Fighting Game Artificial Intelligence Competition Platform

Feiyu Lu, Kaito Yamamoto, Luis H. Nomura, Syunsuke Mizuno, YoungMin Lee, and Ruck Thawonmas

Intelligent Computer Entertainment Laboratory, Ritsumeikan University

ruck@ci.ritsumei.ac.jp

***Abstract*—Game playing has provided an interesting framework for developing and testing artificial intelligence (AI) algorithms, with well-known examples such as Go and Chess. Since the first chess programs were written, there has been steady progress in the level of play to the point where current systems can challenge top-class human players. Recently, academia groups have created competitions to evaluate and compare AI methods, such as Ms Pac-Man versus Ghost Team and Student StarCraft AI Tournament. Those tournaments define their rules and goals based on the game they selected, according to which intelligent techniques and technologies are developed. We focus on a genre called fighting game. Examples of fighting games are Street Fighter, Tekken, and Mortal Kombat. To select the winner in the fighting game is simple: the winner is the character with less damage or the last one standing. In this paper we introduce a brand new competition[1] based on a 2D fighting game written in Java.**

Keywords—artificial intelligence; AI; fighting game; contest; algorithms

I. VIDEO GAMES AND FIGHTING GAMES

From the earliest known interactive electronic game was born in 1948 to today, video games are gaining more popularity. Nowadays, nearly every person is aware of video games, and the majority of them have played one at least once [1]. There are many types of video games, such as role playing games, real-time strategy games, racing games. Yet another one is a Fighting Game that is a common genre for both consumer and arcade games. In a typical fighting game, the player controls his or her character that represents them until a predetermined level of energy is expended [2].

II. ARTIFICIAL INTELLIGENCE IN VIDEO GAMES AND FIGHTING GAMES

Because video games represent controlled environments that can be created to mimic real world environments, they are ideal testing spaces for ideas that may be difficult or dangerous when tested in reality. It is for this reason that video games are widely used to test different types of artificial intelligence used for controlling a non-player character (henceforth, simply called an AI) [1]. To make computer games or AIs more intelligent, various methods, such as finite state machines, fuzzy state machines, artificial life, and neural networks, have been studied and applied.

For example, in the neural network approach, it is crucial to decide when a neural network is trained and which values are used for training. The key element in this approach is the novel structure of the information processing system composed of a large number of highly interconnected processing elements (neurons) working together to solve specific problems. Examples of games which have employed neural network techniques are board games, such as Five-in-a-Row, Tic-Tac-Toe, and Go [4].

However, it is challenging to apply the aforementioned methods to AIs in fighting games. In fighting games, devising overall strategies is still important. However, it is more important to consider how to provide intelligence to an AI in order to take proper actions based on the opponent's positions and actions.

Most fighting games utilize pre-scripted AIs. It means that, for similar situations, such an AI always behaves in the same manner. On the other hand, human players 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.

A non-scripted approach is imitation learning. This technique was used in Tekken 5 Dark Resurrection, for the Sony PSP game system. When a player is playing, the game produces an imitation data file for him. Another system called ghost AI[2] was proposed for commercial fighting games running on emulators. This system has more complex behaviors than just imitating the player's moves from his or her past matches. However, such systems cannot reason when to (or not to) execute each of its learned moves in real-time while fighting against a new opponent, unlike a human player [3]. Work by Heon et al. [4] exists which designs a neural network for this task.

[1] http://www.ice.ci.ritsumei.ac.jp/~ftgaic/

[2] http://www.tekkenzaibatsu.com/forums/ghostlist.php

978-1-4799-0892-9/13/$31.00 ©2013 IEEE

III. GAME AI COMPETITIONS

Game AI competitions have been a part of several international conferences related to artificial intelligence and computational intelligence techniques. For example, IEEE Conference on Computational Intelligence in Games (IEEE CIG) has hosted a number of interesting competitions since 2005[3]. In 2005, the competitions covered Iterated Prisoners Dilemma, Go (7 x 7), and Computer Controlled Car Racing. For many years, such events have evolved to include a lot of different games with new challenges [6]. Below is a list of game AI competitions, in alphabetical order, conducted in 2013, including the proposed fighting game competition.

TABLE I. GAME AI COMPETITIONS IN 2013

Competition	Website
AIIDE StarCraft	http://webdocs.cs.ualberta.ca/~cdavid/starcraftaicomp/
Angry Birds	http://aibirds.org/
CIG StarCraft	http://ls11-www.cs.uni-dortmund.de/rts-competition/start
Fighting Game	http://www.ice.ci.ritsumei.ac.jp/~ftgaic/
Physical Travelling Salesman	http://www.ptsp-game.net/
Platformer	http://platformersai.com/

IV. MOTIVATIONS

A. Encourage students to work in computational intelligence

Games competitions provide frameworks to teach students technologies related to computational intelligence and allow them to enhance such technologies. This framework is a tool for students to test their codes versus other students' codes or default AI codes. Competent challengers should implement the best algorithm and technique to survive in their competitions.

B. Develop real time AI algorithms

For AI developers, fighting game competitions offer many challenges such as energy management, opponent analysis, the utilization of effective tactics and strategies, etc. They can also use competition platforms to evaluate the intelligence and adaptive behavior of their AIs.

C. Compare and test algorithms and techniques

Other challengers have a chance to test the intelligence of their characters against another non-player character (NPC) or human controlled character, comparing which technique is more effective.

D. Strong or Human-like AIs

A solution to this dilemma is the development of AIs that do not rely primarily on classical AI methods, such as scripting, triggers and animations. Game competitions provide an ideal test bed for practitioners to further the development of AIs that play a game intelligently [3] like human players.

V. ORIGINALITY OF THE PROPOSED PLATFORM

Our platform can support Java programing to develop several AIs. This platform also allows designing AIs flexibly: the developer can create an AI using several intelligent algorithms. This means that AI designers can make unique attacks, combos and movements, etc., depending on the algorithms in use; AI can also learn action patterns from its opponent.

The platform was designed to send delayed game information to each AI, and they will use the same interface, as human players, to input their commands to the game. We introduce the delay of 15 frames in order to give more risks to the defense play style and simulate the reaction delay of human players.

Before our engine was developed, there had been some popular fighting game engines, such as, M.U.G.E.N and zero2d. M.U.G.E.N is a 2D fighting game engine designed to be used by people with little or no programming experience. For this reason, it is hard to design a non-rule-based AI on it. As a result, we decided to develop our new game engine.

VI. ARCHITECTURE AND GAME SYSTEM

As shown in Fig. 1, an AI can get its character's information from the system: current hit point energy; position on the screen; speed vector; attack hit box (called hit box in other games); hit box (called boundary box in other games); current action state (stand, in air, crouch or down); current action type (base, move, guard, recovery, and skill) , etc.

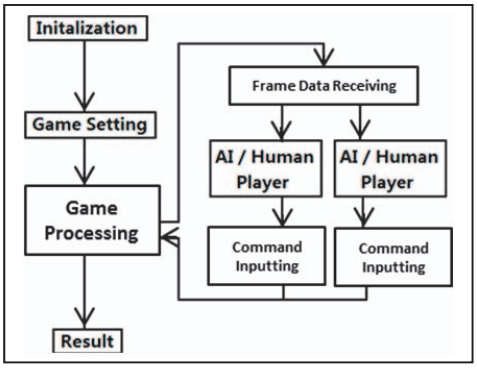

Fig. 1. Game engine's architecture and the processing flow.

[3] http://www.ieee-cig.org/

In this game, the smallest time unit is 1 frame. This means every action can be counted by this. For example, the movement speed will be 10pixel per frame, a light punch can hit the opponent after 8 frames after the player inputs the command; or a complex skill must be input in 4 frames. One second is 60 frames, so the response time is 16.66ms/frame.

In other words, if a command is waited, an AI must process all what it has to do within this 16.66ms and input the command to the game; otherwise, the character will not do anything. For the game system, most characters have the same basic move and defense action like jump, run or dash. Depending on their types, they might have unique attack and skill. In the current platform, however, there is only one character type provided.

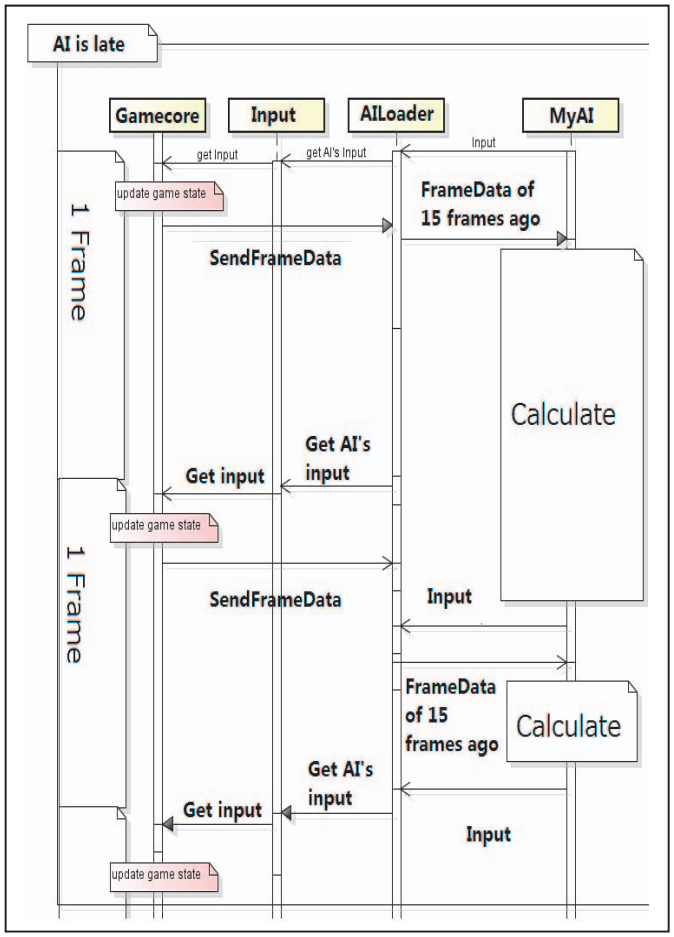

Fig. 2. Time flow of each major class in the case where the AI controller's processing is late.

Figures 2 and 3 show two different cases of the time flow of each major class. In Fig. 2's case, after the AI controller (MyAI) receives a frame data, it does not input a command within the first frame, but does this in the subsequent frame. Therefore this command will be processed in the end of the second frame, unless there are other newer commands, causing a one-frame delay in the response. Figure 3 show a case where such a delay does not occur.

As a result, in our platform, it is important to note that if the AI wants to react on time, it must always input a command based on the current frame data to the game before the current frame is over.

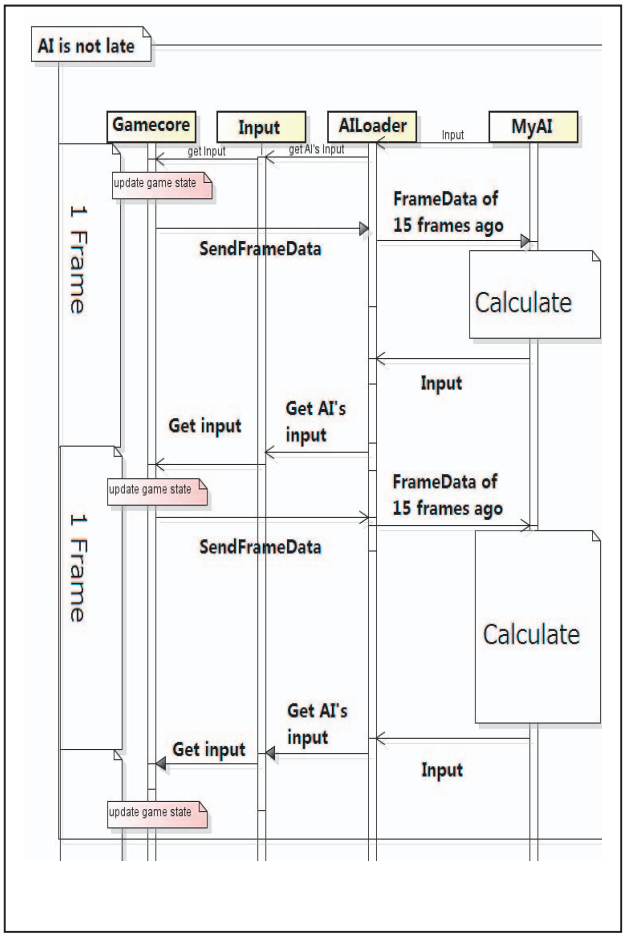

Fig. 3. Time flow of each major class in the case where the AI controller's processing is not late.

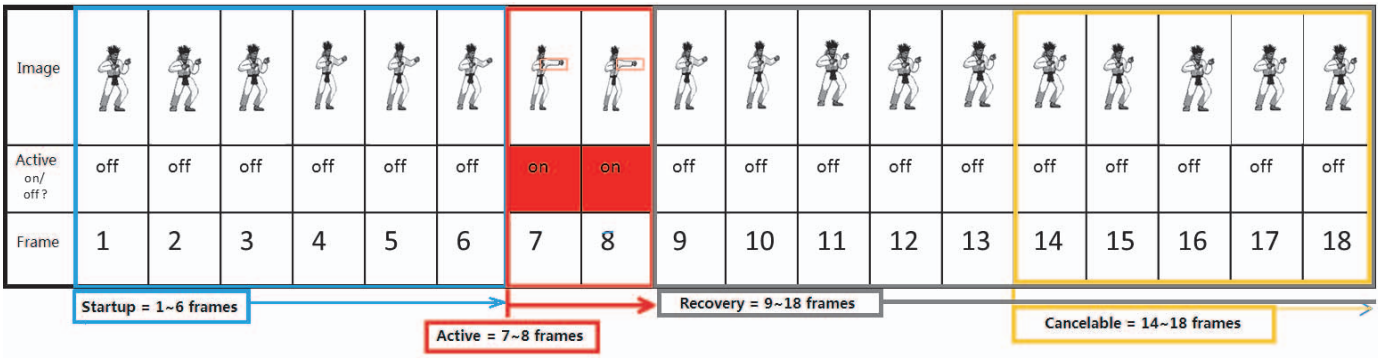

Fig. 4. Example of a skill's three stages.

A skill has three stages as shown in Fig. 4, i.e., startup, active, and recovery. Note that some skills' recovery might have a cancelable period.

1st - Startup: After a given command is confirmed, the corresponding skill will be used. During this period, any other action cannot be used; the character will start to move, but still cannot make any damage to its opponent.

2nd - Active: These are the frames where an "attack hit box" (the red square in the first row of the above figure) appears. If the opponent character's "hit box" coincides the "attack hit box" of your character, the opponent character will be damaged. Note that in other games, the hit box is sometimes called the bounding box.

3rd - Recovery: The attack hit box disappears, and the character turns back to its normal status. In this example, there is a cancelable period, in which if there is a skill, say skill A that can cancel the current skill, skill A can be used.

A cancelable frame means that if there is a skill can cancel the current skill's recovery time, it can be used during the cancelable frames.

VII. Competition Rules

This contest is a fighting game match to find out a high skilled AI controller by using our game engine. It is a league match, and one game has 3 rounds. The fighting time of a round is 60 seconds; after that time, the characters' positions will be reset, and a new round is started. After the game, the score will be calculated as

$$\frac{\text{Opponent Character's HP}}{\text{Player Character's HP} + \text{Opponent Character's HP}} \times 100$$

The player that makes the highest score in total will win the match.

VIII. Conclusions and Future Work

In this paper, we described our platform for a fighting game AI competition. Since August 1, 2013, we have started the 2013 competition, whose midterm submission deadline is August 27, 2013 and the final one is September 25, 2013. One type of sample character and three sample AIs are provided.

In future, we plan to design more sample characters and AIs, and run a series of competitions at related international conferences, such as the 2014 IEEE World Congress on Computational Intelligence (IEEE WCCI 2014) in Beijing, China, July 6 - 11, 2014 and the IEEE Conference on Computational Intelligence in Games (CIG 2014) in Dortmund, Germany, August 26 - 29, 2014. At GCCE 2013, we will report the results of the 2013 competition.

References

[1] Aaron Elkin, "Adaptive Game AI and Video Game Enjoyability," http://cs.union.edu/~rieffelj/classes/497-Finals/Elkin.pdf (2012)

[2] Duck Gun Park and Hee Chan Kim, "Wireless input device for a fighting action game based on the EMG signal and acceleration of the human forearm," http://www.intuinno.com/uploads/1/0/2/9/10297987/muscleman_paper.pdf

[3] Sarayut Lueangrueangroj and Vishnu Kotrajaras, "Real-time imitation based learning for commercial fighting games," Proc. of Computer Games, Multimedia and Allied Technology 09, International Conference and Industry Symposium on Computer Games, Animation, Multimedia, IPTV, Edutainment and IT Security, Amara Hotel, Singapore, pp. 1-3, 2009.

[4] B. Heon, S. Hoon, Y. Rak, and H. Ryoung, "Exploiting intelligence in fighting actions games using neural networks." IEICE TRANS. INF. & SYST., Vol. E89–D, No. 3, pp.1249 – 1256, 2006.

[5] P. Rohlfshagen and S. M. Lucas, "Ms Pac-Man Versus Ghost Team CEC 2011 Competition," Proc. of the 2011 IEEE Congress on Evolutionary Computation, pp. 70-77, 2011.

[6] Kyung-Joong Kim and Sung-Bae Cho, "Game AI Competitions: An Open Platform for Computational Intelligence Education," Computational Intelligence Magazine, IEEE, Vol. 8, No. 3, pp. 64-68, 2013.