

Game Ai Using Parametrized Behavior Monitoring

Raghav Suriyashekar¹, C.N.S.Vinoth Kumar^{2*}

¹Department of Computer Science and Engineering, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur Chennai, India.

²Department of Computer Science and Engineering, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur Chennai, India.

¹rs5200@srmist.edu.in, ²vinothks1@srmist.edu.in

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 10 May 2021

Abstract: Game AI with a Non-Player Controlled Bots is the systems that are built into a game. These systems offer competition or objectives that the player must overcome to progress. Although in most situations, these NPCs provide an acceptable level of competition, over time, they become predictable and no longer pose any challenge to the players. Game AIs and NPCs use a basic detection and computational system to perform actions that seem to oppose the players, and these are even lesser of an opposition to players and will lead to loss of worth in the section of the game that would be more enjoyable to those if the opposition would be more challenging.

Keywords: Parametrized, Behavior Monitoring, NPC, Bots, Player Controlled, Conventional Game AI, Game AI, DQN, mimic.

1. Introduction

The world of gaming is one that is ever increasing. Its user-base grows by the millions with every new game released. Games are played by millions of people every day irrespective of its genre. They offer some immersive experience away from the real world. In all its vastness, almost all games share one common feature, NPCs (Non-Player Controlled (Bots)), or Non-Player Controlled Characters, more commonly known as Bots or Game AI [1]. These games thus have built in systems that challenge players in many ways. These bots exist within games architecture to offer some form of competition or challenge to the player to introduce some form of variety in the gameplay [2].

These bots use specified logic to perform actions that seem human to provide some challenge. One of the most common techniques is using FFSM or Fuzzy Finite State Machines to dictate the actions the bot [3] must perform. These systems collect inputs in the form of observations from the environment and information about the player such as position, movement speed, look direction etc., and classify them based on categories and evaluates them into states. These states are then further classified, and a final “decision” or state is arrived at, which is the action that the bot will take. While this method is very effective at providing challenges to the players, it eventually becomes very uninteresting, as a player can figure out the actions that the bot is going to take and thus the opposition the bot provided will no longer exist [4].

Many games have this method employed and while it works well, over time it cannot keep up with a human’s ability to evolve and beat it. Certain games like chess, however, also have Game AI that develops strategies, but does so in a rather brute force way. It will physically calculate millions and millions of outcomes based on current moves and by process of elimination; it will come up with an optimal winning strategy [5-6]. One such method is alpha-beta pruning of a decision tree which will allow the chess engine to discard moves that are irrelevant [7-9].

With the rapid improvement in technological capabilities surrounding gaming and game development, more and more variety is being brought forward. And with the development of more interesting games, it gives us an opportunity to develop better systems to make these games more enjoyable [10-11]. Although different methods of Game AI are being developed, they are either too powerful for humans to face off against, such as DQN based bots, or fail to give off a human player like gameplay feel [12] and fail the Turing Test[13].

While players don’t often mind the lack of non-responsive and opposition capable of evolving, it improves the overall appeal of the game when there is a new challenge every time the player plays the game. It is this evolution that makes multiplayer games so popular since they play against other people capable of doing something new in terms of gameplay and strategy. Although single player games by their definition do not involve playing against evolving entities, some parts of the game such as facing a boss character and having it evolve in style of gameplay will increase the level of immersion in the game and make it more enjoyable.

2. Related works

A. Introduction to Game AI

Game AI refers to the NPCs or the Non-Player Controlled Elements of a game. This ranges from simple elements such as doors or gates that allow for interactivity in between a level or in between levels and characters that the player can interact with throughout the course of the game. Furthermore, Game AI has become a standard part of certain game genres, such as RPGs or Role-Playing Games. It has become an integral part to the game's appeal and how immersive it will be. Almost all games currently in the market today will have some form of Game AI, no matter how rudimentary its application is. This game AI however in its current implementation in the games in the market today, is not in fact AI. They use a complicated system of finite state logic to decide the actions that the bot or NPC must perform.

B. Current Status and Implementation of Game AI

There are currently many implementations of Game AI that utilize a variety of methods to achieve the desired feel of interactivity. Some popular techniques of Game AI are: Finite State Machines: Finite State Machines divide the game object's potential behavior into possible logical states and computes the most ideal action to take. Agents: Agents perceive their environment and act to pursue those goals. They can be used in a known area against limited competition but cannot sustain the variety of gameplay that humans can achieve. Scripting: Basically scripting offers solutions to control game engines from the outside environment. This cannot understand and mimic player gameplay as they, like FSMs require hard coded constrains.

Fuzzy Logic: Although more flexible than FSMs, they still use categorization and still are bound by its states that it will have to be in to decide, although they do have the ability to be in multiple states at once, hence the term "fuzzy" and as a result, can take more complicated decisions.

Flocking: Flocking is generally used to control many entities in-game (like herds of sheep or schools of fish). Neural Networks: Neural Networks have the task of choosing the variables that it will use to decide its action. This can be successfully implemented in a simple game with few unknowns. Decision Trees: It is very difficult to account for all the possibilities to give unique or reactive decisions to new situations using a finite set of action (leaves) of a decision tree. Simulating a player and all the actions they can perform in any situation cannot be captured easily onto the leaves of decision trees.

All these methods use some form of rigid programming to derive "decisions" to make based on some configuration of its environment. As revied in [14], these game AI methods, excluding a few do not have the capability to evolve in style of gameplay or in the quality of decisions they take and hence do not give the feel of playing against another human [15].

As shown in a certain level of evolution in gameplay and responsiveness of the challenging parts is beneficial to the appeal of a game and its playability [16]. There are, however, systems in place such as DQNs that do learn and evolve in their "style" of gameplay but are however of a caliber that humans cannot achieve due to the enormity of its processing power and iterations they go through to find optimal success paths [17]. Systems [18] that use objectives to evolve using both traditional player progress and human behavior are more likely to imitate human behavior in games with a higher accuracy.

Systems such as the Open AI are designed to play one game very well by studying the best humans in the game and iterating through all the moves and strategies utilized by the players to develop a strategy of its own. While this does provide a very close feel towards playing against a human, its inhumanly low reaction time towards stimuli in game and very low response time towards the moves it must take or avoid also hinder the experience of playing against the bot. As shown in [19-20], it is found that human-like NPCs are far more fun to play against and it helps in the overall enjoyability of the game.

C. Drawbacks of the Current Implementations

The current implementation of Game AI cannot modify its behavior and are bound by the parameters they are set in. This leads to a repetitive nature and an unfulfilling experience on further indulgence of the game. Due to the lack of evolution and change, upon repetition, the opposition and challenge offered by the bots, become negligible and make the game uninteresting to continue playing.

3. Proposed methodology on game ai

The Proposed Parameterized behaviour monitoring is system capable of monitoring player behaviour and mimicking their gameplay in an accurate manner.

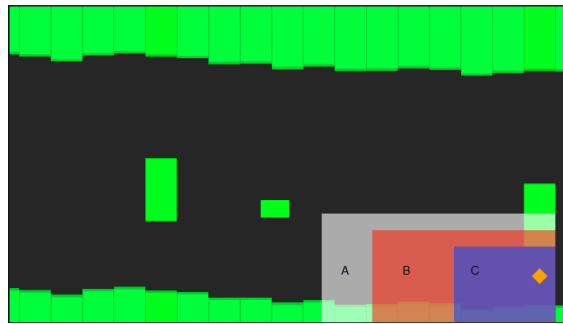


Figure.1 Regions A, B, C is a part of the detector.

This implementation was done in the Unity3D game engine. It is a very powerful physics and rendering engine. These systems were implemented through the scripts in C#.

This method involves setting up a detector script that can monitor the position of the player. The detector will observe the player in three zones. These zones are set up to be leading towards the objective. The zones namely A, B, C shown in Figure.1 will have each responsibility for monitoring the behaviour of the player.

The three zones of the detector will each correspond to the tendency of the player to collect an objective in the game. The tendency of the player to go into each zone will be calculated as the tendency of the player to crossover and proceed to the deeper zone i.e., tendency of entering zone A will be increased if the player has visited the zone B and so on till the tendency of zone C will be calculated as how many times the player has successfully collected the objective. This tendency will be monitored and updated in a script. The tendency will dictate the situations and the conditions in which the player performed certain actions. This script will be responsible for relaying the behavioural information to the bot to recreate the player gameplay through the “MIMIC” bot, another script that takes the behavioural data and converts it into gameplay to review.

Monitoring the player’s frequency and tendency to collect the objective through the amount of space available will be accurately re-created by the MIMIC and we will be able to observe a digital copy of the player’s gameplay.

4. Conclusion and future work

The proposed method of Game AI will be capable of monitoring several actions of the player and be able to adapt to their playstyle. This will help in evolving a digital model of a human and their behavior in situations involving more than just decision making but also a sense of strategy and risk associated with that action. This method can be further developed to be able to monitor many more parameters and hence be capable of simulating human decisions with a more complicated situation and will be able to build a comprehensible digital model of the human. With the further development of this system, a model can be created of a human in more complicated games, such as FPS or First-Person Shooter games which have many more parameters and the number of decisions that can be taken is very large and the possibilities of player action are endless. By monitoring the tendency of humans to make certain decision and to avoid making certain decisions the system can come close to mimicking the actions of the player.

References

1. Bhuman Soni, Philip Hingston, “Bots trained to play like a human are more fun”, DOI : 10.1109/IJCNN.2008.4633818, Conference Proceeding, Neural Networks, IJCNN, July 2008.
2. Shohei Miyashita, Xinyu Lian, Xiao Zeng, Takashi Matsubara, Kuniaki Uehara , ”Developing Game AI Agent Behaving Like Human by Mixing Reinforcement Learning and Supervised Learning”, DOI:10.1109/SNPD.2017.8022767, 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), June 2017.

3. Niels van Hoorn, Julian Togelius, Daan Wierstra and Jurgen Schmidhuber, "Robust player imitation using multiobjective evolution", DOI:10.1109/CEC.2009.4983007, Evolutionary Computation, CES, IEEE Congress, June 2009.
4. Chris Pelling, Henry Gardner, "Two Human-Like Imitation-Learning Bots with Probabilistic Behaviors", DOI:10.1109/ITW.2010.5593336, Computational Intelligence and Games (CIG), 2019 IEEE Symposium, September 2019.
5. Geoff Skinner, Toby Walmsley, "Artificial Intelligence and Deep Learning in Video Games A Brief Review", DOI:10.1109/CCOMS.2019.8821783, IEEE 4th International Conference on Computer and Communication Systems (ICCCS), February 2019.
6. A.Saranya, R.Naresh "Cloud Based Efficient Authentication for Mobile Payments using Key Distribution Method", Journal of Ambient Intelligence and Humanized Computing, Springer, 02 January, 2021.
7. R.Naresh, P.Vijayakumar, L. Jegatha Deborah, R. Sivakumar, "A Novel Trust Model for Secure Group Communication in Distributed Computing", Special Issue for Security and Privacy in Cloud Computing, Journal of Organizational and End User Computing, IGI Global, Vol.32, No. 3, Septemer 2020, Pp. 1-14.
8. R.Naresh, M.Sayeeekumar, G.M.Karthick, P.Supraja, "Attribute-based hierarchical file encryption for efficient retrieval of files by DV index tree from cloud using crossover genetic algorithm", Soft Computing, Springer, Vol.23, No. 8, 2019, Pp. 2561-2574.
9. R Divya Mounika, *R.Naresh, "The concept of Privacy and Standardization of Microservice Architectures in cloud computing", European Journal of Molecular & Clinical Medicine, Vol 7, No 2, Pages 5349-5370, Dec 2020.
10. P.Vijayakumar, R.Naresh, L. Jegatha Deborah, SK Hafizul Islam, "An efficient group key agreement protocol for secure P2P communication", Security and Communication Networks, Wiley, Vol.9, No.17, pp.3952-3965, 2016
11. P.Vijayakumar, R.Naresh, SK Hafizul Islam, L. Jegatha Deborah "An Effective Key Distribution for Secure Internet Pay-TV using Access Key Hierarchies", Security and Communication Networks, Wiley, Vol.9, No.18, pp.5085-5097, 2016.
12. R. Naresh, M Meenakshi, G Niranjana, "Efficient study of Smart Garbage Collection for Ecofriendly Environment", Journal of Green Engineering, Vol.10, No.1, pp.1-10, Feb 2020.
13. C.N.S.Vinoth Kumar, A.Suhasini, "Secured Three-Tier Architecture for Wireless Sensor Networks Using Chaotic Neural Networks", 'Advances in Intelligent Systems and Computing' AISC Series, Springer Science + Business Media Singapore 2017 Vol. No. 507, Chapter No. 13, pp. No. 129-136, ISSN 2194-5357, DOI 10.1007/978-981-10-2471-9_13
14. Gautam Srivastava, C.N.S. Vinoth Kumar, V Kavitha, N Parthiban, Revathi Venkataraman, "Two-Stage Data Encryption using Chaotic Neural Networks", Journal of Intelligent and Fuzzy systems, Vol. no.38, Issue. No.3, pp no.2561-2568, ISSN No: 1875-8967. March 2020
15. Praharsha Sarma, Utkarsh Kumar, C.N.S.Vinoth Kumar, M.Vasim Babu, "Accident Detection And Prevention Using Iot & Python Opencv", International Journal Of Scientific & Technology Research(IJSTR), Volume 9, Issue 04, pp no. 2677-2681, ISSN No: 2277-8616 April 2020.
16. M.Vasim Babu, C.N.S. Vinoth Kumar, M.Venu, International journal entitled "Improvisation of localization accuracy using ERSSI based on ADV-HOP algorithm in wireless sensor network", International journal of innovative technology and exploring engineering (IJITEE), ISSN No.2278-3075 Feb 2019.
17. Jung-Ying Wang, "Classification of Humans and Bots in Two Typical Two-player Computer Games", 3rd International Conference on Computer and Communication Systems (ICCCS), April 2018.
18. Sweetser, Penelope & Wiles, Janet, "Current AI in games: A Review". Australian Journal of Intelligent Information Processing Systems, 8(1), pp. 24-42, 2002.
19. Philip Hingston, "A Turing Test for Computer Game Bots", DOI:10.1109/TCIAIG.2009.2032534, IEEE Transactions on Computational Intelligence and AI in Games 1(3):169 – 186, October 2009.
20. Johan Hagelback, Stefan J. Johansson, "A Study on Human like Characteristics in Real Time Strategy Games", DOI: 10.1109/itw.2010.5593362, IEEE Conference on Computational Intelligence and Games (CIG), 2010.