Optimization of Enemy’s Behavior in Super Mario Bros Game Using Fuzzy Sugeno Model

Nanang Ismail¹, Husnul R M¹, M Nurul Subkhi², Miftahudin¹, Eki A Z Hamidi¹
¹Electrical Engineering Departement, UIN Sunan Gunung Djati Bandung, Indonesia
²Physics Departement, UIN Sunan Gunung Djati, Bandung, Indonesia
* E-mail: nanang.is@uinsgd.ac.id

Abstract. One of the key factors to make the action game more interesting is the smart behavior of an enemy. This paper describes the enhancement of smart enemy’s behavior in the game that similar to Super Mario Bros using Sugeno Fuzzy Model. Sugeno Fuzzy Model is applied to regulate the enemy’s behavior and determine the steps to be taken. The enhancement of the enemy’s behaviors performed in the process of chasing down moving player and enemy’s behaviors when meeting the player. Tests on the chasing player process, performed with 4 scenarios and shows achievement of 75% success, while in the game before adding Fuzzy Sugeno Model, the success is only 20%. Meanwhile, testing for enemy’s behavior in this game is 100% successful, while in previous games only 33.3%. An enemy behavior logic test is performed to obtain defuzzification results as a Cartesian axis graph. Testing is done using Matlab®. Testing yielded 3 Cartesian axis graphs. Each graph has a behavioral decision. Therefore, there are three enemy’s behavior decisions in this game, while Super Mario Bros game only produces 1 graph of Cartesian axis that does not result in enemy’s behavior decisions.

1. Introduction

Video games take a lot of developments from simple concepts to complex technology products. One of the most attractive fields to be a research until now is artificial intelligence in games. This technology has been successfully implemented in various types of games, one of them is a strategy game [1].

Action games or the strategy games apply artificial intelligence systems in determining attitudes and actions of "enemies" in the game. A smart enemy due to implementing an artificial intelligence system (AI), will give the challenges to players [2]. Generally, the users of the game want the challenge and experience of playing which is better. Therefore, it is needed to continue the development of research on the creation of non-player characters with unpredictable behaviors [3].

Research on the development of games by applying various methods of artificial intelligence had been done by Cole, et al by implementing a genetic algorithm for regulating and developing Bot’s behavior pattern, so it becomes more intelligent [4]. Its artificial intelligence is applied to “the non-player enemy” in the game against player strategy in Counter-strike games. Increased intelligence of "enemy" characters in a game can also help to complete the journey on the map [5]. References [5] use the genetic algorithm in the Lemmings game, and the result shows that the use of genetic algorithms can solve complex paths on the map. Fuzzy logic has also been used in improving the "enemy" intelligence in a Pac-Man game, so that the "enemy" in this game acts as a smart agent [6].

This paper describes the optimization of the enemy’s behavior in an action game. The action game is a kind of game that emphasizes physical challenges and it needs fast reaction with coordination
between eyes and hands. The chosen game concept as a comparator is Super Mario Bros Game. This game features a plumber named Mario who has to save the princess of the Mushroom Kingdom who was abducted by a King of Koopas named Bowser. The gameplay is displayed with platforming levels theme, where the player must avoid the different enemies and obstacles at each level. Based on the same concept and story with Super Mario Bros Game, then a new game is developed which will optimize the enemy’s behavior as well as establish the steps will be taken using the fuzzy logic of Sugeno’s method. All of these approaches will be implemented using Phaser framework as a media of game creation based on the web [7]. The Phaser framework is an open source platform and directly integrated with the web. In addition, with Phaser, games can be developed into a mobile game. Phaser also allows the user to create an online game without having to download or install additional applications [8]. Furthermore, this game is named Chase and Conquer Game (ChaCon Game).

2. Method
The steps of research development activities refer to the following diagram.

![Diagram](image)

Figure 1. The steps of development

ChaCon Game is designed with 1 player and 3 enemies. The character of player is shown in Figure 2 and the character of enemies are shown in Figure 3.

![Figure 2](image)

Figure 2. Player.

![Figure 3](image)

Figure 3. Enemy.

2.1. Implementation of Sugeno Fuzzy Model
The fuzzy logic implementation of ChaCon Game is divided into 2 categories, they are fuzzy logic for enemy chasing/following the player, and for “enemy’s behavior”.

“Enemy chasing/following player” is a setting of the enemy to find the location of the player, then to follow it anywhere. This command is important for artificial intelligence because an enemy can move like a player which is controlled by a human. The “enemy chasing/following player” command is created using the fuzzy Sugeno rules presented in Table 1.
Table 1. Fuzzy rules for “enemy chasing/following player”.

| Distance       | Velocity | Decision-Making |
|----------------|----------|-----------------|
| 1. IF Enemy X < Player X AND 25 px THEN | Move to right |
| 2. IF Enemy X > Player X AND -25 px THEN | Move to left  |
| 3. IF Enemy Y > Player Y AND 250 px THEN | Jump          |

In Fuzzy for Enemy’s Behavior Implementation, “enemy’s behavior” will be regulated with fuzzy logic Sugeno model, either when the health condition is fit or injured. The pattern of “enemy’s behavior” that has a health point in a fit condition will be different from “enemy’s behavior” that has a health point in injured. The pattern of “enemy’s behavior” is presented in Table 2.

Table 2. Fuzzy rules for enemy’s behavior.

| Distance | Health | Decision-Making |
|----------|--------|-----------------|
| 1. IF Near AND Fit THEN | Attack |
| 2. IF Near AND Injured THEN | Avoid  |
| 3. IF Far AND Fit THEN | Attack |
| 4. IF Far AND Injured THEN | Avoid  |

Meanwhile, Fuzzy for Enemy Chasing at Super Mario Bros Game uses the rule as in Table 3.

Table 3. Fuzzy Rules for enemy chasing in Super Mario Bros game.

| Distance       | Velocity | Decision Making |
|----------------|----------|-----------------|
| 1. IF Enemy X > Player X AND -20 px THEN | Move to left |

Afterward, the fuzzy logic for “enemy’s behavior” in Super Mario Bros Game has a function as an attacking command from an enemy to the player. The “enemy’s behavior” command in Super Mario Bros game is shown in table 4.

Table 4. Fuzzy Rules for “enemy’s behavior” in Super Mario Bros game

| Distance | Decision Making |
|----------|-----------------|
| 1 IF Near THEN | Attack |
| 2 IF Far THEN | Attack |

Based on Table 4, the enemy’s behavior is not distinguished by health condition, since there is only the condition which is represented by values 1 and 0. The value of 1 means alive, the value of 0 is dead.

3. Result and Analysis

ChaCon Game involves 1 player and 3 enemies. The number of enemies are not created as much as Super Mario Bros game, for this reason, the analysis is focused on “enemy’s behavior” that is not associated with the number of enemies.

3.1. Game Interface

The game display visually is shown in Figure 4.
3.2. Testing
There are two parameters will be tested, they are “enemy chasing” and “enemy’s behavior”. The parameter “enemy chasing” is tested by the black-box method. Meanwhile, the testing of “enemy’s behavior” uses Matlab software to ensure defuzzification values.

3.2.1. Black-Box Testing. The fuzzy testing results for “enemy chasing/following player” are found in Table 5.

| The Test Scenario | Expected results | Conclusion on ChaCon Game | Conclusion on Super Mario Bros game |
|-------------------|-----------------|-----------------------------|------------------------------------|
| 1. When the player is on the right side | Enemy moves along with the player | Successful | Unsuccessful |
| 2. When the player is on the left side | Enemy moves along with the player | Successful | Successful |
| 3. When the player is on top of the left side/right | Enemy jumps | Successful | Unsuccessful |
| 4. When the player is on top/bottom | Enemy moves along with the player | Unsuccessful | Unsuccessful |

Meanwhile, the fuzzy testing results for the parameter “enemy’s behavior” are listed in Table 6.
### Table 6. The fuzzy testing for “enemy’s behavior” parameter.

| The Test Scenario                  | Expected results | Conclusion on ChaCon Game | Conclusion on Super Mario Bros game |
|------------------------------------|------------------|---------------------------|------------------------------------|
| 1. When the player is on the right and close to the enemy | Enemy attacks to right | Successful                 | Unsuccessful                        |
| 2. When the player is on the left and close to the enemy | Enemy attacks to left       | Successful                 | Successful                           |
| 3. When enemy’s health 50%         | Enemy jumps to avoid player’s attack | Successful                 | Unsuccessful                        |

#### 3.2.2. Testing with Matlab. The next step is to test a fuzzy algorithm using the Matlab on the ChaCon game and it is compared with the Super Mario Bros game. There are several steps carried out for testing using this Matlab, they are fuzzification step, entering the fuzzy input variable and creating the defuzzification output.

Fuzzification is the process of mapping a numerical value into a fuzzy set and determining the degree of membership. Fuzzification on Chacon Game has 3 inputs, namely distance, health, and avoid.

The distance variable of ChaCon Game is divided into 2 sets, namely “near” and “far”. The range of value for distance variable between 0-1000 in accordance with map length, with the criteria presented in equations (1) and (2),

\[
\mu[\text{Near}] = \begin{cases} 
0; & x \geq 700 \\
1; & x \leq 300 \\
\frac{700-x}{700-300}; & 300 < x \leq 700 
\end{cases} \quad (1)
\]

\[
\mu[\text{Far}] = \begin{cases} 
0; & x \leq 300 \\
1; & x \geq 700 \\
\frac{x-300}{700-300}; & 300 < x \leq 700 
\end{cases} \quad (2)
\]

The result of calculation, then it is presented in a graph form as shown in Figure 5.

![Figure 5. Input value of distance variable on ChaCon Game.](image)

The health variable on ChaCon Game is divided into 2 sets, namely, fit and injured. The range of value for health variable between 0-200 in accordance with health input on the program, with the criteria presented in equations (3) and (4),
\[\mu[Injured] = \begin{cases} 
0; & x \geq 140 \\
1; & x \leq 60 \\
\frac{140-x}{140-60}; & 60 < x \leq 140 
\end{cases}\] (3)

\[\mu[Fit] = \begin{cases} 
0; & x \leq 60 \\
1; & x \geq 140 \\
\frac{x-60}{140-60}; & 60 < x \leq 140 
\end{cases}\] (4)

Then, the result of calculation is presented in graph form as shown in Figure 6.

![Figure 6](image)

**Figure 6.** An Input value of health variable on ChaCon Game.

The avoid variable on ChaCon Game has a range of value 60 – 140, with the criteria presented in equation (5),

\[\mu[Avoid] = \begin{cases} 
0; & x \leq 60 \text{ or } x \geq 140 \\
\frac{x-60}{100-60}; & 60 < x \leq 100 \\
\frac{140-x}{140-100}; & 100 \leq x < 140 
\end{cases}\] (5)

The result of the calculation is presented in graph form as shown in Figure 7.

![Figure 7](image)

**Figure 7.** Input value of avoid variable on ChaCon Game.
The input variables of the Super Mario Bros game are distance and health. The distance variable of an enemy on Super Mario Bros is divided into 2 sets, they are ‘near’ and ‘far’. The range of value for distance variable between 0 - 1000 in accordance with map length will be explained as follows:

1. Near = 0 - 700
2. Far = 300 - 1000

Manually calculated fuzzification values for near and far functions are performed using equations (2) and (3). The result of the calculation is presented in graph form as shown in Figure 8.

![Figure 8. Input value of distance variable on Super Mario Bros game.](image1)

The Health variable on Super Mario Bros game is dead. The range of value for health variable is between 0 – 1. The dead curve is formulated by the following equation (6),

\[
\mu[Dead] = \begin{cases} 
0; & x = 0 \\
1; & x = 1 \\
\frac{x-0}{1-0}; & 0 < x \leq 1 
\end{cases} \tag{6} 
\]

The result of the calculation is presented in graph form as shown in Figure 9.

![Figure 9. Input variable of health on Super Mario Bros game.](image2)

After all input variables are entered, then the defuzzification process is done to get the output result of ChaCon game as well as Super Mario Bros game. In ChaCon Game, the results obtained are as follows:

- IF distance = 0-500 AND health = 100-200 THEN decision = 0
• IF distance = 0-500 AND health = 0-100 THEN decision = avoid
• IF distance = 500-1000 AND health = 100-200 THEN decision = 0
• IF distance = 500-1000 AND health = 0-100 THEN decision = avoid

In Super Mario Bros game, the results obtained from Matlab output are as follows:
• IF distance 0-500 AND health 0-1 THEN decision = 0
• IF distance 500-1000 AND health 0-1 THEN decision = 0

3.3. Analysis
Based on testing of the parameter “enemy chasing”, it can be concluded that there is only one failed test scenario in ChaCon game, it is when the player is on the top/bottom of the player. When the player position is on the top/bottom of the enemy, then the enemy cannot follow the player, for this reason, in the design there is no input of the command for the condition. While on testing for “enemy chasing” in Super Mario Bros game, there is only one successful test scenario that is when the player is on the left. When the player is on the left, the enemy will follow the player to the left. The testing for other positions, the enemy failed to follow the player’s movement. This failed due to fuzzy input for parameter “enemy chasing” in Super Mario Bros game is only available when the player is only on the left.

Meanwhile, the testing for parameter “enemy’s behavior” in ChaCon game shows that all test scenarios are successful, for this reason, all the fuzzy inputs entered in the game do not have an error. While the testing for parameter “enemy’s behavior” in Super Mario Bros game, it is only one successful test scenario when the player is on the left. When the player is on the left, the enemy will attack the player because the fuzzy input for “enemy’s behavior” in Super Mario Bros game does not provide input to avoid.

The testing applies Matlab for parameter “enemy’s behavior” will be reviewed using a cartesian axis graph to determine the relation between input-output of the Fuzzy Sugeno method. Based on the defuzzification result, this ChaCon game produces 3 input-output relations, they are the relation between inputs of health and distance, the relation between inputs of avoid and distance, and the relation between inputs of avoid and health.

The “enemy’s behavior” decision by entering the value of health = 0-200 (Z-axis), and the distance value = 0 - 1000 (X-axis), produce the decision value of the enemy = 1 (Y-axis). The input-output relationship for health and distance variables in ChaCon game can be seen in Figure 10.

![Figure 10](image-url)

Figure 10. The relation of input-output for health and distance variables on ChaCon Game.
The decision of “enemy’s behavior” by entering the value of avoid = 60 – 140 (Z-axis), and distance value = 0 – 1000 (X-axis), produce the decision value of the enemy = 0.5 (Y-axis). The input-output relationship for avoid and distance variables in ChaCon Game can be seen in Figure 11.

![Figure 11](image1.png)

**Figure 11.** The relation of input-output for avoid and distance variables on ChaCon Game.

The decision of “enemy’s behavior” by entering the value of avoid = 60 – 140 (Z-axis), and health value = 0 – 200 (X-axis), produces the decision value of the enemy = 1 (Y-axis). The relation of input-output for avoid and health variables on ChaCon Game can be seen in Figure 12.

![Figure 12](image2.png)

**Figure 12.** The relation of input-output for avoid and health variables in ChaCon Game.

The defuzzification results in Super Mario Bros game produce 1 graph of the Cartesian axis, that is the relationship between input of health and distance. The “enemy’s behavior” decision by entering the value of health = 0 - 1 (Z-axis), and the distance value = 0 - 1000 (X-axis), produce the decision value of the enemy = 0.5 (Y-axis). The relation of input-output for health and distance variables in Super Mario Bros game can be seen in Figure 13.

![Figure 13](image3.png)
Figure 13. The relation of input-output for health and distance variables in Super Mario Bros game.

Based on the input-output relation graph in both games, it can be concluded that ChaCon Game produces the decision value = 2.5 and Super Mario Bros game produces the decision value = 0.5. These results prove that artificial intelligence on ChaCon Game is better than Super Mario Bros game, for this reason, the ChaCon Game produces a new value that is an input of avoid.

4. Conclusion
Based on the research which is conducted, testing of the “enemy chasing” of the player on ChaCon Game, which is carried out with 4 scenarios, it shows the success of 75%. Meanwhile, in Super Mario Bros game, the success only reaches 20%. Then, the testing for “enemy’s behavior” on ChaCon Game shows the success of 100%, while in Super Mario Bros game only reaches 33.3%. The testing for “enemy’s behavior” using Matlab on ChaCon Game produces 3 cartesian axis graphs with the decision value = 2.5, while in Super Mario Bros game only produces 1 graph of the Cartesian axis with the decision value = 0.5. The results show that artificial intelligence on ChaCon Game is better than Super Mario Bros, and it has an impact on enemy’s behavior in facing the player.

References

[1] M. A. Akbar, A. Muttaqin and E. M. A. J, “Designing Word Slash Games Using Kinect as A Game's Control Sensor With Hand Movement (In Bahasa Indonesia),” PTIK DORO, 2013.

[2] A. Pamungkas, E. P. Widiyanto and R. Angreni, “Application of Algorithm A * In The Android-Based Maze Island Educational Game (In Bahasa Indonesia),” STMIK MDP, 2014.

[3] A. M. Mora, J. J. Merelo, P. García-Sánchez, P. A. Castillo, M. S. Rodríguez-Domingo and R. M. Hidalgo-Bermúdez, “Creating autonomous agents for playing Super Mario Bros game by means of evolutionary finite state machines,” Evolutionary Intelligence, vol. 6, no. 4, pp. 205-218, 2014.

[4] S. J. L. a. C. M. N. Cole, “Using a genetic algorithm to tune first-person shooter bots,” in in Evolutionary Computation, 2004. CEC2004, Portland, 2004.

[5] G. Kendall and K. Spoerer, “Scripting the game of Lemmings with a genetic algorithm,” in Evolutionary Computation, 2004. CEC2004. Congress on, Portland, 2004.

[6] A. Shaout, B. W. King and L. A. Reisner, “Real-Time Game Design of Pac-Man Using Fuzzy Logic,” Int. Arab J. Inf. Technol, vol. 3, no. 4, pp. 315-325., 2006.
[7] J. Pranata, K. R. Purba and Liliana, “Strategy Game Application Using Genetic Algorithm Methods for Making Troops (in Bahasa Indonesia),” *Jurnal Infra*, vol. 4, no. 1, pp. 199-204, 2016.

[8] A. Katmada, “Towards Building An Affective Educational Game Framework: A Proof of Concept,” Aristotle University of Thessaloniki, Thessaloniki, 2015.