Real-Time Imitation Based Learning for Commercial Fighting Games

Sarayut Lueangrueangroj
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Payathai Road, Patumwan
Bangkok 10330, Thailand
Tel. +66813762763
Sarayut.L@student.chula.ac.th,
yutic@hotmail.com

Vishnu Kotrajaras
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Payathai Road, Patumwan
Bangkok 10330, Thailand
Tel. +66890212323
Vishnu@cp.eng.chula.ac.th,
ajarntoe@gmail.com

Abstract

Ghost AI is an Artificial Intelligence (AI) for fighting games which is capable of observing and imitating any player's style of play. By using Ghost AI, a player can play the game as if playing against another player. Although Ghost AI can simulate a human player, current research only applies it offline. Therefore, a human player can easily learn to beat Ghost AI during a match. We propose a methodology that allows Ghost AI to learn in real-time. The real-time data is also used to vary the frequency of each imitated action, so that the action can only be executed in situations which favor the AI. Our experimental result shows that the enhanced AI appears more human-like and more intelligent than the unmodified Ghost AI and provides a more satisfying experience for human players.

Keywords

Ghost AI, Imitation Based Learning, Dynamic Scripting, Fighting Game.

1. Introduction

Fighting games have a long history along with the history of video games. Fighting games have gone through numerous improvements over the years, but one aspect remains almost unchanged. It is artificial intelligence. Most fighting games utilize pre-scripted AI. It means that, for similar situations, the AI always behaves in the same manner. Commercial fighting games usually allow players to select various difficulty levels according to their skills. However, each difficulty level is still pre-scripted. Consequently, experience gamers find these games repetitive and boring. Most experienced players therefore opt to play against real humans because real humans provide greater challenges. Human players can imitate any useful move performed by an opponent who uses the same player character. They can learn and adapt their playing style during gameplay. Through trial and error, they can also learn how one move can be more effective than other moves for a given situation and select the best move for that situation or similar situations.

Better internet access and online play allow players to find their human opponents much easier than before. However, online play requires all participants to be connected at the same time. A computer controlled AI will still have to be used when online play is unavailable.

Some interesting technique was proposed in order to obtain a better computer controlled AI. Graepel et al. [1] applied reinforcement learning to a real commercial fighting game. It showed that machine learning techniques had good potential for applications in the domain of commercial games.

Another approach was imitation learning. This technique was used in Tekken 5 Dark Resurrection, for Sony PSP game system. When a player was playing, the game produced an imitation data file for him. Other players could then obtain the file in order to play with the imitative AI. There was a research which explored this technique. Ghost AI system [2] was proposed for commercial fighting games running on emulators. By recording actions carried out by a player in various situations, the system generated a probabilistically weighed case based AI that performed comparable to the player it imitated. Players were able to play with a Ghost AI as if playing with the person who trained that AI.

Since players vary in skill and different players play differently, even with the same character, a ghost AI system can provide players with many different challenges from a single game character. However, the Ghost AI system proposed in current research only learned offline. Therefore, its actions and style of play were only limited to what it learned in past matches. It would keep its play style even though the style was not useful in the match it was playing. This allowed human players to eventually spot its weaknesses and beat it using such weaknesses, rendering the same problem as pre-scripted AI. Human players, however, are able to change their play style during play to remove such weaknesses. In order to learn to fight other players effectively, the AI should not just imitate the players' moves from past matches. It should be able to reason when to (or not to) execute each of its learned move in real-time, just like a human player. If there are alternative moves available for a situation, the AI should learn to execute the move that troubles current opponent the most.

This paper proposes a technique to improve an imitation based AI for fighting games. By applying dynamic scripting [3] [4], an efficient unsupervised learning algorithm for a fighting game character is introduced. Our AI agent learns and improves itself from how its opponent plays the game in real-time. The AI observes, imitates its opponent's actions and adjusts how it executes its learned moves in real-time. This leads to a more challenging and more realistic imitation based AI opponent.

2. Testbed: AI-TEM Framework

Our proposed AI operated under AI-TEM (AI-Testbed in Emulator [5]), an efficient testbed environment we created for testing game AI. AI-TEM was created from Visual Boy Advance (VBA), an open source emulator for Nintendo Game Boy Advance (GBA). A commercial Game Boy Advance game ROM could be played on AI-TEM and used for testing AI. By using and controlling values stored in the memory of the emulator, game states and game behaviors could be controlled. Testing on actual commercial games guaranteed the quality of any developed AI. For this paper, we tested our imitation based AI on Street Fighter Zero 3 Upper, which was one of the well known fighting games on the system. The image of the game is shown in Figure 1.

Figure 1. Street Fighter Zero 3 Upper

3. Online Imitation Based AI

3.1 Overview

Our imitation based AI could be broken up into two parts. The first part handled imitation. The AI observed how a player responded to each situation and created cases from the observed data. A case contained the details of a game state and actions the player performed in that state. The AI used cases as its scripts. Therefore it played the game using the same style as the player it recorded its data from. The other part of the AI handled learning. In ordinary Ghost AI, the AI utilized each imitated action with the same frequency as played by its trainer. However, our system worked differently. When an action was executed, our system checked whether the action contributed to the fight score. The probability for the action to be executed in the same situation altered according to the observed fight score. Therefore the AI learned to utilize the play style it had imitated in a way that was most beneficial to itself.

3.2 Data in a Case

A case contained details of a game state, as viewed by the AI. A game state contains:

- The distance between the AI character and its opponent on the x-axis and the y-axis.
- The distance between the AI character and a projectile weapon (if there was any on the screen) on the x-axis and the y-axis.
- AI state.
- The list of actions a human player performed in response to that game state.

Possible values for each parameter are shown in Table 1. For the list of actions performed by the human player, each action stored:

- How often the human player executed the action in that situation.
- The action's weight, which was used in our learning mechanism.

The contents of each list of actions are displayed in Table 2.

Table 1. Game state details

Details	Value	
Delta X	1-9	
Delta Y	1-4	
Delta WX	1-3	
Delta WY	1-4	
AI State	1-5	
Player Actions List [0-76]	Freq	Weight

Table 2. List of actions and values in a case

Actions	Frequency	Weight
Action[0]	x	x
Action[1]	x	x
Action[2]	x	x
…	x	x
…	x	x
…	x	x
Action[76]	x	x

3.3 Imitation Process

When playing against a human player, whenever the player performed an action, the AI determined whether there was a matching case. Parameter values in each state were grouped into ranges in order to allow similar cases to be matched. The list of the player's actions was not used in this matching.

If the case did not exist in the database, the AI would add the case to the database using the game state details. The player's action would be recorded with an initial frequency (the value was 1) and an initial weight (the value was 0). If the case already existed, the AI would check whether the player's action was recorded in the list of actions for that case. If the action was in the list, the AI would then increase the frequency of that action by one. Otherwise, the AI would add that action to the list with the initial frequency and the initial weight. The imitation process is described in Figure 2.

3.4 Learning Process

During gameplay, when a character was dealt damage, the AI updated the weight of the action that created damage in that situation. Since our imitation cases were created when a human player performed actions and any actions the AI could perform came from those cases, it was certain that a case existed for the damage situation. When a character was damaged, it was either the AI character or the player character. If the AI character was damaged, the weight of the action that the AI character performed would decrease. At the same time, the weight of the action performed by the player character would increase. Similar updates took place when the player character took damage. This process is shown in Figure 3.

3.3 Using Cases to Play the Game

When playing against a human opponent, the AI would check if the opponent's state matched the situation details in one of our cases. The AI would check the followings:

- The distance between the player and the AI in the x-axis and the y-axis.
- The distance between the player and a projectile weapon on the x-axis and the y-axis.
- It also checked whether the player's state matched with the AI's state in the case.

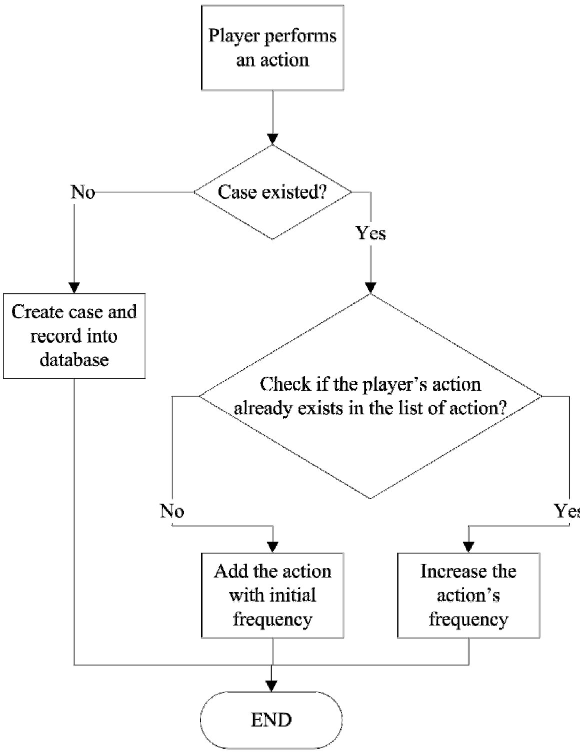

Figure 2. Imitation process

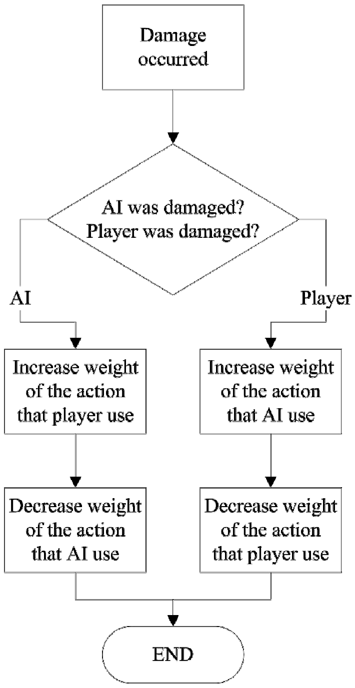

Figure 3. Learning process

If all values matched, it meant the player behaved in the same way as the AI when that case was created. The AI would then be able to choose an action from the player's action list to perform in order to imitate the player. If there was no match, the AI would perform a random action in order to create more chances for a matching case. Any random action that damaged the human-control character would not be used to update the weight because we only wanted to adjust actions obtained from imitation.

A player's action had its frequency and weight. The frequency reflected the imitation aspect and the weight reflected the learning aspect. To choose an action from the list of actions from a case, these two values had to be considered.

Action would be chosen based on probability. This process of choosing an action could be divided into two parts.

- First, the AI used the frequency value of each action to calculate a percentage value for each action. This percentage value was the probability for each action to be chosen based on imitation alone. The frequency based probability was represented by P_f.
- Second, the AI considered the weight value of each action and used it to calculate a percentage value for each action. This percentage value was the probability for each action based on the learned success rate of the action. This probability was represented by P_w.

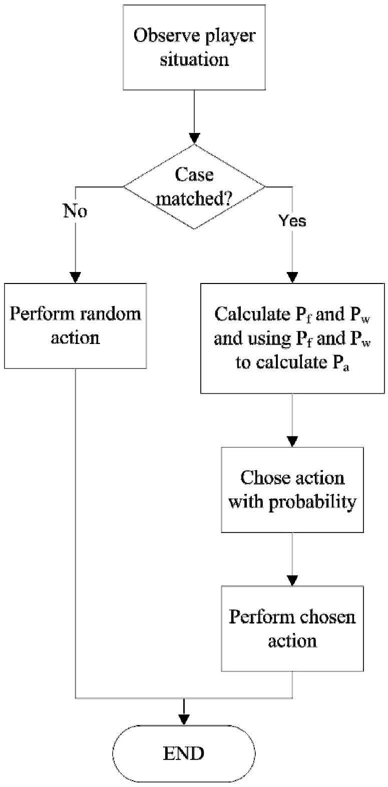

Figure 4. Choosing cases to play the game

The AI then computed an adjusted probability value for each action using P_f as a base value and using P_w to adjust P_f. The adjusted probability for an action was represented by P_a. P_a would be calculated using equation (1).

$$P_a = P_f + (C \times (P_f - P_w)) \quad (1)$$

Where C is a constant value (which is 0.1 in our experiment).

P_a would be used as a probability value for choosing each action in the list. The process of choosing cases is illustrated in Figure 4.

4. Experiment

We conducted an experiment to compare the performance between the Ghost AI (using the code for our AI, but performed only imitation, without learning) and our AI. In this experiment, we tested our AI with 9 human players; each player played against the AI for 10 matches, each match took 3 rounds to win.

First, we had human players play against the Ghost AI. We recorded damage data that each player did to the AI and damage data that AI did to the player for each round. The result is shown in Figure 5. Due to the reason of space, only the result from one person is shown. But all results were similar.

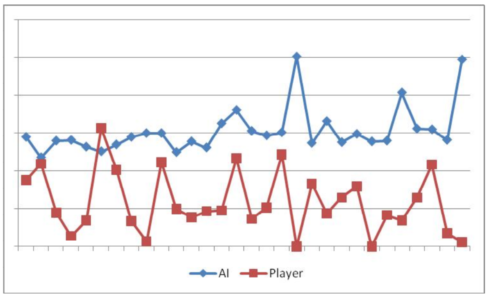

Figure 5. Ghost AI - Player

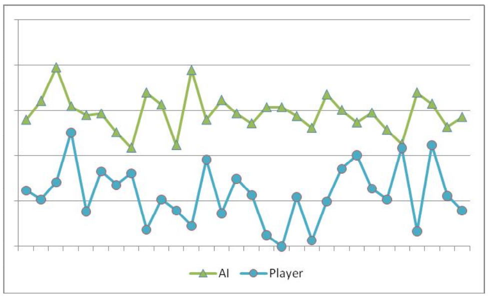

Figure 6. Imitation based learning AI - Player

Then, the experiment was conducted in the same way but human players played against our AI instead. The result of this experiment (for the same person from figure 5) is shown in Figure 6. The line referred to as AI line stands for the amount of damage AI received from a player for each round and the line referred to as Player line stands for the amount of damage a player received from the AI for each round.

4.1 Discussion

Experimental results would be considered in separate parts. The first part's result showed that for most players the amount of damage the Ghost AI received from the players increased over time. This implied that the players learned faster than the ordinary Ghost AI. The amount of damage the players received from the AI increased for a period of time, then suddenly decreased, then increased for another period of time. This was probably caused by the AI imitating players, learning to perform players' moves until players adjust their plays (represented by sudden decrease in damage caused by the AI). The players could exploit weakness from the Ghost AI by adjusting his style.

For the second part of the experiment, we could see that for most players the amount of damage that AI received from them was quite invariant, only slightly increasing at the end. This indicated that the human player still played better, but it was more difficult to see. The amount of damage that players received from the AI appeared to swing between increase and decrease period and appeared to be an invariant pattern similar to the AI. This was probably caused by the AI learning from players, performing better moves until players adjust their plays, similar to how the Ghost AI performed.

From the collected damage results alone, it could be concluded that our imitation based AI could perform online learning in real time, similar to the online Ghost AI. But human players still learned much faster. This caused the result from our AI to appear slightly better than the result from the Ghost AI.

Table 3. Players' opinions on which AI was cleverer and more human-like

Player	Ghost AI (GAI) / Imitation Based Learning AI (IMBL AI)
Player 1	GAI
Player 2	IMBL AI
Player 3	GAI
Player 4	IMBL AI
Player 5	IMBL AI
Player 6	IMBL AI
Player 7	IMBL AI
Player 8	IMBL AI
Player 9	IMBL AI

Besides the damage data, players who tested the AI had given their opinion on how both AI performed. Most players agreed that the imitation based learning AI could perform more clever maneuvers than the pure Ghost AI. They were satisfied with the performance of the imitation based AI. This information is shown in Table 3. From the graphs and players' opinions, our experimental result showed that the imitation based learning AI was able to learn and adapt itself to suit the game situation in real-time. The imitation based AI could perform cleverer moves and appeared more intelligent and human-like than the ordinary Ghost AI.

Conclusion

In this paper, we introduced a new technique for creating an AI in fighting game using Imitation based learning method. This technique had improved the Ghost AI technique, creating an AI that not only simulated the player style of play but also utilized the observed style in an efficient way to favor itself. The experimental results showed that our AI was able to improve itself overtime during gameplay. It also provided more realistic challenges for human players. In our future work, we intend to improve this AI further by introducing a feature for the AI to counter a player's maneuver.

References

[1] Graepel, T. Herbrich, R. and Gold, J. Learning to Fight. *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*. 2004.

[2] Thunputtarakul, W. and Kotrajaras, V. Data Analysis for Ghost AI Creation in Commercial Fighting Games. *8th International Conference on Intelligence Games and Simulation (GAME-ON 2007)*, Bologna, Italy.

[3] Spronck, P. Sprinkhuizen-Kuyper, I. and Postma, E. Online Adaptation of Game Opponent AI with Dynamic Scripting. *International Journal of Intelligent Games and Simulation (eds. N.E. Gough and Q.H. Mehdi), March/April 2004, Vol.3, No.1, ISSN: 1477-2043, pp. 45-53*. University of Wolverhampton and EUROSIS.

[4] Spronck, P. Sprinkhuizen-Kuyper, I. and Postma, E. Enhancing the Performance of Dynamic Scripting in Computer Games. *Entertainment Computing - ICEC 2004 (ed. Matthias Rauterberg), Lecture Notes in Computer Science 3166, pp. 296-307*. Springer-Verlag.

[5] Thunputtarakul, W. and Kotrajaras, V. AI-TEM: Testing AI in Commercial Fighting Game With Emulator. *International Conference on Computer Games: Artificial Intelligence and Mobile Systems*, 24-27 July 2006, Lousiville, Kentucky, USA.