Approaches to design human-like characters for fighting games

Risa Nishizawa s1260182 Supervised by Prof. Maxim Mozgovoy

Abstract

Fighting games are primarily one-on-one battles between human players and AI players, or human players and human players.

Almost all of the AI in fighting games exhibit a pronounced behavior pattern. If players become accustomed to this behavior pattern, they may become bored with the game. Several studies have been conducted on fighting game AI.

In this thesis, I surveyed these studies and investigate in detail the AI methods that are currently mainly used in fighting games, divided into five major genres, starting with the most commonly used ones. Then I summarize the literature review to make game AI more human-like.

The purpose of this research is to deepen knowledge of the various methods of fighting game AI and to derive appropriate methods to make the game AI more human-like.

1 Introduction

A fighting game is a one-on-one game in which a character attacks the other character with various techniques such as punches and kicks. Each character has a set health, and the duration of the game is set from the beginning. The winner is the one who reduces the opponent's health to zero by attacking him or loses more health within the time limit. In a versus human game, the behavioral patterns of the opponent's character will vary from player to player, and in many cases, even the same opponent will behave differently in each game. As a result, there is a lot of adaptation between players as they try to decide how to fight each other and when to use their moves.

However, when playing against an AI, unlike a human opponent, the current AI acts based on information such as the opponent's status and timing, but due to the fact that it is scripted in advance [12][13], there is little variation in its behavior patterns. Therefore, there is no bargaining when fighting against the AI, and the problem is that players get bored when they learn the AI's behavior patterns.

There are various researches on fighting game AI. For example, there are those that predict the opponent's behavior and attack, those that aim to achieve a high win rate, and those that aim to keep players from getting bored by changing the strategies and behaviors of the AI's opponents and environment according to the player's skill level. There is also a lot of research related to the Fighting Game AI Competition held during the IEEE Conference on Games (CoG). This is a competition to develop an AI to control a 2D fighting game "FightingICE" and have the AI fight each other to win or lose.

In this thesis, I will investigate the methods and results of the AIs that participated in the Fighting Game AI Competition and investigate the literature on fighting game AI in order to find out what methods should be used to make AI in fighting games more human-like in order to solve the problems just mentioned.

2 Method

2.1 Classification of game AI methods

I have surveyed several papers written about research related to fighting game AI. I then divided the game AI methods used in those studies into four main categories. These are Monte Carlo tree search, dynamic scripting, k-neighborhood algorithms, and neural networks. Among them, I searched for the appropriate method to make the game AI used in fighting games more human-like and investigated those methods in detail.

2.2 Monte Carlo Tree Search

Monte Carlo Tree Search (MCTS) is an algorithm that can efficiently perform tree search by using an evaluation function that adds a Monte Carlo (random) component to the tree search [3] To build the tree, MCTS repeatedly iterates through 4 phases. (Fig.1.)

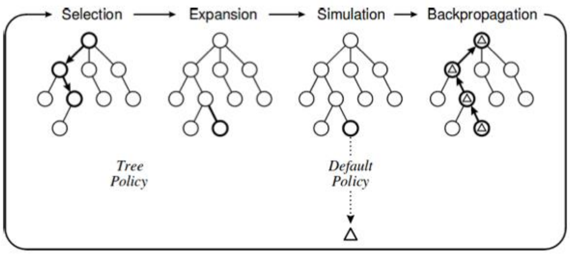

Fig. 1. One iteration of the general MCTS approach

○How to write an English title
The title needs to be capitalized according to the IEEE style.
・Capitalize the first letter of the first and last word and all the nouns, pronouns, adjectives, verbs, adverbs, and subordinating conjunctions (If, Because, That, Which). ・Capitalize abbreviations that are otherwise lowercase (e.g., use DC, not dc or Dc) except for unit abbreviations and acronyms. ・Articles (a, an, the), coordinating conjunctions (and, but, for, or, nor), and most short prepositions are lowercase unless they are the first or last word. ・Prepositions of more than three letters (Before, Through, With, Without, Versus, Among, Under, Between) should be capitalized.

First, I surveyed two theses written about fighting game AI using Monte Carlo tree search [1, 2].

The authors of the literature in [1] state that Monte Carlo tree search in fighting games simulates only a few random actions of the opponent, not all actions. This is because fighting games have a large number of action options and the time available for computation is limited and the AI cannot simulate all actions. Therefore, in their research in [1], they proposed to improve the performance of the AI by predicting the opponent player's behavior and developing a tree based on the prediction instead of random selection. Specifically, the opponent's state is first discretized into 6 patterns (3 patterns of distance between player and opponent, and whether on the ground or in the air). Then, for each of the six patterns, it creates a ranking of the opponent's actions, and use the top five to expand the Monte Carlo tree. Two hundred matches were played between the developed AI and each of the top three AI from the 2016 IEEE CIG Fighting game AI competitions, and the performance of the AI was evaluated based on the difference in HP. The results showed that the developed AI showed 40% wins rate against the winner Thunder01 (2016's Competition 1st) and it showed good results against the rest of AIs (2016's competition 2nd and 3rd). Especially, it showed high performance with 86% wins against the basic Monte Carlo tree search.

In the paper [2], the authors state that varying the difficulty of the game according to the player's skill can lead to more fun and longer playing video games. Therefore, they proposed a DDA (Dynamic Difficulty Adjustment) AI agent that can vary the difficulty level depending on the player to always have a 50% chance of winning against the opponent. When deciding on an action in the Monte Carlo tree search, this AI would not choose the maximum damage, but would choose the action where the damage inflicted on the opponent minus the damage received from the opponent is closest to zero. Also, when building a Monte Carlo tree, the closer the damage inflicted to the opponent minus the damage received from the opponent is to zero, the higher the rating of each node, and the priority is given to expanding its child nodes. They tested the DDA AI agent against a number of bots and all four agents from the 2016 CIG FightingICE competition. They also pitted it against five humans to see if it would perform adequately against human players. The results showed that the DDA AI agents could provide appropriate difficulty levels against a variety of bots and human players.

2.3 Dynamic Scripting

Dynamic scripting is an online competitive machine-learning technique for game AI, that can be characterized as stochastic optimization.

The dynamic-scripting technique is illustrated in Fig. 2 in the context of a commercial game. In the figure, the team dressed in grey is controlled by a human player, while the computer controls the team dressed in black. The rulebase associated with each computer-controlled agent (named ‘A’ and ‘B’ in Fig. 1) contains manually designed rules derived from domain-specific knowledge. It is imperative that the majority of the rules in the rulebase define effective, or at least sensible, agent behavior. [4]

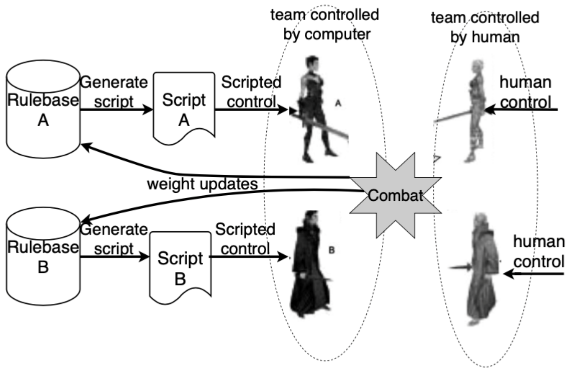

Fig. 2. Dynamic Scripting

However, Dynamic scripting has the disadvantage that it does not work well if the set of rules it is given is not effective.

In the research of [5], with the aim of improving the disadvantage by using Dynamic scripting, an unsupervised learning algorithm that dynamically generates scripts based on a given set of rules, and combining the rules according to their weights to create scripts. During the learning process, they discovered inefficient rules in the rule set and replaced them with newly generated rules. Specifically, rules with weights below a certain level were considered inefficient and replaced with randomly generated if-then rules. As a result, they were able to improve the performance, which was the disadvantage, but the slow convergence was an issue.

The research in [6] also aimed to improve the disadvantage. As in [5], inefficient rules in the rule set are replaced by newly generated rules, and the new rule to be added is one selected from multiple generated rules. Three different ways of selecting new rules were tested: MOS (selecting the rule that is similar to the rule with the highest weight), LES (selecting the rule that is least similar to the discarded rule), and MIX (selecting the rule that is similar to the rule with the highest weight and not similar to the discarded rule). It is The results show that LES has the best performance in

○How to write an English title
The title needs to be capitalized according to the IEEE style.
・Capitalize the first letter of the first and last word and all the nouns, pronouns, adjectives, verbs, adverbs, and subordinating conjunctions (If, Because, That, Which). ・Capitalize abbreviations that are otherwise lowercase (e.g., use DC, not dc or Dc) except for unit abbreviations and acronyms. ・Articles (a, an, the), coordinating conjunctions (and, but, for, or, nor), and most short prepositions are lowercase unless they are the first or last word. ・Prepositions of more than three letters (Before, Through, With, Without, Versus, Among, Under, Between) should be capitalized.

terms of win rate under both mediocre and weak initial conditions of the rule base.

2.4 k-Nearest Neighbor Algorithm

k-Nearest Neighbor is a method for class discrimination. The training data is plotted on a vector space, and when unknown data is obtained, any data are taken from it in the order of closest distance, and the class to which the data belongs is estimated by majority vote. The k-Nearest Neighbor algorithm does not have a training phase, so there is no need to train beforehand, and the analysis can be started immediately after preparing the data.

In [7], research was conducted with the objective of creating an AI that can predict the opponent's next action and take counter actions with rule-based AI, which is considered difficult to predict. The method used here is to record the attack actions taken by the opponent along with their relative position from player. Then, for the opponent's current relative position, if there are more than a certain number of attack records in the vicinity, the AI predicts that the opponent will attack. The prediction of the opponent's attack is made by referring to k records that are close to the current relative position of the opponent. The results show that the AI learns as it plays against the opponent, and that it is fully capable in the modeling process.

The research in [8] aimed to adjust the strength of the AI player during a match so that it slightly outperforms the level of its opponents. The method was similar to the study in [7], where the opponent's past attack actions were recorded along with their relative positions, and the next attack action of the opponent was predicted based on the information in the vicinity of k, the current relative coordinate. In addition to that, they simulated the state of the opponent one second later based on the predicted opponent's attack behavior and tried to select an action that would be moderately strong. As a result, this suggestion was effective.

2.5 Neural Network

A neural network is a mathematical model of a neural network in the form of artificial neurons. It consists of an input layer, an output layer, and a hidden layer, and between the layers there is a weight "W" that indicates the strength of the connections between the neurons. There are many methods in what is called machine learning, and one of them is the method using neural networks. And one method of machine learning that combines neural networks in multiple layers to enhance the representation and learning capabilities is deep learning.

In [9], research was conducted to ensure that the strength of the AI player does not become too strong and that there is diversity in the actions taken by the AI player. This is because an AI that simply aims to become stronger will tend to converge and take the same actions, and the player will get bored. The author proposed a method to adjust the strength of the AI to match the user's skill and to improve the behavioral diversity of the AI by adjusting the teacher signal used for training the neural network according to the player's score. The results showed that the proposed method was effective in improving behavioral diversity in many situations, and that the ratio of scores converged to a defined threshold, indicating that flexible difficulty adjustment is possible.

2.6 Deep Q- learning Network

The Deep Q- learning Network (DQN) network model consists of two convolutional layers followed by two fully connected layers. There are four channels in the input layer. Each channel takes as input one of the consecutive frames. The output dimension is equal to the number of actions of the fighting game agent. In [10], the Deep Q- learning Network, which has been successfully used in Atari Games and Visual Doom AI contests, was used with the goal of beating Monte Carlo tree search based agents [11]. In a fighting game, each character performs a total of 41 actions. There are more input patterns in fighting games compared to Atari games where DQN was tested. Therefore, redundant actions were removed to reduce the number of actions to 11, AI input was limited to 4 frames of screen information, and damage inflicted was maximized assuming the opponent was a player who did not take any action. The results showed that the Deep Q Networks approach is feasible for two-player fighting games. However, none of the experiments in this study compared game AI using Deep Q- learning Networks with game AI using Monte Carlo tree search.

3 Discussion

3.1 Effectiveness of each method

There are many other methods, some of which have not been investigated in this thesis. In this section, I summarize the effects of each of the methods described in the previous sections of this work.

Based on the research on AI using Monte Carlo tree search in [1][2], I think that Monte Carlo tree search is suitable for adjusting the strength of AI according to the objective, since it can achieve strong AI that shows a high win rate or match the level of the player.

From the research on AI using dynamic scripting [5][6], I think that dynamic scripting can create AI players that are stronger the more they play against

○How to write an English title
The title needs to be capitalized according to the IEEE style.
・Capitalize the first letter of the first and last word and all the nouns, pronouns, adjectives, verbs, adverbs, and subordinating conjunctions (If, Because, That, Which). ・Capitalize abbreviations that are otherwise lowercase (e.g., use DC, not dc or Dc) except for unit abbreviations and acronyms. ・Articles (a, an, the), coordinating conjunctions (and, but, for, or, nor), and most short prepositions are lowercase unless they are the first or last word. ・Prepositions of more than three letters (Before, Through, With, Without, Versus, Among, Under, Between) should be capitalized.

each other, because the script is updated to become stronger as the AI players play against each other repeatedly.

From research on AI using the k-nearest Neighbor Algorithm [7][8], this method can predict the opponent's attack by recording the opponent's attack behavior while playing against each other. Therefore, I think that it can make AI players stronger more efficiently.

From the research on AI using neural networks [9][10], the learning function of neural networks can read the skill and strength of the player and adjust the strength of the AI player. It also made it possible to improve the diversity of behaviors to keep the player from getting bored.

3.2 Human like game AI

There are many differences between humans and machines: the resulting behavior, the mindset that created that behavior, the emotions during play, and the various mistakes that humans make.

I think that the simplest way to create a human-like game AI is to learn from human play data. In fact, most of the research on human-like transitions uses supervised learning. Studies such as [1][2] using Monte Carlo tree search and [9] using neural networks can learn from the behavior of human players and adjust the strength of the AI player. I also think that since these methods learn from the behavior of human players, they could mimic the behavior of actual humans. And I think that by learning and imitating human behavior, the resulting behavior and human mistakes can be reproduced. In fact, in the research of [2] introduced in the previous chapter, it was stated that behavioral tables were introduced to represent the playing patterns of opponents. It was then stated that it was suggested that the bots could perform similarly to humans when playing against human players. However, this experiment has only anecdotal value because the number of games is quite small.

In order to create a human-like game AI, the emotions during play and the way of thinking that produced the behavior should be used in a different way. I was not able to investigate methods of incorporating emotions and thinking into AI this time, so I would like to investigate this as future work.

Conclusion and Future work

In this thesis, I introduced the AI methods that are currently mainly used in fighting games. I also considered what methods would be suitable for making fighting game AI more human-like. In this investigation, I considered that the ability to change the difficulty level for each human player by predicting the opponent's behavior during a match in Monte Carlo tree search, and the use of neural networks to improve diversity in behavior, which tends to be biased as well as strength, are suitable for humanizing game AI. I recognize that the methods used in fighting game AI and the focus on human-like behavior are insufficient for this study of human player behavior.

As future work, I would like to deepen my knowledge by investigating methods that I was not able to investigate this time, and by conducting research on the human-like nature of AI. Among the human-like aspects, I would also like to proceed with the investigation of methods regarding emotions during play and the way of thinking when acting

Acknowledgement

I would like to thank Prof. Maxim Mozgovoy for supervising my research. I would also like to thank all the members of my lab and my professor for the useful discussions and presentations.

References

[1] Man-Je Kim, and Kyung-Joong Kim, “Opponent Modeling based on Action Table for MCTS-based Fighting Game AI”, Department of Computer Science and Engineering Sejong University, Seoul, South Korea. 2017 IEEE Conference on Computational Intelligence and Games (CIG), 2017, pp. 178-180

[2] Simon Demediuk, Marco Tamassia, William L. Raffe, Fabio Zambetta, Xiaodong Li and Florian Floyd Mueller, “Monte Carlo Tree Search Based Algorithms for Dynamic Difficulty Adjustment”, School of Media and Communication. Royal Melbourne Institute of Technology RMIT, Melbourne, Australia. 2017 IEEE Conference on Computational Intelligence and Games (CIG), 2017, pp. 53-59,

[3] Remi Coulom. “Efficient selectivity and backup operators in monte-carlo tree search.” International conference on computers and games, pages 72–83. Springer, 2006

[4] Pieter Spronck, Marc Ponsen, Ida Sprinkhuizen-Kuyper, Eric Postma, “Adaptive game AI with dynamic scripting “ machine learning 63, pages 217–248 (2006)

[5] Ruck Thawonmas, Syota Osaka, “A Method for Online Adaptation of Computer-Game AI Rulebase”, roceedings of the International

○How to write an English title

The title needs to be capitalized according to the IEEE style.

・Capitalize the first letter of the first and last word and all the nouns, pronouns, adjectives, verbs, adverbs, and subordinating conjunctions (If, Because, That, Which). ・Capitalize abbreviations that are otherwise lowercase (e.g., use DC, not dc or Dc) except for unit abbreviations and acronyms. ・Articles (a, an, the), coordinating conjunctions (and, but, for, or, nor), and most short prepositions are lowercase unless they are the first or last word. ・Prepositions of more than three letters (Before, Through, With, Without, Versus, Among, Under, Between) should be capitalized.

Conference on Advances in Computer Entertainment Technology, ACE 2006, Hollywood, California, USA, June 14-16, 2006

[6] Syota Osaka, Ruck Thawonmas, Tomoya Shibazaki, “Investigation of Various Online Adaptation Methods of Computer-Game AI Rulebase in Dynamic Scripting”, Simulator, Game. (2007).

[7] T. Kristo and N. U. Maulidevi, "Deduction of fighting game countermeasures using Neuroevolution of Augmenting Topologies," 2016 International Conference on Data and Software Engineering (ICoDSE), 2016, pp. 1-6, doi:10.1109/ICODSE.2016.7936127.

[8] Y. Nakagawa, K. Yamamoto and R. Thawonmas, "Online adjustment of the AI's strength in a fighting game using the k-nearest neighbor algorithm and a game simulator," 2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE), 2014, pp. 494-495, doi: 10.1109/GCCE.2014.7031274.

[9] 中川明紀, 逢坂翔太, 柴崎智哉, “ニューラルネットワークによる格闘ゲーム AI の難易度調節及び行動多様性向上法”, 全国大会講演論文集, vol. 70, pp. 801-802, 2008-03-13

[10] S. Yoon and K. Kim, "Deep Q networks for visual fighting game AI," 2017 IEEE Conference on Computational Intelligence and Games (CIG), 2017, pp. 306-308, doi: 10.1109/CIG.2017.8080451.

[11] Michal Kempka, Marek Wydmuch, Grzegorz Runc, Jakub Toczek & Wojciech Jaskowski, Poznan University of Technology, Poznan, Poland, “ViZDoom: A Doom-based AI Research Platform for Visual Reinforcement Learning.”

[12] 松浦健一郎,”3D 格闘ゲームプログラミング”, ソフトバンククリエイティ ブ,2007/06.

[13] David M.Bourg,Glenn Seemann,”ゲーム開発者のための AI 入門”, オーム 社,2005/01.

○How to write an English title
The title needs to be capitalized according to the IEEE style.
・Capitalize the first letter of the first and last word and all the nouns, pronouns, adjectives, verbs, adverbs, and subordinating conjunctions (If, Because, That, Which). ・Capitalize abbreviations that are otherwise lowercase (e.g., use DC, not dc or Dc) except for unit abbreviations and acronyms. ・Articles (a, an, the), coordinating conjunctions (and, but, for, or, nor), and most short prepositions are lowercase unless they are the first or last word. ・Prepositions of more than three letters (Before, Through, With, Without, Versus, Among, Under, Between) should be capitalized.