

Implementasi Logika *Fuzzy* Untuk Mengatur Perilaku Musuh dalam *Game* Bertipe *Action-RPG*

Kristo Radion Purba, Rini Nur Hasanah dan M. Azis Muslim

Abstrak - Dalam penelitian ini, dirancang penerapan logika *fuzzy* Sugeno untuk mengatur perilaku musuh, pada *game* (permainan) bertipe *Action-RPG* yang berjudul "*Song of Ruination 2*". Logika *fuzzy* adalah cabang dari sistem kecerdasan buatan (*artificial intelegent*) yang mengemulasi kemampuan manusia dalam berfikir ke dalam bentuk algoritma yang kemudian dijalankan oleh mesin. Setiap musuh dan pemain memiliki status kekuatan, terdiri dari *attack* (Kekuatan serang), *defense* (Ketahanan), *speed* (Kecepatan gerak), *agility* (Kemampuan menghindari) dan *life* (Nyawa). Terdapat 3 jenis musuh, yaitu penyerang, pemanah dan boss. Musuh penyerang akan menyerang dari jarak dekat dengan pedang. Musuh ini memiliki parameter input *fuzzy* yaitu nyawa dan jarak, dia akan mundur ketika nyawanya lemah atau jaraknya dekat dengan pemain. Musuh pemanah memiliki parameter input *fuzzy* yaitu amunisi dan jarak, dia akan mundur ketika didekati pemain atau ketika amunisinya sedikit. Musuh boss memiliki parameter input *fuzzy* yaitu nyawa dan jarak, ia sangat agresif, ia mundur ketika nyawanya sangat sedikit. Penerapan *fuzzy* dalam permainan ini telah berjalan dengan baik, dimana musuh tipe penyerang menjadi cukup agresif (45% perilaku menyerang), tipe pemanah agresif jika berada pada jarak jauh (49% perilaku menyerang), dan musuh boss sangat agresif (89% perilaku menyerang).

Kata kunci - Logika *fuzzy* sugeno, *action RPG*, kecerdasan buatan.

I. PENDAHULUAN

PERMAINAN komputer (*computer game*), atau yang lebih dikenal sebagai permainan/*game* adalah aplikasi hiburan dengan sejumlah standar aturan untuk mengatur perkembangan sesi, yang memiliki nilai kuantitatif tentang kesuksesan dan kegagalan saat pemain memainkannya [1]. Terdapat 4 elemen utama dalam permainan, yaitu representasi, interaksi, konflik dan keamanan [2].

Action Game (Permainan Aksi) adalah jenis permainan yang menekankan pada tantangan fisik, yang membutuhkan reaksi cepat dengan koordinasi antara mata dan tangan [3]. *RPG* (*Role-playing Game*) adalah permainan yang para pemainnya memainkan peran

tokoh-tokoh khayalan dan berkolaborasi untuk merajut sebuah cerita bersama [4]. Permainan *action-RPG* merupakan gabungan dari keduanya, dimana permainan ini menekankan pada tantangan fisik, dimana gerakan karakter pemain terjadi terus menerus sesuai dengan kontrol yang diberikan pemain, dan para pemainnya merupakan tokoh khayalan.

Pada perangkat lunak permainan yang dirancang pada penelitian ini diimplementasikan logika *fuzzy* untuk mengatur perilaku musuh. Logika *fuzzy* adalah cabang dari sistem kecerdasan buatan (*artificial intelegent*) yang mengemulasi kemampuan manusia dalam berfikir ke dalam bentuk algoritma yang kemudian dijalankan oleh mesin [5].

Penerapan logika *fuzzy* pada permainan ini adalah untuk mengatur perilaku musuh, secara singkatnya, jika musuh sedang dalam keadaan lemah, dia akan melarikan diri jika sudah lemah hampir mati. Selain itu, faktor yang menentukan perilaku adalah jarak musuh dengan pemain, dan jumlah amunisi yang dimiliki musuh. Penerapan logika *fuzzy* pada permainan ini menghasilkan *output* yaitu musuh yang memiliki kepintaran dalam bertindak..

Algoritma untuk logika *fuzzy* pada penelitian ini dikembangkan dari [6], yaitu perancangan *fuzzy* untuk mengatur pergerakan musuh dalam game Pacman menggunakan *Fuzzy IF-THEN rule*. Rincian rule yang digunakan untuk menentukan musuh akan *hunting* (menyerang *pacman*) :

- If (pacman_near AND skill_good) then hunting_behavior
- If (pacman_near AND skill_med AND pellet_med) then hunting_behavior
- If (pacman_near AND skill_med AND pellet_long) then hunting_behavior
- If (pacman_med AND skill_good AND pellet_long) then hunting_behavior
- If (pacman_med AND skill_med AND pellet_long) then hunting_behavior
- If (pacman_far AND skill_good AND pellet_long) then hunting_behavior

Pada penelitian lain [7], dirancang sebuah permainan menembak dengan 2 jenis musuh, yaitu *scout* dan *sniper*, dimana perilakunya diatur dengan logika *fuzzy* sugeno *zero order*. Pada penelitian ini digunakan *Fuzzy Sugeno* karena model ini menghasilkan *output* berupa konstanta tegas, yang dapat mewakili nilai perilaku yang sudah didesain sebelumnya. Untuk desain *Fuzzy*

Kristo Radion Purba adalah mahasiswa Program Magister Fakultas Teknik Elektro, Universitas Brawijaya Malang (08123577791, duke_dreadmoore@yahoo.com)

Rini Nur Hasanah adalah dosen jurusan Teknik Elektro Universitas Brawijaya Malang (rini.hasanah@ub.ac.id)

M. Azis Muslim adalah dosen jurusan Teknik Elektro Universitas Brawijaya Malang (muh_aziz@ub.ac.id).

pada NPC Scout, dua variabel yang mempengaruhi perilaku adalah jumlah amunisi dan kesehatan. Jika amunisi sedikit dan kesehatan lemah, NPC cenderung melarikan diri, demikian sebaliknya. Untuk desain *Fuzzy* pada NPC Sniper, dua variabel yang mempengaruhi perilaku adalah jumlah amunisi dan jarak musuh. Jika amunisi sedikit dan jarak semakin dekat, NPC cenderung melarikan diri, demikian sebaliknya, jika amunisi semakin banyak dan jarak semakin jauh, musuh akan semakin menyerang.

II. METODE IMPLEMENTASI

Pada bagian ini, akan dibahas tentang sistem fuzzy yang dipakai, perancangan musuh dan perlengkapan yang ada dalam permainan, dan juga bagaimana penerapan fuzzy pada permainan.

A. Logika Fuzzy Sugeno

Logika *fuzzy* yang pertama kali diperkenalkan oleh Lotfi A. Zadeh, memiliki derajat keanggotaan dalam rentang 0(nol) hingga 1(satu), berbeda dengan logika digital yang hanya memiliki dua nilai yaitu 1(satu) atau 0(nol). Proses *fuzzy inference* dapat dibagi dalam lima bagian [8], yaitu :

- Fuzzyfikasi Input : FIS mengambil masukan-masukan dan menentukan derajat keanggotaannya dalam semua *fuzzy set*.
- Operasi logika fuzzy :Hasil akhir dari operasi ini adalah derajat kebenaran *antecedent* yang berupa bilangan tunggal.
- Implikasi :Merupakan proses mendapatkan *consequent* atau keluaran sebuah IF-THEN rule berdasarkan derajat kebenaran *antecedent*. Proses ini menggunakan mengambil nilai MIN/terkecil dari dua bilangan : Hasil operasi *fuzzy logic OR* dan *fuzzy set* banyak.
- Agregasi :Yaitu proses mengkombinasikan keluaran semua IF-THEN rule menjadi sebuah *fuzzy set* tunggal. Pada dasarnya agregasi adalah operasi *fuzzy logic OR* dengan masukannya adalah semua *fuzzy set*.
- Defuzzyfikasi : Keluaran dari defuzzyfikasi adalah sebuah bilangan tunggal, cara mendapatkannya ada beberapa versi, yaitu *centroid*, *bisector*, *middle of maximum*, *largest of maximum* dan *smallest of maximum*

Fuzzy Model Sugeno merupakan varian dari model Mamdani dan memiliki bentuk aturan seperti :

$$\text{IF } x_1 \text{ is } A_1 \text{ AND.. } x_n \text{ is } A_n \text{ THEN } y = f(x_1, x_2.. x_n) \quad (1)$$

Dimana *x* merupakan parameter input, *A* merupakan nilai dari parameter, *f* merupakan sembarang fungsi dari variabel-variabel masukan yang nilainya berada dalam interval variabel keluaran.

A. Status Kekuatan Pemain/Musuh

Dalam permainan "*Song of Ruination 2*" yang akan dirancang, status karakter/musuh yang akan digunakan adalah :

- *attack* (Kekuatan Serang) /*atk*

Merupakan angka yang akan mengurangi *life* (nyawa) musuh ketika si penyerang menyerang. Angka maksimalnya 99.

- *defense* (Ketahanan) /*def*

Merupakan angka persentase yang akan mengurangi besarnya *attack* yang masuk saat musuh menyerang. Angka maksimalnya 99.

- *life* (Nyawa)

Merupakan nyawa yang dimiliki individu. Jika nyawa habis, individu ini akan mati. Angka maksimalnya adalah 999.

- *speed* (Kecepatan gerak) /*spd*

Merupakan angka pixel/frame yang merepresentasikan kecepatan gerak individu. Angka maksimalnya 99.

- *agility* (kelincahan) /*agi*

Merupakan angka persentase yang menandakan berapa peluang musuh dapat menghindari dari serangan yang masuk. Angka maksimalnya 99.

- *ammo* (amunisi)

Jumlah peluru yang dimiliki oleh musuh pemanah. Angka maksimalnya 49.






- *experience* (pengalaman)

Merupakan angka pengalaman pemain. Semakin tinggi pengalamannya, semakin tinggi level pemain.

B. Jenis Musuh

Dalam permainan yang akan dirancang dalam penelitian ini, pemain berhadapan dengan musuh-musuh berkarakter ksatria kegelapan. Rincian musuh dapat dilihat pada tabel I.

TABEL I.
RINCIAN TIPE MUSUH, NAMA DAN TAMPILANNYA

| <i>Type</i> | <i>Nama</i> | <i>Gambar</i> |
|-------------|---------------------|---|
| Penyerang | <i>Knight</i> |  |
| Penyerang | <i>Paladin</i> |  |
| Pemanah | <i>Cross bowman</i> |  |
| Pemanah | <i>Artemis</i> |  |
| Boss Misi 1 | <i>High-Priest</i> |  |

Terdapat 3 tipe musuh dalam permainan ini yaitu :

- Penyerang
Memiliki sifat agresif. Mereka akan maju sebagai *front-liner* musuh. Membawa senjata pedang untuk menyerang dari jarak dekat.
- Pemanah
Memiliki sifat tidak agresif. Membawa panah jarak jauh untuk menembak dari tempat tersembunyi.

Musuh ini memiliki karakteristik serangan yang kuat, kecepatan gerak tinggi, namun nyawanya sedikit.

▪ Boss

Memiliki sifat sangat agresif. Musuh tipe boss memiliki karakteristik serangan kuat, nyawa banyak, namun kecepatan gerak lambat.

Pada saat awal misi dimulai, program akan membangkitkan musuh-musuh dengan jenis acak pada posisi yang telah ditentukan di lahan. Setiap musuh akan diberikan kontrol agar dapat bergerak. Setelah musuh mati, dalam jeda 30 detik dia akan dibangkitkan kembali dengan jenis acak pada posisi awal tempat ia muncul

C. Perlengkapan Pemain/Musuh

Setiap musuh dan karakter utama nantinya akan memiliki perlengkapan (equipment) tambahan, yang terdiri dari 3 kategori, yaitu *gloves* (Sarung tangan), *armor* (Pelindung badan), *shoes* (Sepatu). Setiap kali memakai perlengkapan tertentu, status kekuatan musuh/pemain akan bertambah/berkurang sesuai dengan perlengkapan yang digunakan. Berikut ini tabel perlengkapan yang ada di permainan ini, pada Tabel II.

TABEL II.

RINCIAN PERLENGKAPAN DAN STATUSNYA

| Nama | atk | def | spd | agi |
|-------------|-----|-----|-----|-----|
| Iron Glove | +10 | +10 | 0 | 0 |
| Steel Glove | +15 | +10 | 0 | 0 |
| Dark Glove | +30 | -10 | 0 | 0 |
| Demon Glove | +40 | -15 | 0 | 0 |
| White Glove | -10 | +30 | 0 | 0 |
| Holy Glove | -15 | +40 | 0 | 0 |
| Iron Armor | 0 | +25 | -5 | 0 |
| Steel Armor | 0 | +35 | -10 | 0 |
| Dark Armor | 0 | -10 | +30 | 0 |
| Demon Armor | 0 | -15 | +40 | 0 |
| White Armor | 0 | +30 | -10 | 0 |
| Holy Armor | 0 | +40 | -15 | 0 |
| Iron Shoes | 0 | 0 | +5 | +10 |
| Steel Shoes | 0 | 0 | +5 | +15 |
| Dark Shoes | 0 | 0 | +30 | -10 |

D. Variabel Penentu Perilaku

Logika *fuzzy* pada game ini akan memberikan perilaku berbeda untuk setiap tipe musuh. Model *fuzzy* yang digunakan adalah Sugeno, dengan mempertimbangkan bahwa output dari *fuzzy* Sugeno merupakan konstanta tegas, sehingga dapat mewakili nilai perilaku yang sudah didesain sebelumnya.

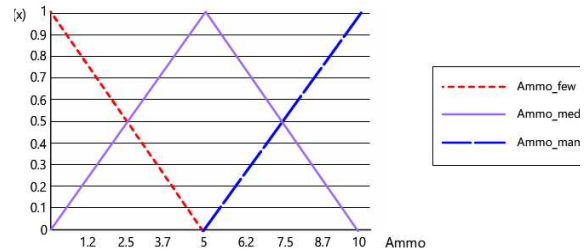
Dalam logika *fuzzy* yang akan dirancang ini, terdapat 3 variabel linguistik sebagai penentu perilaku musuh, yaitu *ammo*, *life* dan *range*. *ammo* adalah jumlah amunisi, yang termasuk sebagai status musuh. *life* adalah nyawa, juga termasuk sebagai status musuh. Sedangkan *range* adalah jarak musuh ke karakter pemain, dalam pixel. Pada bagian ini juga akan dirinci variabel mana saja yang akan dipakai oleh setiap tipe musuh, misalnya musuh tipe penyerang hanya menggunakan 2 variabel input yaitu *range* dan *life*.

Berikut ini 3 variabel penentu tersebut, dan juga fungsi keanggotaannya :

▪ Variabel ammo

Jumlah *ammo* merupakan angka bulat dengan nilai antara 0-10. Jumlah *ammo* musuh bertambah sebanyak 1 setiap 5 detik. Jumlah awal *ammo* adalah 5. Jumlah maksimal *ammo* adalah 10.

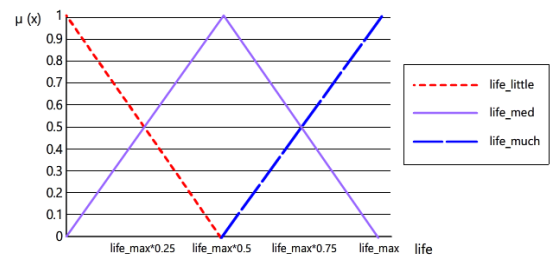
Berikut ini fungsi keanggotaan untuk variabel *ammo* :



Gambar 1. Fungsi keanggotaan ammo

▪ Variabel life

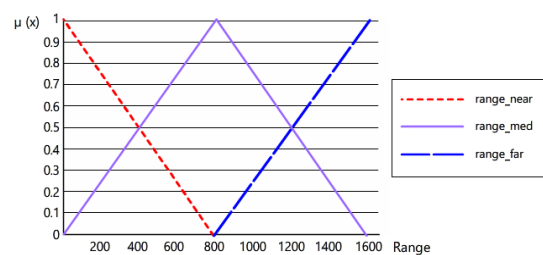
Jumlah *life* merupakan angka bulat. Variabel *life_max* merupakan *life* (nyawa) maksimal yang dimiliki oleh jenis musuh tertentu, mengacu pada tabel 1. Berikut ini adalah fungsi keanggotaannya :



Gambar 2. Fungsi keanggotaan life

▪ Variabel range

Range merupakan jarak pemain terhadap musuh, dalam pixel. Berikut ini adalah fungsi keanggotaan untuk *range* :



Gambar 3. Fungsi keanggotaan range

Nantinya, input variabel penentu perilaku tersebut akan menentukan musuh akan berperilaku apa. Perilaku musuh terbagi dalam 5 kategori yaitu :

- Melarikan Diri (Disingkat : lari)
- Bertahan
- Ragu-ragu (Disingkat : ragu)
- Menyerang
- Menyerang Brutal (Disingkat : brutal)

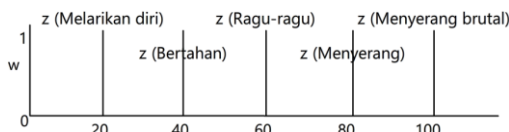
Keseluruhan proses logika *fuzzy* dapat diterangkan sebagai berikut :

- Fuzzyfikasi input : Mencari derajat keanggotaan

dari input fuzzy yang memenuhi fuzzy rule

- Operasi fuzzy : Menggunakan fungsi AND, yaitu mengambil nilai MAX dari derajat keanggotaan setiap input fuzzy.
- Implikasi : Mendapatkan keluaran dari fuzzy rule menggunakan fungsi MIN.
- Agregasi : Mengkombinasikan keluaran semua fuzzy rule menjadi fuzzy set tunggal dengan menggunakan fungsi MAX.
- Hasil fuzzy : Menghasilkan nilai perilaku musuh, antara skala 0-100 dengan metode weighted average. Nilai 0-100 ini nantinya menentukan peluang musuh untuk menyerang pemain.

Diagram *output* untuk perilaku musuh terlihat pada gambar 4 berikut ini :



Gambar 4. Diagram perilaku musuh (FIS Sugeno)

E. Fuzzy Untuk Musuh Penyerang

Berikut ini adalah *fuzzy rule* untuk musuh penyerang :

- IF range_near AND life_few THEN lari
- IF range_near AND life_med THEN bertahan
- IF range_near AND life_much THEN menyerang
- IF range_med AND life_few THEN lari
- IF range_med AND life_med THEN bertahan
- IF range_med AND life_much THEN menyerang
- IF range_far AND life_few THEN bertahan
- IF range_far AND life_med THEN menyerang
- IF range_far AND life_much THEN brutal

F. Fuzzy Untuk Musuh Pemanah

Berikut ini adalah tabel *fuzzy rule* musuh pemanah :

- IF range_near AND ammo_little THEN lari
- IF range_near AND ammo_med THEN lari
- IF range_near AND ammo_many THEN lari
- IF range_med AND ammo_little THEN bertahan
- IF range_med AND ammo_med THEN ragu
- IF range_med AND ammo_many THEN ragu
- IF range_far AND ammo_little THEN menyerang
- IF range_far AND ammo_med THEN menyerang
- IF range_far AND ammo_many THEN menyerang

G. Fuzzy Untuk Musuh Boss

Berikut ini adalah tabel *fuzzy rule* untuk musuh boss :

- IF range_near AND life_few THEN bertahan
- IF range_near AND life_med THEN menyerang
- IF range_near AND life_much THEN brutal
- IF range_med AND life_few THEN bertahan
- IF range_med AND life_med THEN brutal
- IF range_med AND life_much THEN brutal
- IF range_far AND life_few THEN menyerang
- IF range_far AND life_med THEN brutal

- IF range_far AND life_much THEN brutal

H. Proses Fuzzy untuk Nilai Perilaku

Operasi logika *fuzzy* yang digunakan pada penelitian ini adalah menggunakan fungsi AND (fungsi MIN). Proses *fuzzyfikasi* yang digunakan adalah FIS tipe Sugeno berorder nol, karena *fuzzy* Sugeno menghasilkan konstanta tegas, sehingga dapat mewakili nilai perilaku yang telah didesain.

Proses implikasi menggunakan fungsi MIN, dan agregasi menggunakan fungsi MAX. Kemudian hasil *fuzzy* dihitung dengan *weighted average*, yaitu dengan rumus :

$$output = \frac{\sum_{i=1}^N w_i z_i}{\sum_{i=1}^N w_i} \tag{2}$$

dimana :

N = Jumlah *fuzzy rule*

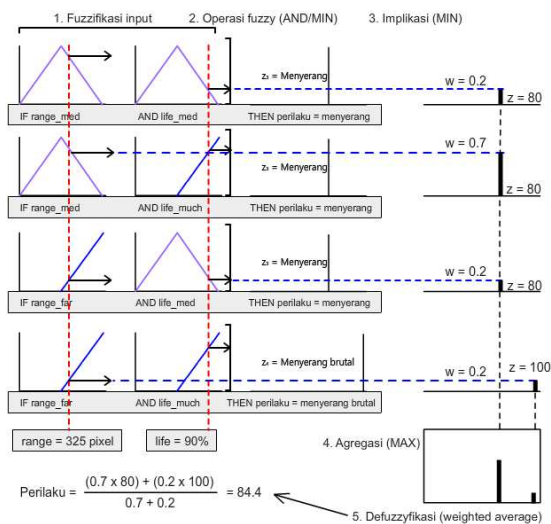
w = Bobot hasil implikasi

z = *Output* perilaku musuh (Mengacu pada gambar 4)

Jika mengambil contoh seorang musuh tipe penyerang, dengan keadaan saat ini ia memiliki *life* 90% dan *range* 325 pixel dari pemain, maka *rule* yang memenuhi adalah :

- IF range_med AND life_med THEN menyerang
- IF range_med AND life_much THEN menyerang
- IF range_far AND life_med THEN menyerang
- IF range_far AND life_much THEN brutal

Berdasarkan contoh yang tertulis diatas, diagram hasil dapat digambarkan pada gambar 5 berikut ini :



Gambar 5. Proses *fuzzy*

Dari diagram proses *fuzzy* pada gambar 5 terlihat bahwa hasil *fuzzy* adalah 84.4, berarti musuh penyerang yang memiliki *range* (jarak ke pemain) 325 pixel dan *life* (nyawa) 90% memiliki peluang 84.4% untuk maju menyerang pemain, dan 15.6% untuk mundur.

III. HASIL DAN PEMBAHASAN

Pada bagian ini akan dibahas tentang hasil penelitian yang telah diterapkan pada sebuah permainan berjudul

"Song of Ruination 2" versi purwarupa, yang hanya terdiri dari 1 level. Hasil yang akan dibahas adalah meliputi spesifikasi sistem, tampilan permainan, penerapan algoritma dan pengujian algoritma.

A. Spesifikasi Sistem

Permainan ini dirancang dan diuji coba pada spesifikasi sistem sebagai berikut :

- Processor Intel Core 2 Duo T6600 2.2 GHz
- Memori 2GB DDR3
- Harddisk 5400 RPM
- Layar berukuran 1366 x 768
- Sistem operasi Windows 8
- Kartu VGA Intel GMA 3100

Berdasarkan pengujian, spesifikasi minimum untuk menjalankan game ini, adalah sebagai berikut :

- Processor Intel P4 atau Centrino (Single Core) 1.5 GHz
- Memori 512MB
- Harddisk 5400 RPM
- Layar berukuran 800 x 600
- Sistem operasi Windows XP atau Linux
- Kartu VGA 32MB

B. Tampilan Karakter

Karakter dalam permainan ini, baik musuh maupun karakter utama, dapat menggunakan perlengkapan. Tampilan karakter akan berubah sesuai dengan warna perlengkapan yang dipakainya.

Karena masing-masing tipe perlengkapan memiliki 6 macam benda, maka kemungkinan perlengkapan yang dipakai seorang karakter adalah $6 \times 6 \times 6 = 216$ kemungkinan. Contoh tampilan karakter menggunakan perlengkapan dapat dilihat pada gambar 6 berikut :



Gambar 6. Contoh tampilan karakter serta perlengkapan yang digunakan

C. Pengujian Logika Fuzzy pada Musuh Penyerang

Pada musuh tipe penyerang, logika *fuzzy* akan mengatur peluang menyerang musuh berdasarkan parameter *life* dan *range* (Jarak antara musuh ke pemain). Secara umum, jika *life* semakin besar, musuh akan semakin agresif, dan sebaliknya. Jika *range* semakin jauh, musuh akan semakin agresif, dan sebaliknya. Pada Tabel III hasil pengujian logika *fuzzy* pada musuh penyerang untuk parameter masukan *range* dan *life* yang berbeda-beda.

TABEL III.
HASIL PENGUJIAN PERILAKU MUSUH PENYERANG

| <i>range/</i> <i>life</i> | 0 | 200 | 400 | 600 | 800 | 1000 | 1200 | 1400 | 1600 | 1800 | 2000 |
|------------------------------|----|-----|-----|-----|-----|------|------|------|------|------|------|
| 0 | 20 | 20 | 20 | 20 | 20 | 25 | 30 | 35 | 40 | 40 | 40 |
| 10 | 24 | 24 | 26 | 24 | 24 | 34 | 38 | 43 | 48 | 48 | 48 |
| 20 | 28 | 28 | 29 | 28 | 28 | 38 | 44 | 49 | 56 | 56 | 56 |
| 30 | 32 | 32 | 31 | 32 | 32 | 42 | 49 | 55 | 64 | 64 | 64 |
| 40 | 36 | 36 | 34 | 36 | 36 | 45 | 53 | 62 | 72 | 72 | 72 |
| 50 | 40 | 40 | 40 | 40 | 40 | 50 | 60 | 70 | 80 | 80 | 80 |
| 60 | 48 | 48 | 51 | 48 | 48 | 58 | 67 | 75 | 84 | 84 | 84 |
| 70 | 56 | 56 | 58 | 56 | 56 | 65 | 71 | 78 | 88 | 88 | 88 |
| 80 | 64 | 64 | 62 | 64 | 64 | 71 | 76 | 82 | 92 | 92 | 92 |
| 90 | 72 | 72 | 69 | 72 | 72 | 78 | 82 | 86 | 96 | 96 | 96 |
| 100 | 80 | 80 | 80 | 80 | 80 | 85 | 90 | 95 | 100 | 100 | 100 |

Dari tabel hasil pengujian pada Tabel III, terlihat bahwa perilaku musuh penyerang adalah :

- Melarikan diri : 4% (5 dari 121 data)
- Bertahan : 28% (34 dari 121 data)
- Ragu-ragu : 22% (27 dari 121 data)
- Menyerang : 28% (34 dari 121 data)
- Menyerang Brutal: 17% (21 dari 121)

D. Pengujian Logika Fuzzy pada Musuh Pemanah

Pada musuh tipe pemanah, logika *fuzzy* akan mengatur peluang menyerang musuh berdasarkan parameter *range* dan *ammo*. Secara umum, jika *range* semakin jauh, musuh akan semakin agresif, dan sebaliknya. Jika *ammo* semakin banyak, musuh akan semakin agresif, dan sebaliknya. Pada Tabel IV hasil pengujian logika *fuzzy* pada musuh pemanah untuk parameter masukan *range* dan *ammo* yang berbeda-beda.

TABEL IV.
HASIL PENGUJIAN PERILAKU MUSUH PEMANAH

| <i>range/</i> <i>ammo</i> | 0 | 200 | 400 | 600 | 800 | 1000 | 1200 | 1400 | 1600 | 1800 | 2000 |
|------------------------------|----|-----|-----|-----|-----|------|------|------|------|------|------|
| 0 | 20 | 25 | 30 | 35 | 40 | 50 | 60 | 70 | 80 | 80 | 80 |
| 10 | 20 | 31 | 35 | 39 | 44 | 52 | 60 | 68 | 80 | 80 | 80 |
| 20 | 20 | 34 | 39 | 42 | 48 | 54 | 60 | 66 | 80 | 80 | 80 |
| 30 | 20 | 34 | 40 | 46 | 52 | 58 | 61 | 66 | 80 | 80 | 80 |
| 40 | 20 | 32 | 40 | 48 | 56 | 61 | 65 | 69 | 80 | 80 | 80 |
| 50 | 20 | 30 | 40 | 50 | 60 | 65 | 70 | 75 | 80 | 80 | 80 |
| 60 | 20 | 30 | 40 | 50 | 60 | 65 | 70 | 75 | 80 | 80 | 80 |
| 70 | 20 | 32 | 40 | 48 | 60 | 66 | 70 | 74 | 80 | 80 | 80 |
| 80 | 20 | 32 | 40 | 48 | 60 | 66 | 70 | 74 | 80 | 80 | 80 |
| 90 | 20 | 30 | 40 | 50 | 60 | 65 | 70 | 75 | 80 | 80 | 80 |
| 100 | 20 | 30 | 40 | 50 | 60 | 65 | 70 | 75 | 80 | 80 | 80 |

Dari tabel hasil pengujian pada Tabel IV, terlihat bahwa perilaku musuh pemanah adalah :

- Melarikan diri : 9% (11 dari 121 data)
- Bertahan : 21% (25 dari 121 data)
- Ragu-ragu : 21% (26 dari 121 data)
- Menyerang : 49% (59 dari 121 data)
- Menyerang Brutal : 0% (0 dari 121)

E. Pengujian Logika Fuzzy pada Musuh Boss

Pada musuh tipe boss, logika *fuzzy* akan mengatur peluang menyerang musuh berdasarkan parameter *life* dan *range*. Secara umum, perilakunya sama dengan musuh penyerang, namun boss lebih agresif. Musuh tipe boss ini sama sekali tidak memiliki peluang untuk melarikan diri. Dia hanya akan bertahan (sedikit mundur) ketika *life* nya sudah hampir habis. Pada Tabel

V disajikan hasil pengujian logika *fuzzy* pada musuh boss untuk parameter masukan *range* dan *life* yang berbeda-beda.

TABEL V
HASIL PENGUJIAN PERILAKU MUSUH BOSS

| <i>range/</i> <i>life</i> | 0 | 200 | 400 | 600 | 800 | 1000 | 1200 | 1400 | 1600 | 1800 | 2000 |
|------------------------------|-----|-----|-----|-----|-----|------|------|------|------|------|------|
| 0 | 40 | 40 | 40 | 40 | 40 | 50 | 60 | 70 | 80 | 80 | 80 |
| 10 | 48 | 57 | 62 | 57 | 52 | 58 | 67 | 75 | 84 | 84 | 84 |
| 20 | 56 | 65 | 71 | 67 | 64 | 67 | 71 | 78 | 88 | 88 | 88 |
| 30 | 64 | 71 | 76 | 77 | 76 | 77 | 75 | 82 | 92 | 92 | 92 |
| 40 | 72 | 78 | 82 | 86 | 88 | 86 | 82 | 86 | 96 | 96 | 96 |
| 50 | 80 | 85 | 90 | 95 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| 60 | 84 | 85 | 90 | 95 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| 70 | 88 | 88 | 90 | 94 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| 80 | 92 | 92 | 91 | 94 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| 90 | 96 | 96 | 94 | 96 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |

Dari tabel hasil pengujian pada Tabel V, terlihat bahwa perilaku musuh boss adalah :

- Melarikan diri : 0% (0 dari 121 data)
- Bertahan : 4% (5 dari 121 data)
- Ragu-ragu : 7% (8 dari 121 data)
- Menyerang : 20% (24 dari 121 data)
- Menyerang Brutal: 69% (84 dari 121)

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari hasil perancangan dan uji coba pada penelitian ini, dapat diambil kesimpulan antara lain, permainan "*Song of Ruination 2*" yang dirancang dalam penelitian ini merupakan permainan bertipe *Action-RPG* yang menggabungkan hal-hal umum yang terdapat pada permainan *action/RPG* populer, antara lain *Final Fantasy* dan *Defense of the Ancients*.

Logika *fuzzy* dapat diterapkan untuk mengoptimasi perilaku musuh, kapan musuh maju atau mundur. Perilaku ini ditentukan dari *range* (Jarak musuh terhadap pemain), *ammo* (Jumlah amunisi) dan *life* (Nyawa). Secara umum, musuh akan mundur ketika keadaannya semakin lemah atau amunisinya sedikit.

Penerapan *fuzzy* dalam permainan ini telah berjalan dengan baik, dimana musuh tipe penyerang menjadi cukup agresif (28% perilaku menyerang, 17% menyerang brutal), tipe pemanah agresif jika berada pada jarak jauh (49% perilaku menyerang), dan musuh boss sangat agresif (20% perilaku menyerang, 69% perilaku menyerang brutal).

B. Saran

Untuk pengembangan penelitian dan permainan ini, maka disarankan untuk menambahkan parameter input untuk *fuzzy*, terutama dalam hal kerjasama antar anggota musuh. Misalnya, jika sedang ada 2 musuh yang berdekatan, meskipun nyawanya sedang sedikit, mereka akan maju bersamaan. Atau jika rekan musuh sedang lemah, musuh yang kuat akan mendekatinya untuk berusaha melindunginya.

Kemungkinan pengembangan lain adalah menambahkan jenis musuh yang ada, misalnya musuh yang dapat menggunakan jurus. Parameter untuk tipe musuh ini pun dapat dibuat bervariasi, antara lain kekuatan sihirnya, jangkauan sihirnya dan sebagainya.

DAFTAR PUSTAKA

- [1] J. Juul. 2000. (20 Juni 2013). "What computer games can/can't do". Tersedia di <http://www.jesperjuul.net/text/wcgacad.html>.
- [2] C. Crawford. "The Art of Computer Game Design". Columbus: McGraw-Hill, 1984, p.1-3.
- [3] Wikipedia. (25 Maret 2013). "Action Game". Tersedia di http://en.wikipedia.org/wiki/Action_game.
- [4] Wikipedia. (25 Maret 2013). "Role-Playing Game". Tersedia di http://en.wikipedia.org/wiki/Role-playing_game.
- [5] A.G. Abdullah. "Pengantar Kecerdasan Buatan". Bandung : Universitas Pendidikan Indonesia, 2008, p.2.
- [6] Shaout, Adnan, B. King dan L. Reisner. "Real Time Game Design of Pacman using Fuzzy Logic". The International Arab Journal of Information Technology, 2006, vol. 3 no. 4. p.320
- [7] Arif, Y. Miftachul, F. Kurniawan, F. Nugroho. "Desain Perubahan Perilaku pada NPC Game Menggunakan Logika Fuzzy". Seminar on Electrical, Informatics and Education, 2011, p. A2.110 - A2.111
- [8] A. Naba. "Belajar Cepat Fuzzy Logic Menggunakan Matlab". Yogyakarta : Penerbit ANDI, 2009, p.31