Evaluating Navigation Behavior of Agents in Games using Non-Parametric Statistics

Ian Colbert
Advanced Micro Devices, Inc.
San Diego, United States
ian.colbert@amd.com

Mehdi Saeedi
Advanced Micro Devices, Inc.
Markham, Canada
mehdi.saeedi@amd.com

***Abstract*—Recent advancements in deep reinforcement learning have demonstrated highly skilled agents that are capable of complex behavior. In video games, such agents are increasingly deployed as non-playable characters (NPCs) to enhance the gaming experience, as convincing human-like behavior is known to increase player engagement. However, the believability of an agent's behavior is often measured solely by its proficiency at a given task, which alone is not sufficient to discern human-likeness. In this paper, we build a non-parametric two-sample hypothesis test to compare the behaviors of NPCs to those of human players using distributions of their movement patterns. We show that the resulting p-value metric not only aligns with anonymous human judgment of human-like behavior, but it can also be used as a measure of similarity.**

***Index Terms*—Human-like Behavior, Game AI, Navigation, Non-parametric Statistics, Reinforcement Learning**

I. Introduction

The development of non-playable characters (NPCs) has received increased attention in recent years due to impressive successes in training highly skilled agents to play complex games [1], [2]. The believability of these agents is often contextually measured by their proficiency in accomplishing a specifically designed goal; however, the behavior that an agent exhibits in the process is equally as important. In fact, it has been shown that developing artificial agents to emulate human-like behavior leads to increased engagement in games [3]. Yet, the focus of game AI research is heavily skewed towards the development of specific *skills*, leaving the development and analysis of specific *behaviors* an open challenge.

The core contribution of this work is our non-parametric statistical hypothesis testing framework, which we built to measure the behavioral similarity of any two agents. We demonstrate the efficacy of this framework within the context of evaluating the human-like navigation behavior of NPCs in 3D space. While previous work has also sought to use statistics to analyze navigation behavior [4], they holistically evaluate entire episodes using parametric models of navigation paths collected from a fixed environment. Unlike previous work, we focus on using non-parametric statistics to measure the similarity in navigation behavior using distributions of fixed-length movement patterns collected from randomly generated environments. We combine the use of kernel-based divergence metrics with statistical resampling to analyze the differences in these movement patterns without making assumptions about how data is distributed.

Our hypothesis test extends the PT-MMD [5] framework from the domain of generative modeling into that of 3D navigation. By splitting entire navigation paths into a distribution of fixed-length movements, our framework is able to recognize differences in navigation behavior through variations in movement patterns using non-parametric statistical inferencing. Thus, when controlling the sensitivity of the test, we are able to rank NPCs by their similarity to the navigation behavior of human players using the resulting p-value. The interpretation of p as a similarity measure has been successfully applied in contexts such as clustering [6], and we observe sufficient stability to extend it to evaluating 3D navigation. Furthermore, we show that our p-value metric can be used as a measure of similarity that aligns with anonymous human judgment of human-like behavior. To the best of our knowledge, we are the first to propose a systematic ranking criteria for NPCs using a statistical measure of human-likeness.

II. Related Work

Standard approaches to evaluating human-like navigation behavior require either expert human judges that rely heavily on time-intensive manual efforts, or domain-specific metrics that fail to capture fine-grained details of human-likeness [7], [8]. Consequently, there has been growing interest in designing automated proxies that leverage machine learning techniques [9], [10]. These works are primarily motivated to classify human-like navigation using datasets of trajectories pulled from various types of artificial or biological agents; thus, the outcome of their work is a detection model that produces as output a probability that a sample is human. The general applicability of such techniques outside the environment in which the model is trained requires further study.

III. The Human-like Behavior Hypothesis Test

To construct our test, we represent navigation behavior as a distribution of movements using episodic trajectories collected from an agent. We define an *episode* as a sequence of state-action pairs spanning an initial state s_0 to a terminal state s_N. We define a *trajectory*, denoted as τ, as a continuous subsequence of an episode. To motivate our test, we articulate our *behavioral similarity hypothesis*—the behaviors of any two

agents are sufficiently similar if the distributions over their respective trajectories are sufficiently similar.

A. Representing the Navigation Behavior of Agents

To represent the navigation behavior from each episode, we use the absolute 3D location of the agent at each time step. We then transform each episode into a distribution of movements by subsampling fixed-length trajectories uniformly with replacement from each episode using a time horizon denoted by T. More formally, let c_t be the 3-dimensional Cartesian coordinates of an agent at time t such that $c_t = (x_t, y_t, z_t)$, and let $\mathbf{c}(\tau)$ be the sequence of coordinates for a given trajectory τ such that $\mathbf{c}(\tau) = \{c_0, \cdots, c_T\}$. Given an episode of length N, we consider overlapping trajectories to be uniquely different such that $\mathbf{c}(\tau_i)$ and $\mathbf{c}(\tau_j)$ have the same probability $\frac{1}{N-T}$ of being sampled for $T \leq N$. To ensure that we are analyzing movement without being biased by absolute location, we subtract the initial Cartesian coordinate c_0 from each sample $\mathbf{c}(\tau)$ so that each movement starts from the origin.

Because we use a set of episodes, we independently subsample K trajectories of length T from each episode with replacement. In large environments, the number of time steps taken to complete even simple tasks is heavily skewed right. Thus, to correct for any biases from larger episodes, we set K to the length of the largest episode in a given set. This ensures that a sampled trajectory has a uniform probability of being drawn from any of the episodes collected.

B. Evaluating Behavioral Similarity using Hypothesis Testing

Our central hypothesis posits that the similarity between the behaviors of any two given agents can be estimated by the similarity between their respective distributions of trajectories. To test this, we consider the setting in which sample distributions X and Y are independently drawn from distributions P^* (for a human player) and P_θ (for another agent), respectively. To measure their behavioral similarity, we evaluate the null hypothesis (H_0) against the alternative hypothesis (H_1), as summarized below. Note there is no point-to-point correspondence between X and Y, making the test independent of the source of the data and nature of the agent.

$$H_0 : P^* = P_\theta$$
$$H_1 : P^* \neq P_\theta$$

Our test is motivated by the following insight: if the null hypothesis is true, then any difference between P^* and P_θ should be due to sampling error. To evaluate the differences in these distributions, we use maximum mean discrepancy (MMD) and m out of n bootstrap resampling. MMD is a kernel-based divergence metric often used to compute the distance between the projections of two high-dimensional data distributions [11]. In our experiments, we apply its pairwise estimation formulation using the standard Gaussian kernel and Euclidean distance function. With m out of n bootstrap resampling, samples of size m are repeatedly drawn with replacement from a sample distribution of size n to recompute a sample statistic without making *a priori* assumptions of how the data is distributed [12]. Thus, to derive our p-value, we evaluate and compare distributions of MMD distances in two settings: separated and pooled sample distributions.

First, we consider the setting in which we evaluate over separated distributions X and Y. Given that $\mathbf{x}_i$ and $\mathbf{y}_i$ each denote subsamples of size m that are independently drawn with replacement from X and Y, respectively, we form a distribution of MMD distances by repeatedly recomputing $\text{MMD}_k[\mathbf{x}_i, \mathbf{y}_i]$ over S iterations where $i \in \{1, \cdots, S\}$. We refer to this distribution of distances as $\delta_{X,Y}$. Our test statistic, which we denote as δ, is then calculated using Eq. 1, where $\text{quantile}(\delta_{X,Y}, \alpha)$ returns the α-th quantile over the distribution $\delta_{X,Y}$ and α is a hyperparameter designed to control the sensitivity of the test.

$$\delta = \text{quantile}(\delta_{X,Y}, \alpha) \text{ where } \alpha \in (0, 1) \tag{1}$$

Next, we combine sample distributions X and Y to create a pooled sample distribution, which we refer to as Z. To evaluate in this setting, we form a distribution of MMD distances by repeatedly recomputing $\text{MMD}_k[\mathbf{x}_i, \mathbf{y}_i]$ over another S independently drawn samples where $i \in \{1, \cdots, S\}$; however, in this setting, $\mathbf{x}_i$ and $\mathbf{y}_i$ are both independently sampled from pooled distribution Z with replacement. We refer to this distribution of estimates as δ_Z.

Finally, to evaluate our null hypothesis, we define our p-value as the percentage of estimates greater than our test statistic δ, as shown in Eq. 2.

$$p = \frac{\#(\delta_Z > \delta)}{N} \tag{2}$$

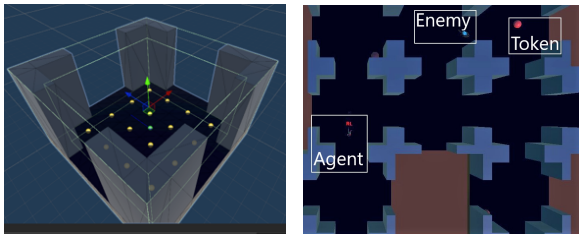
Given that P^* and P_θ are the same distribution, then distribution $\delta_{X,Y}$ should be the same as distribution δ_Z. Thus, when $P^* = P_\theta$, it follows that p converges towards $1 - \alpha$ as $S \to \infty$. When $P^* \neq P_\theta$, we can interpret p as a measure of closeness between distributions P^* and P_θ. Low values of p indicate that the movement patterns in sample distributions X and Y are not similar. Because this measure is inherently noisy, we evaluate the test over multiple runs and report the interquartile range (IQR) to demonstrate stability.

IV. Experimental Analysis

To evaluate the efficacy of our hypothesis test, we deploy agents to complete a navigation task in a controlled 3D virtual environment as shown in Fig. 1.

A. Environment and Navigation Task

Each environment is divided into N equal-sized segments, each sub-divided into M spawn points which can generate a token with uniform probability p_{token}. These spawn points similarly govern the procedural generation of enemies, but with probability p_{enemy}. Enemies have a visual radius of roughly 5 segments and are controlled using a NavMesh [13] to target and attack the agent without collaboration. In our experiments, $p_{\text{token}} = 75\%$, $p_{\text{enemy}} = 25\%$, $M = 16$, and N is sampled from a discrete uniform distribution between 5 and



(a) (b)

Fig. 1: The primitive segment (left) used to generate random maze-like environments (right). The platform is colored in **black**, while off-platform (*i.e.,* falling off the map) is colored in **brown**.

30, and all generated segments are connected. At the start of each episode, an agent is deployed to collect tokens that are randomly scattered around the rendered maze. To successfully complete the navigation task, an agent must collect all of the tokens available within the environment without falling off the map or being attacked by an enemy. An episode ends either when all tokens are collected or if the maximum step count is reached. If an agent falls off the map or is attacked by an enemy, the agent is respawned to continue.

B. Agent Characteristics

We study the behavior of human players, agents trained using reinforcement learning (RL-based), and NavMesh-based agents, each bounded to the same action space and rendered as humanoid characters with physics-based movement dynamics. At each time step, an agent has two sets of independent actions: (1) move forward or not; and (2) move left or right or not, which yields actions such as "Forward-Left".

Human players are controlled using WASD keys. Unlike other agents, the observation space of the human player is a first-person point-of-view of the humanoid agent.

RL-based agents are controlled by a policy modeled by a 3-layer linear DNN inspired by [1] to use self-attention, and optimized using proximal policy optimization (PPO) [14]. To form its observation space, the agent is equipped with several raycast sensors that detect walls, floors, enemies, and tokens. In our experiments, we use a simple reward signal where the agent receives +1 for collecting a token, -1 for either falling off the map or being attacked, and 0 otherwise.

NavMesh-based agents are controlled using an omniscient behavior tree that greedily collects tokens according to proximity by using a NavMesh [13] as its guide.

C. Comparing the Human-Like Behavior of Agents

To visualize the complexities in the navigation behaviors of each agent, we show four examples of episodic trajectories in Fig. 2. Note that the navigation behavior of each agent varies greatly within the same environment. Previous work has shown that RL-based agents exhibit more human-like navigation behavior than NavMesh-based counterparts [2]. Qualitatively, we observe this to hold true with our agents. To quantify these differences, we evaluate each agent using

(a) (b) (c) (d)

Fig. 2: The navigation behavior of each agent over four different episodes. Here, blue and orange are the two human players, while green and red are the RL-based and NavMesh-based agents.

α	T=4	T=8	T=16	T=32
0.10	70.1% (2.35%)	69.1% (2.55%)	79.4% (1.77%)	77.3% (3.13%)
0.25	48.4% (2.35%)	47.7% (2.40%)	62.7% (1.86%)	62.4% (1.93%)
0.50	24.9% (2.43%)	24.7% (2.78%)	40.0% (2.53%)	39.6% (2.45%)

TABLE I: Navigation behavior comparison between two human players across time horizons (T) and alpha levels (α). We report the median p-value and IQR (in parenthesis). Experiments show that the hypothesis test yields high values for human-to-human likeness.

our hypothesis test. For an even comparison, we deploy each agent to complete the navigation task in the same 40 mazes and sample their 3D locations directly from the game engine. We repeat each experiment 10 times using $S = 1000$ iterations and a subsample size $m = 1000$.

We first evaluate human-to-human likeness by comparing the sample datasets collected from each human player, and report the results in Table I. Next, we apply our hypothesis test to compare the navigation behaviors of both the RL-based and NavMesh-based agents to those of the human players. We use the aggregated 80 episodes from both humans to form our sample set X, and evaluate each agent over the same 40 mazes. With $T = 32$ and $\alpha = 0.10$, we observe a median (and IQR) p-value of 78.3% (2.23%) and 0.0% (0.00%) for the RL-based and NavMesh-based agents, respectively. Finally, we directly compare the human-likeness of the RL-based and NavMesh-based agents by evaluating the median difference in their p-values within each of the 40 maze environments. In this experiment, we reduce the number of repeats to 3 to obtain our median and IQR and keep all other hyperparameters the same. We observe that the navigation behavior of the RL-based agent is more similar to human players than its NavMesh-based counterpart in 89.5% of episodes with a median difference of 29.2% and a maximum IQR of 5%.

D. Human Judgment of Human-like Behavior

In their study of human-like navigation behavior, Devlin *et al.* [10] trained two RL-based agents, which they refer to as *symbolic* and *hybrid* agents, to complete a navigation task with a level of success sufficiently similar to that of human players so as to focus solely on learned behavior. To analyze human judgment of human-like behavior, they design a Navigational Turing Test in which they administer a survey to 60 human assessors. They report not only that participants were able to accurately detect human players with statistical significance, but also that the behavior of the hybrid agent was judged to

	α	Human	Hybrid	Symbolic
T = 4	0.10	90.5% (0.95%)	19.6% (1.63%)	8.6% (2.28%)
	0.25	75.3% (3.50%)	7.0% (1.48%)	2.8% (0.88%)
	0.50	50.9% (1.93%)	1.7% (0.25%)	0.4% (0.25%)
T = 8	0.10	89.7% (1.80%)	19.7% (1.43%)	8.5% (0.75%)
	0.25	76.7% (2.75%)	6.6% (1.20%)	2.5% (0.45%)
	0.50	50.5% (1.63%)	1.6% (0.25%)	0.5% (0.18%)
T = 16	0.10	90.4% (1.38%)	20.6% (3.03%)	8.7% (0.85%)
	0.25	73.5% (2.48%)	7.6% (0.97%)	2.6% (0.28%)
	0.50	49.8% (2.87%)	1.7% (0.45%)	0.5% (0.18%)
T = 32	0.10	88.9% (1.60%)	20.9% (4.98%)	7.8% (2.60%)
	0.25	75.2% (1.50%)	8.0% (1.65%)	3.1% (1.15%)
	0.50	50.0% (1.78%)	1.8% (0.75%)	0.5% (0.45%)

TABLE II: Using data provided by [10], we evaluate the human-to-human likeness using random splits of the human player data, and the human-to-hybrid and human-to-symbolic evaluations using the full distributions of movements collected from each agent. We report the median p-value and IQR (in parenthesis).

be more human-like when directly compared to the symbolic agent. Furthermore, they provide a dataset composed of 100 episodes collected from a pool of anonymous human players, and 50 episodes separately collected from pre-trained symbolic and hybrid agents. To evaluate the efficacy of our hypothesis test, we benchmark it against this human judgment of human-like behavior using the data reported in their study. Note that our test evaluates navigation patterns, and is intentionally left unaware of player skill levels or gameplay strategies.

We first analyze our test's ability to measure human-to-human likeness for this navigation task. Both the environment and game engine are different from the previous experiment. Because human player data is anonymously pooled, we use random splits of the data over $S = 1000$ iterations with a subsample size of $m = 250$. We repeat each experiment 10 times and report the median p-value and IQR across different time horizons (T) and alpha levels (α) in Table II. We then compare the behaviors of the symbolic and hybrid agents to those of human players. Using the aggregated 100 episodes from all human players to form our sample set X, we evaluate each agent using their 50 respective episodes. We observe that our test aligns with the human judgment reported in [10] as the hybrid agent is determined to be more human-like than the symbolic agent across selections of T and α. Note that across all selections of T and α, the human-to-human likeness is greater than the human-likeness of both the hybrid and symbolic agents.

V. Conclusion and Future Work

We proposed a non-parametric two-sample hypothesis testing framework based on statistical resampling methods to measure behavioral similarity through the lens of 3D navigation in games. We demonstrate that our test is agnostic to the nature of the agent by comparing the human-like navigation behavior of both RL-based and NavMesh-based agents. Finally, we use the results reported by [10] to show that the p-value metric resulting from our test can be used as a measure of similarity that aligns with anonymous human judgment.

The study of human-like behavior has a rich history that intersects across a variety of domains, each with a unique vantage point of the same larger problem. Yet, understanding the driving factors for emulating human-likeness still remains an open challenge. With the rapid pace of research surrounding the development and analysis of human-like NPCs, we hope more researchers adopt and further develop methods to analyze behavior through the lens of distributions. As humans tend to be "creatures of habit", we conjecture that analyzing distributions of behavior could yield further insights into its driving factors. Furthermore, such formulations may be used to reward agents to learn such behaviors as distribution divergence measures could provide stability whereas static classifiers tend to be exploited [10]. We leave explorations into reward design or imitation learning for future work.

Acknowledgements

We would like to thank Gabor Sines, Thomas Perry, Alex Cann, Tian Yue Liu, and the rest of the AMD Software Technology and Architecture team for insightful discussions and infrastructure support.

References

[1] B. Baker *et al.*, "Emergent tool use from multi-agent autocurricula," *arXiv preprint:1909.07528*, 2019.

[2] E. Alonso *et al.*, "Deep reinforcement learning for navigation in AAA video games," *arXiv preprint:2011.04764*, 2020.

[3] B. Soni and P. Hingston, "Bots trained to play like a human are more fun," in *International Joint Conference on Neural Networks*, pp. 363–369, IEEE, 2008.

[4] H.-K. Pao *et al.*, "Game bot detection via avatar trajectory analysis," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, no. 3, pp. 162–175, 2010.

[5] A. Potapov *et al.*, "PT-MMD: A novel statistical framework for the evaluation of generative systems," in *53rd Asilomar Conference on Signals, Systems, and Computers*, pp. 2219–2223, IEEE, 2019.

[6] M. Bompais *et al.*, "The p-value as a new similarity function for spectral clustering in sensor networks," in *Statistical Signal Processing Workshop*, pp. 95–99, IEEE, 2018.

[7] I. V. Karpov *et al.*, "Believable bot navigation via playback of human traces," in *Believable bots*, pp. 151–170, Springer, 2013.

[8] R. Kirby *et al.*, "Companion: A constraint-optimizing method for person-acceptable navigation," in *18th International Symposium on Robot and Human Interactive Communication*, pp. 607–612, IEEE, 2009.

[9] W. J. de Cothi, *Predictive maps in rats and humans for spatial navigation*. PhD thesis, University College London, 2020.

[10] S. Devlin *et al.*, "Navigation turing test (NTT): Learning to evaluate human-like navigation," in *International Conference on Machine Learning*, pp. 2644–2653, PMLR, 2021.

[11] A. Gretton *et al.*, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.

[12] P. J. Bickel and J.-J. Ren, "The bootstrap in hypothesis testing," *Lecture Notes-Monograph Series*, pp. 91–112, 2001.

[13] G. Snook, "Simplified 3D movement and pathfinding using navigation meshes," in *Game Programming Gems* (M. DeLoura, ed.), pp. 288–304, Charles River Media, 2000.

[14] J. Schulman *et al.*, "Proximal policy optimization algorithms," *arXiv preprint:1707.06347*, 2017.