveness was at or below 4 on a 7 point Likert-scale. Furthermore, self-play can be used to improve the task performance, as currently the HRL agent loses around 80 percent of the time against human players.

I did not find a correlation between the believability of an AI opponent and the human-likeness of its game-level playstyles. What was not tested however, is whether playing against an AI opponent with the same playstyle repeatedly is less believable and engaging for the player than playing against an AI opponent that follows a human-like and diverse set of playstyles, with the AI opponent using a different playstyle each game, as if the player plays against a different human player each game.

Correlating the gameplay behaviour of the AI opponents and their believability was not completed. I did find a quantitative metric that was strongly correlated with the level of challenge the participants said they experienced. I also found a metric for responsiveness that was loosely correlated with the perceived fairness and adaptiveness. Finding explainable quantitative metrics that correlate strongly with each of the opponent traits and with believability could help steer the design of the AI opponents for the game the metrics were made for, and it could also give insights into what players value in their AI opponents in general. Ultimately, quantitative metrics could replace qualitative metrics entirely, enabling imitation learning without the need for human data.

V Conclusion

In this research, I developed a two-player turn-based strategy game and evaluated three AI opponents, including two hierarchical reinforcement learning (HRL) agents, to investigate how to quantify human-like behavior. While the HRL agents were able to exhibit game-level playstyles similar to human players without being trained on human data, this similarity did not translate to increased believability. Instead, perceived intelligence and fairness were stronger predictors of believability. By combining qualitative and quantitative analyses, I demonstrated how believability assessment can be used to validate playstyle profiling.

References

[1] Penelope Drennan, Stephen Viller, and Peta Wyeth. Engaging game characters: Informing design with player perspectives. In *Entertainment Computing–ICEC 2004: Third International Conference, Eindhoven, The Netherlands, September 1-3, 2004. Proceedings 3*, pages 355–358. Springer, 2004.

[2] Henrik Schoenau-Fog. The player engagement process–an exploration of continuation desire in digital games. In *Proceedings of DiGRA 2011 Conference: Think Design Play*, 2011.

[3] Xinyi Tan and Chin Ike Tan. Empathy in game design-exploring a human-centric approach in designing engaging video game experiences. *Journal of ICT in Education*, 9(2):123–136, 2022.

[4] Hua Qin, Pei-Luen Patrick Rau, and Gavriel Salvendy. Measuring player immersion in the computer game narrative. *Intl. Journal of Human–Computer Interaction*, 25(2):107–133, 2009.

[5] Zhijian Zhang, Haozheng Li, Luo Zhang, Tianyin Zheng, Ting Zhang, Xiong Hao, Xiaoxin Chen, Min Chen, Fangxu Xiao, and Wei Zhou. Hierarchical reinforcement learning for multi-agent moba game. *arXiv preprint arXiv:1901.08004*, 2019.

[6] Athul Paul Jacob, David J Wu, Gabriele Farina, Adam Lerer, Hengyuan Hu, Anton Bakhtin, Jacob Andreas, and Noam Brown. Modeling strong and human-like gameplay with kl-regularized search. In *International Conference on Machine Learning*, pages 9695–9728.

PMLR, 2022.

[7] Reid McIlroy-Young, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Aligning superhuman ai with human behavior: Chess as a model system. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1677–1687, 2020.

[8] Maciej Świechowski, Daniel Lewiński, and Rafal Tyl. Combining utility ai and mcts towards creating intelligent agents in video games, with the use case of tactical troops: Anthracite shift. In *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8. IEEE, 2021.

[9] Yu Iawasaki and Koji Hasebe. Identifying playstyles in games with neat and clustering. In *2021 IEEE Conference on Games (CoG)*, pages 1–4. IEEE, 2021.

[10] Diego Perez-Liebana, Cristina Guerrero-Romero, Alexander Dockhorn, Linjie Xu, Jorge Hurtado, and Dominik Jeurissen. Generating diverse and competitive play-styles for strategy games. In *2021 IEEE Conference on Games (CoG)*, pages 1–8. IEEE, 2021.

[11] Pierre Le Pelletier de Woillemont, Rémi Labory, and Vincent Corruble. Automated play-testing through rl based human-like play-styles generation. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 18, pages 146–154, 2022.

[12] Matthew Barthet, Ahmed Khalifa, Antonios Liapis, and Georgios Yannakakis. Generative personas that behave and experience like humans. In *Proceedings of the 17th International Conference on the Foundations of Digital Games*, pages 1–10, 2022.

[13] Bas Van Opheusden, Ionatan Kuperwajs, Gianni Galbiati, Zahy Bnaya, Yunqi Li, and Wei Ji Ma. Expertise increases planning depth in human gameplay. *Nature*, 618(7967):1000–1005, 2023.

[14] Markus Guhe and Alex Lascarides. Game strategies for the settlers of catan. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–8. IEEE, 2014.

[15] Bernard Gorman1 Christian Thurau2 Christian Bauckhage and Mark Humphrys. Believability testing and bayesian imitation in interactive computer games.

[16] Ryan Marcotte and Howard J Hamilton. Behavior trees for modelling artificial intelligence in games: A tutorial. *The Computer Games Journal*, 6:171–184, 2017.

[17] Devang Jagdale. Finite state machine in game development. *algorithms*, 10(1), 2021.

[18] Matteo Iovino, Edvards Scukins, Jonathan Styrud, Petter Ögren, and Christian Smith. A survey of behavior trees in robotics and ai. *Robotics and Autonomous Systems*, 154:104096, 2022.

[19] Ian Millington and John Funge. Artificial intelligence for games 2nd edition 2009. *Cité en*, page 63, 2009.

[20] Ron Sun. Theoretical status of computational cognitive modeling. *Cognitive Systems Research*, 10(2):124–140, 2009.

[21] Razan Ghzouli, Thorsten Berger, Einar Broch Johnsen, Andrzej Wasowski, and Swaib Dragule. Behavior trees and state machines in robotics applications. *IEEE Transactions on Software Engineering*, 2023.

[22] Henrik Schønau-Fog and Thomas Bjørner. "sure, i would like to continue" a method for mapping the experience of engagement in video games. *Bulletin of Science, Technology & Society*, 32(5):405–412, 2012.

[23] Daniel Gomme. *Player Expectations of Strategy Game AI.* PhD thesis, University of Essex, 2024.

[24] Yuhang Song, Jianyi Wang, Thomas Lukasiewicz, Zhenghua Xu, and Mai Xu. Diversity-driven extensible hierarchical reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 4992–4999, 2019.

[25] Weigui Jair Zhou, Budhitama Subagdja, Ah-Hwee Tan, and Darren Wee-Sze Ong. Hierarchical control of multi-agent reinforcement learning team in real-time strategy (rts) games. *Expert Systems with Applications*, 186:115707, 2021.

[26] Harshit Sethy, Amit Patel, and Vineet Padmanabhan. Real time strategy games: a reinforcement learning approach. *Procedia Computer Science*, 54:257–264, 2015.

[27] Stefan Wender and Ian Watson. Using reinforcement learning for city site selection in the turn-based strategy game civilization iv. In *2008 IEEE symposium on computational intelligence and games*, pages 372–377. IEEE, 2008.

[28] Remi Niel, Jasper Krebbers, Madalina M Drugan, and Marco A Wiering. Hierarchical reinforcement learning for real-time strategy games. In *10th International Conference on Agents and Artificial Intelligence*, pages 470–477. SciTePress, 2018.

[29] Matthias Hutsebaut-Buysse, Kevin Mets, and Steven Latré. Hierarchical reinforcement learning: A survey

and open research challenges. *Machine Learning and Knowledge Extraction*, 4(1):172–221, 2022.

[30] Manfred Eppe, Christian Gumbsch, Matthias Kerzel, Phuong DH Nguyen, Martin V Butz, and Stefan Wermter. Intelligent problem-solving as integrated hierarchical reinforcement learning. *Nature Machine Intelligence*, 4(1):11–20, 2022.

[31] Matthew Michael Botvinick. Hierarchical reinforcement learning and decision making. *Current opinion in neurobiology*, 22(6):956–962, 2012.

[32] Xinyi Xu, Tiancheng Huang, Pengfei Wei, Akshay Narayan, and Tze-Yun Leong. Hierarchical reinforcement learning in starcraft ii with human expertise in subgoals selection. *arXiv preprint arXiv:2008.03444*, 2020.

[33] Hoang Le, Nan Jiang, Alekh Agarwal, Miroslav Dudík, Yisong Yue, and Hal Daumé III. Hierarchical imitation and reinforcement learning. In *International conference on machine learning*, pages 2917–2926. PMLR, 2018.

[34] Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. Hierarchical reinforcement learning: A comprehensive survey. *ACM Computing Surveys (CSUR)*, 54(5):1–35, 2021.

[35] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[36] Philipp Moritz, Robert Nishihara, Stephanie Wang, Alexey Tumanov, Richard Liaw, Eric Liang, Melih Elibol, Zongheng Yang, William Paul, Michael I Jordan, et al. Ray: A distributed framework for emerging {AI} applications. In *13th USENIX symposium on operating systems design and implementation (OSDI 18)*, pages 561–577, 2018.

[37] Jukka Vahlo, Jouni Smed, and Aki Koponen. Validating gameplay activity inventory (gain) for modeling player profiles. *User modeling and user-adapted interaction*, 28(4):425–453, 2018.

[38] Johanna Pirker, Simone Griesmayr, Anders Drachen, and Rafet Sifa. How playstyles evolve: progression analysis and profiling in just cause 2. In *Entertainment Computing-ICEC 2016: 15th IFIP TC 14 International Conference, Vienna, Austria, September 28-30, 2016, Proceedings 15*, pages 90–101. Springer, 2016.

[39] PC MAHALANOBIS. On the generalised distance in statistics. In *Proceedings of the National Institute of Science of India*, volume 12, pages 49–55, 1936.

[40] Cindy Even, Anne-Gwenn Bosser, and Cédric Buche. Assessing the believability of computer players in video games: A new protocol and computer tool. *Frontiers in Computer Science*, 3:774763, 2021.

[41] Siqi Guo. Developing a scale for measuring the believability of virtual agents. In *International Conference on Artificial Reality and Telexistence & Eurographics Symposium on Virtual Environments (ICAT-EGVE)*. Eurographics Digital Library, 2023.

[42] Juan Ortega, Noor Shaker, Julian Togelius, and Georgios N Yannakakis. Imitating human playing styles in super mario bros. *Entertainment Computing*, 4(2):93–104, 2013.