

STRATEGI MENYERANG NPC GAME FPS MENGUNAKAN FUZZY FINITE STATE MACHINE

Ady Wicaksono¹⁾, Mochamad Hariadi²⁾, Supeno Mardi S. N³⁾

¹⁾ Pasca Sarjana Teknik Elektro ITS, Surabaya 60111

²⁾ Jurusan Teknik Elektro, ITS Surabaya 60111

³⁾ Jurusan Teknik Elektro, ITS Surabaya 60111

email : Adynk75@gmail.com¹⁾, mochar@ee.its.ac.id²⁾, mardi@ee.its.ac.id³⁾

Abstrak

Pengembangan kecerdasan buatan untuk NPC (Non-Playable Character) dalam game menjadi salah satu fokus pengembangan yang penting karena NPC seringkali mempunyai fungsi yang esensial membuat dunia artifisial dalam game menjadi lebih imersif. Penelitian ini disusun untuk membahas rancangan behavior (perilaku) strategi menyerang NPC pada game FPS menggunakan Finite State Machine, sedangkan untuk respon perilaku menyerang NPC diimplementasikan dengan logika fuzzy. 3 NPC menjadi fokus penelitian dengan masing-masing NPC memiliki karakteristik menyerang berbeda. NPC 1 mempunyai atribut serangan jarak dekat dan menengah, NPC 2 untuk serangan jarak menengah dan jarak jauh sedangkan NPC 3 memiliki spesifikasi serangan jarak dekat, menengah sekaligus jarak jauh. Perilaku NPC 1, NPC 2 dan NPC 3 didasarkan pada perubahan jarak musuh, jumlah musuh serta tingkat kesehatan (hp) dari masing-masing NPC. Simulasi dengan game engine menunjukkan perubahan respon jenis serangan dari masing-masing NPC. NPC 1 menunjukkan perubahan respon variasi menyerang jarak dekat dan menengah. NPC 2 menunjukkan variasi respon perilaku menyerang jarak menengah dan jarak jauh. NPC 3 menunjukkan variasi respon serangan jarak dekat, menengah dan jarak jauh.

Kata kunci :

Non-Playable Character, strategi menyerang, Fuzzy Finite State Machine, logika fuzzy

1. Pendahuluan

AI merupakan cara atau bagaimana membuat komputer mampu melakukan dan menunjukkan kegiatan berfikir seperti halnya manusia dan binatang [1]. Dalam game komputer, Artificial Intelligence (AI) secara umum berarti membuat komputer mampu berfikir rasional dan juga mampu bertindak seperti manusia [2].

Dalam game FPS, pada saat menyerang, NPC tentu diharapkan tidak sekedar menyerang, namun bisa memilih strategi khusus untuk menyerang player. Pada akhirnya, strategi NPC untuk menyerang, itu merupakan bagian dari taktik untuk mengalahkan player. Penelitian ini dilakukan untuk mengimplementasikan Fuzzy State

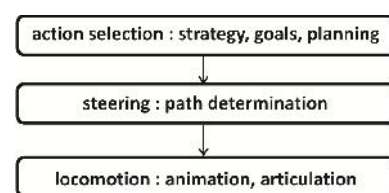
Machine pada game jenis First Person Shooter (FPS). Model dari behavior NPC akan di desain menggunakan Finite State Machine, sedangkan untuk respon behavior menyerang NPC akan diimplementasikan dengan logika fuzzy. Untuk implementasi dan simulasi perilaku NPC akan digunakan game engine Torque Game Engine 1.4.

2. Tinjauan Pustaka

Penelitian ini merupakan penelitian yang disusun dan berdasarkan beberapa penelitian lain dalam bidang yang sama (game) yang telah dilakukan sebelumnya.

2.1. Non – Playable Character (NPC)

NPC (Non-Playable Character) merupakan semua obyek atau karakter dalam game, yang tidak dimainkan oleh player (manusia) [3], yang dijalankan oleh komputer dan bisa berinteraksi dengan pemain, sehingga bersifat otonom, maka dari itu pula, sistem agen, atau karakter otonom dalam game memainkan peran yang sangat signifikan dalam pengembangan game. Terkait dengan perilaku strategis, terdapat pembagian perilaku agen otonom dalam lapisan yang hirarkis seperti pada gambar 1, dan fokus pada action selection menjadi titik berat untuk penelitian ini



2.2 Logika Fuzzy

Logika Fuzzy adalah sebuah metode untuk menangani masalah ketidakpastian, yaitu suatu masalah yang mengandung keraguan, dan nilai kebenarannya bersifat sebagian.

Dalam game, ada kondisi karakter bergerak dalam lingkungan yang berbahaya. Dengan pendekatan finite state machine normal, maka akan ada pilihan “waspada” atau “santai”. Dengan logika fuzzy, sistem memperkenalkan fungsi untuk berperilaku “jalan pelan-pelan saat waspada”, namun kedua kondisi “waspada” dan “santai” masih memiliki range untuk derajat tingkatan kondisi yang dimaksud [1].

Sistem yang menggunakan logika *Fuzzy* mampu menangani masalah ketidakpastian dengan masukan suatu nilai yang kebenarannya bersifat sebagian (parsial). Sebagai bentuk teknis, logika *fuzzy* merupakan teknik untuk pencapaian agar, secara matematis maupun sistematis bisa menyamai proses *reasoning* dan *decision making* seperti manusia [9]. Atas dasar tersebut, logika *Fuzzy* digunakan pada penelitian ini untuk mendapatkan respon perilaku NPC berdasarkan variabel inputnya.

Terdapat banyak model aturan *Fuzzy* yang bisa digunakan dalam proses *inference* akan tetapi ada dua model aturan yang paling sering digunakan yaitu:

1. Model Mamdani

Bentuk aturan yang digunakan pada model Mamdani adalah sebagai berikut [8] :

$$IF x_1 \text{ is } A_1 \text{ AND } \dots \text{ AND } x_n \text{ is } A_n \text{ THEN } y \text{ is } B \dots\dots\dots(1)$$

Aturan tersebut menyatakan bahwa A_1, \dots, A_n, B adalah nilai-nilai linguistik, sedangkan “ $x_1 \text{ is } A_1$ “ menyatakan bahwa nilai dari variabel x_1 adalah anggota *fuzzy set* A_1 .

2. Model Sugeno

Model Sugeno merupakan varian dari model Mamdani dan memiliki bentuk aturan sebagai berikut [8]:

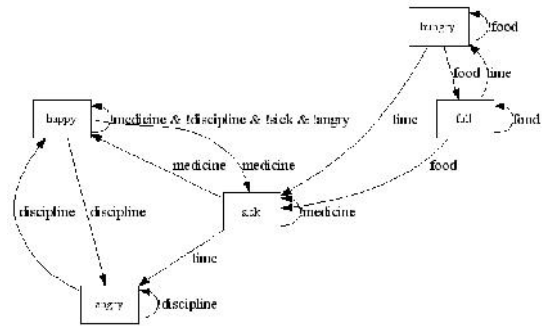
$$IF x_1 \text{ is } A_1 \text{ AND } \dots \text{ AND } x_n \text{ is } A_n \text{ THEN } y = f(x_1, \dots, x_n) \dots\dots\dots(2)$$

Model Sugeno menyatakan bahwa f bisa berupa sembarang fungsi dari variabel-variabel *input* yang nilainya berada dalam interval variabel *output*. Biasanya, fungsi ini dibatasi dengan menyatakan f sebagai kombinasi *linier* dari variabel-variabel *input*.

2.3 Fuzzy Finite State Machine

State machine yang umum dipergunakan dirancang untuk berada dalam satu *state* atau *state* yang lain secara ketat, atau satu *state* sebagai nilai tunggal [1]. Pada kondisi yang lebih nyata, banyak situasi yang tidak selalu hitam atau putih, dalam artian berada dalam area abu-abu [5].

Fuzzy State Machine (FuSM) bisa saja dianggap sebagai *state machine* yang mengandung elemen *fuzzy* (kesamaran), sehingga memungkinkan sistem secara simultan berada sebagian di *current state* dan sebagian di *next state*. Gambar 2 menggambarkan *fuzzy state machine* untuk salah satu siklus dari karakter Tomagotchi.

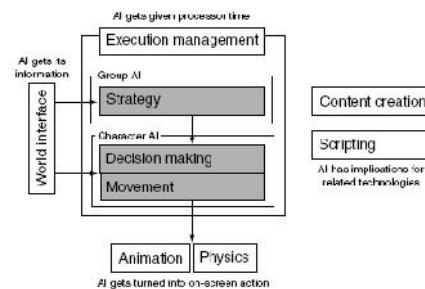


Gambar 2 *Fuzzy State Machine* karakter Tomagotchi pada siklus “mood” [6]

2.4 Permodelan AI dalam Game

Setiap *game* membutuhkan *AI* spesifik yang mendukung mekanisme *game* tersebut [4]. *Game* memiliki varian *genre* yang berbeda, dengan karakteristik berbeda, baik dari visualisasi maupun interaktifitasnya, namun berorientasi sama, yaitu untuk meningkatkan level “*believable*” dari *game* yang dibuat.

Dalam tulisan ini, hanya akan difokuskan pada *game* dalam *genre First Person Shooter*, sebagaimana kesesuaiannya dengan subyek penelitian. Secara umum, rancangan *AI* dalam *game* terbagi dalam blok-blok *AI* sesuai dengan orientasi operasi dan interaktifitas seperti dalam gambar 3.

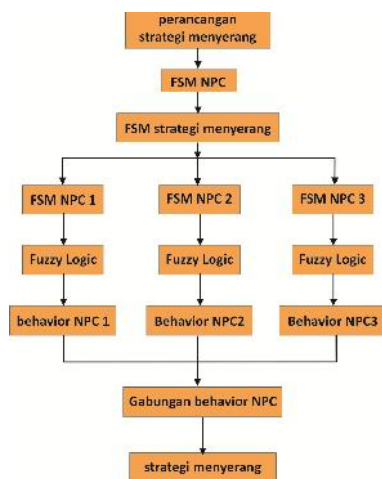


Gambar 3 Model *Artificial Intelligence* untuk *game* [1]

Secara umum, struktur *AI* untuk *game FPS* tersusun dalam beberapa blok *layer*, yaitu *movement*, *animation*, *combat* dan *behavior* [1] [7]. *Layer movement* bertanggungjawab membentuk karakteristik NPC untuk melakukan aksi “bergerak” di lingkungan *game*. *Layer animation* bertanggungjawab untuk mengontrol tubuh dari karakter berinteraksi dengan lingkungannya secara fisik. *Layer combat* bertanggungjawab untuk kontrol situasi dan seleksi taktis, membidik dan menembak, keputusan pilihan senjata dan seterusnya. Karena pertempuran merupakan inti *gameplay* dari *game FPS*, maka performa dari subsistem ini sangat krusial. *Layer behavior* adalah subsistem yang menentukan tujuan, sasaran dari karakter. *Layer* ini adalah level tertinggi karena menentukan perilaku secara keseluruhan, dan tujuan utama [7].

3. Metode Penelitian

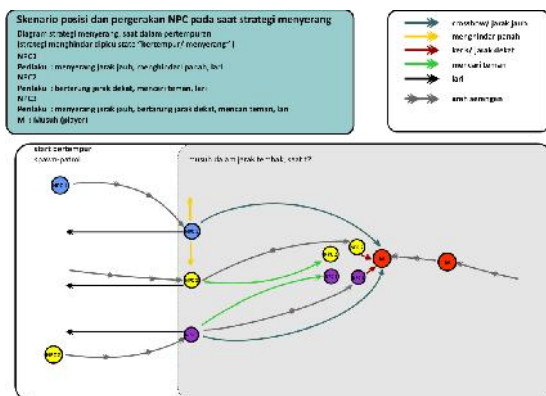
Metode yang digunakan untuk mendesain strategi menyerang pada penelitian ini adalah *Fuzzy State Machine (FuSM)*, dan perilaku *NPC* berdasarkan variabel yang dimilikinya digunakan logika *fuzzy*. Penelitian ini disusun dalam tahapan yang ditunjukkan dalam gambar 4, yang meliputi perancangan skenario *game* untuk strategi menyerang, *Finite State Machine* untuk *NPC 1*, *NPC 2* dan *NPC 3*, serta logika *fuzzy* yang diterapkan pada *NPC*.



Gambar 4 Diagram blok metodologi penelitian

3.1 Skenario Strategi Menyerang

Untuk penelitian ini, dibuat skenario untuk *game* seperti dalam gambar 5, yang dijadikan medium untuk simulasi dan uji coba dengan dua jenis karakter, yakni sebagai pemain (*player*), serta tim *NPC*, yang terdiri dari *NPC 1*, *NPC 2* dan *NPC 3* yang menjadi obyek penelitian ini. *Fuzzy logic* mempengaruhi perilaku pada saat *state* menyerang.

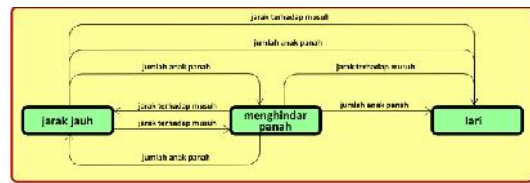


Gambar 5 Rancangan skenario strategi menyerang NPC

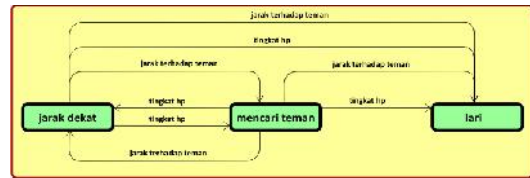
3.2 Fuzzy Finite State Machine Strategi Menyerang

Dengan logika *fuzzy* dalam perancangan perilaku, 3 *NPC*, maka aplikasi *multiple state* menjadi mungkin. Dalam penelitian ini, logika *fuzzy*

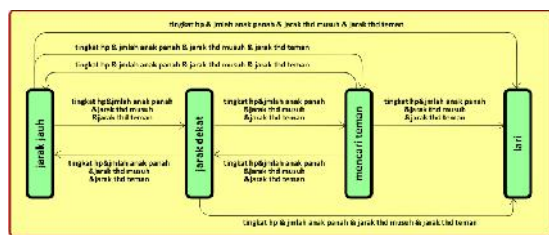
diaplikasikan pada perilaku tiga *NPC* pada saat strategi menyerang, seperti pada gambar 6, 7, dan 8.



Gambar 6 FuSM menyerang untuk NPC 1



Gambar 7 FuSM menyerang untuk NPC 2



Gambar 8 FuSM menyerang untuk NPC 3

3.3 Logika Fuzzy

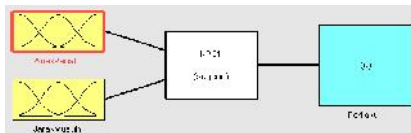
Pada penelitian ini, logika *fuzzy* digunakan untuk merancang perilaku menyerang *NPC 1*, *NPC 2* dan *NPC 3*. Metode *fuzzy* yang digunakan adalah metode Sugeno, karena metode ini menghasilkan *output* yang berupa konstanta tegas, sehingga dapat mewakili nilai perilaku yang sudah didesain sebelumnya.

3.4 Desain Fuzzy untuk NPC 1, NPC 2 dan NPC 3

Perilaku *NPC 1* didesain dengan variabel input “jumlah anak panah” dan variabel “jarak terhadap musuh”. Logika *fuzzy* dan desain fuzzy perilaku *NPC 1* ditunjukkan dalam gambar 9 dan gambar 10. Untuk *NPC 2*, variabel *input* yang untuk merancang perilaku adalah “tingkat hp” dan “jarak terhadap teman”, seperti ditunjukkan dalam gambar 11 dengan desain *fuzzy* seperti pada gambar 12. Perilaku untuk *NPC 3* merupakan gabungan dari *NPC 1* dan *NPC 2*. *NPC 3* memiliki 4 variabel *input*, yaitu “tingkat hp”, “jumlah anak panah”, “jarak terhadap musuh” dan “jarak terhadap teman”. Diagram penerapan logika *fuzzy* untuk perilaku pada *NPC 3* ditunjukkan dalam Gambar 13, dan desain *fuzzy* perilaku *NPC 3* pada gambar 14.



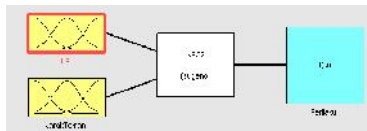
Gambar 9 Logika Fuzzy untuk menghasilkan perilaku NPC 1



Gambar 10 Desain Fuzzy untuk menghasilkan perilaku NPC 1



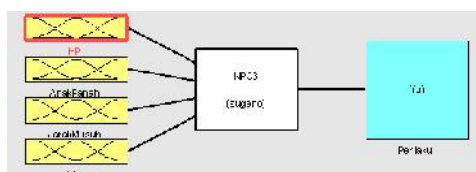
Gambar 11 Logika Fuzzy untuk menghasilkan perilaku NPC 2



Gambar 12 Desain Fuzzy untuk menghasilkan perilaku NPC 2



Gambar 13 Logika Fuzzy untuk desain perilaku NPC 3



Gambar 14 Desain Fuzzy untuk menghasilkan perilaku NPC 3

4. Hasil dan Pembahasan

Langkah uji coba dilakukan dengan melakukan uji pada logika fuzzy yang dipergunakan untuk variasi perilaku NPC sesuai dengan *FuSM* sesuai dengan variabel input untuk masing-masing NPC. Setelah hasil uji fuzzy sudah didapatkan, maka hasilnya diuji cobakan dalam bentuk simulasi dalam bentuk *game*, dengan menggunakan *Torque Game Engine*.

4.1. Pengujian Fuzzy pada NPC 1, NPC 2 dan NPC 3

Untuk menghasilkan perubahan perilaku NPC 1, parameter yang dimiliki NPC 1 diujikan, yakni beberapa nilai variabel jumlah anak panah (amunisi) dan variabel jarak terhadap musuh mulai dari minimum sampai maksimum. Notasi dari masing-masing perilaku diset pada rentang lari (notasi nilai 0), menghindari tembakan (nilai 1), dan menyerang jarak jauh (nilai 2). Hasilnya seperti ditunjukkan dalam tabel 1.

Tabel 1 Hasil pengujian NPC 1 dengan menggunakan variabel parameter masukan yang berbeda.

		JARAK MUSUH																				
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15					
JUNJAH ANAK PANAH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,47	1	1	1	1
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,47	1	1	1	1
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,47	1	1	1	1
	3	0	0	0	0	0	0,12	0,5	0,5	0,5	0,5	0,5	0,5	0,77	1	1	1	1	1	1	1	1
	4	0	0	0	0	0,25	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	5	0	0	0	0	0,25	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	6	0	0	0	0	0,25	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	7	0	0	0	0,18	0,44	1	1	1	1	1	1	1	1	1,12	1,25	1,25	1,25	1,25	1,25	1,25	1,25
	8	0	0	0	0,36	0,62	1	1	1	1	1	1	1	1	1,24	1,5	1,5	1,5	1,5	1,5	1,5	1,5
	9	0	0	0	0,54	0,81	1	1	1	1	1	1	1	1	1,36	1,5	1,5	1,5	1,5	1,5	1,5	1,75
	10	0	0	0	0,72	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	11	0,25	0,25	0,25	0,79	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	12	0,5	0,5	0,5	0,86	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	13	0,75	0,75	0,75	0,93	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	14	1	1	1	1	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	15	1	1	1	1	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	16	1	1	1	1	1	1	1	1	1	1	1	1	1	1,47	2	2	2	2	2	2	2
	17	1	1	1	1	1,13	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,74	2	2	2	2	2	2	2	2
	18	1	1	1	1	1,25	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	19	1	1	1	1	1,25	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	20	1	1	1	1	1,25	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Blok warna hijau menunjukkan rentang respon 0 hingga 0,5, merah muda pada rentang 1 hingga 1,5, sedangkan biru muda untuk rentang nilai respon 1,74 hingga 2. Dari pembagian tersebut, warna hijau mewakili perilaku lari, merah muda untuk menghindari tembakan dan biru muda untuk menyerang jarak jauh.

Pada pengujian yang dilakukan untuk NPC 2, variabel input yang diujicobakan berjumlah dua, yakni jarak terhadap teman dan tingkat kesehatan (hp). NPC 2 memiliki 3 variasi perilaku, yaitu lari (notasi nilai 0), mencari/mendekat teman (nilai 1) dan perilaku menyerang jarak dekat (nilai 2).

Hasil menunjukkan respon perubahan perilaku sesuai kombinasi nilai variabel input jarak terhadap teman dan tingkat hp. Blok warna hijau berisi rentang nilai antara 0 hingga 0,57 untuk perilaku lari, blok merah muda dengan rentang nilai 0,86 hingga 1,47 untuk perilaku mendekati teman, serta blok biru muda dengan rentang 1,72 hingga 2 untuk perilaku menyerang

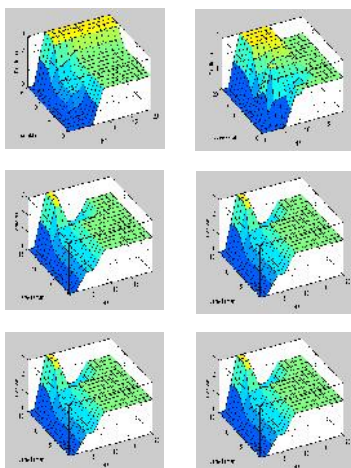
jarak dekat. Hasil-hasil tersebut ditunjukkan seperti pada tabel 2.

Tabel 2 Hasil pengujian perilaku NPC 2 dengan menggunakan variabel parameter masukan yang berbeda

		JARAK TEMAN															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
TINGKAT HP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0,27	0,57	0,57	0,57
	4	0	0	0	0	0	0	0	0	0	0	0	0	0,47	1	1	1
	5	0	0	0	0	0	0	0	0	0	0	0	0	0,47	1	1	1
	6	0	0	0	0	0	0	0	0	0	0	0	0	0,47	1	1	1
	7	0,86	0,86	0,86	0,86	0,86	0,86	0,86	0,86	0,86	0,86	0,86	0,93	1	1	1	1
	8	2	2	2	2	2	2	2	2	2	2	2	2	1,50	1	1	1
	9	2	2	2	2	2	2	2	2	2	2	2	2	1,53	1	1	1
	10	2	2	2	2	2	2	2	2	2	2	2	2	1,53	1	1	1
	11	2	2	2	2	2	2	2	2	2	2	2	2	1,53	1	1	1
	12	2	2	2	2	2	2	2	2	2	2	2	2	1,53	1	1	1
	13	2	2	2	2	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,25	1	1	1	1
	14	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1
	15	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1
	16	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1
	17	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1
	18	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1
	19	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1
	20	2	2	2	2	1,47	1	1	1	1	1	1	1	1	1	1	1

Perilaku NPC 3 merupakan gabungan dari NPC 1 dan NPC 2, baik variabel input maupun variasi respon outputnya. Variabel input untuk NPC 3 adalah tingkat hp, jumlah amunisi (anak panah), jarak terhadap teman dan jarak terhadap musuh. Output perilaku dari NPC 3 adalah menyerang jarak jauh, menyerang jarak dekat, mencari teman dan lari.

Pengujian untuk NPC 3 tidak dibentuk dalam matriks tabel. Pengujian dilakukan sesuai dengan fuzzy rule yang telah dibuat untuk perilaku NPC 3. Dengan 4 input dan 4 output, maka terdapat 300 rule, maka pengujian dilakukan dengan memasukkan beberapa nilai yang mewakili masing-masing variabel linguistik dari input sebagai sampel, yang hasil dalam bentuk grafiknya ditunjukkan dalam gambar 15.



Gambar 15 Grafik permukaan perilaku NPC 3

4.2 Simulasi dalam game

Dengan kompleksitas game engine yang dipergunakan, uji simulasi dirujukkan dengan tujuan utama memperoleh respon fuzzy dari tiap NPC. Untuk itu, parameter variabel input dirubah didasarkan pada perubahan jarak musuh, jumlah musuh serta tingkat

kesehatan (hp) dari masing-masing NPC, sedangkan output perilaku dirupakan dalam bentuk pilihan senjata (jarak pendek,, menengah/blaster, dan jarak jauh/rocket launcher) yang secara langsung merupakan respon perilaku serangan/ strategi menyerang jarak pendek, menengah, dan jarak jauh.

Simulasi dilakukan dengan memainkan game sebanyak 5 kali untuk mendapatkan data perilaku terhadap jarak musuh, dengan jumlah musuh (player) bervariasi dari 1 player, 3 player, dan 5 player. prosentase perubahan perilaku menyerang seperti ditunjukkan pada tabel 3.

Tabel 3 Prosentase perubahan perilaku menyerang NPC 1 terhadap jarak musuh

Σ musuh	klasifikasi jarak	prosentase perilaku menyerang	
		blaster	s serbu
1	jauh	100%	0%
	medium	96%	4%
	dekat	4%	96%
3	jauh	100%	0%
	medium	16%	84%
	dekat	0%	100%
5	jauh	100%	0%
	medium	96%	4%
	dekat	0%	100%

Data respon perilaku menunjukkan pengaruh jumlah musuh terhadap prosentase pilihan senjata, saat jumlah musuh=1, pilihan senjata blaster direspon sebesar 76%, sedangkan senapan serbu sebanyak 24%. Saat jumlah musuh=3, blaster sebesar 60% dan senapan serbu sebesar 40%, dan saat jumlah musuh=5, blaster sebesar 56% dan senapan serbu sebesar 44%.

NPC 2 menunjukkan respon serangan jarak menengah (blaster) dan jarak jauh (rocket) dengan prosentase seperti ditunjukkan pada tabel 4.

Tabel 4 Prosentase perubahan perilaku menyerang NPC 2 terhadap jarak musuh

Σ musuh	klasifikasi jarak	prosentase perilaku menyerang	
		blaster	rocket
1	jauh	0%	100%
	medium	32%	68%
	dekat	100%	0%
3	jauh	100%	0%
	medium	32%	68%
	dekat	100%	0%
5	jauh	0%	100%
	medium	8%	92%
	dekat	100%	0%

Untuk respon terhadap perubahan tingkat hp, NPC 2 menunjukkan respon saat jumlah musuh=1, penggunaan rocket sebesar 46% dan blaster sebesar 54%. Saat jumlah musuh=2, blaster sebesar 36% dan rocket sebesar 64%, sedangkan saat jumlah musuh=5, blaster sebesar 18% dan rocket sebesar 89%. Hal ini menunjukkan adaptasi terhadap kondisi pertempuran.

Sama dengan yang dilakukan pada NPC 1 dan NPC 2, simulasi untuk mendapatkan data perilaku NPC 3 dilakukan dengan klasifikasi yang sama. Tabel 5 menunjukkan hasil yang diperoleh, meliputi perilaku menyerang jarak jauh, jarak menengah dan jarak dekat (rocket, blaster, dan senapan serbu).

Tabel 5 Prosentase perubahan perilaku menyerang NPC 3 terhadap jarak musuh

Σ musuh	klasifikasi jarak	prosentase perilaku menyerang		
		blaster	s serbu	rocket
1	jauh	0%	0%	100%
	medium	68%	0%	32%
	dekat	48%	52%	0%
3	jauh	0%	0%	100%
	medium	80%	0%	20%
	dekat	52%	48%	0%
5	jauh	0%	0%	100%
	medium	68%	0%	32%
	dekat	52%	48%	0%

Pengaruh tingkat hp terhadap respon serangan NPC 3 saat jumlah musuh=1 menunjukkan hasil sebesar 74% untuk *rocket*, 16% untuk *blaster* dan 10% untuk senapan serbu. Saat jumlah musuh=3, *rocket* sebesar 60%, *blaster* sebesar 30% dan senapan serbu sebesar 10%. Pada saat jumlah musuh=5, *rocket* digunakan sebesar 62%, *blaster* sebesar 32% dan senapan serbu sebesar 6%. Hal ini menunjukkan bahwa NPC 3 memberikan fokus respon untuk melakukan cover serangan jarak jauh.

5. Kesimpulan dan Saran

Dari uji coba yang dilakukan untuk penelitian ini diperoleh kesimpulan:

1. *Fuzzy State Machine* bisa dipergunakan untuk membentuk variasi respon NPC dalam game ketika berinteraksi dengan musuh (*player*) dalam situasi konfrontasi didalam lingkungan game.
2. Perilaku pada NPC 1, NPC 2 dan NPC 3 berubah sesuai perubahan variabel input masing-masing NPC. Perubahan penggunaan senjata untuk jarak dekat (senapan serbu), jarak menengah (*blaster*) dan jarak jauh (*rocket*) juga berubah sesuai dengan perubahan kondisi kesehatan (hp), jumlah musuh serta koordinat terhadap musuh (*player*)

Untuk penelitian lanjutan dengan topik yang homogen dibidang *game development*, saran yang bisa disampaikan adalah:

1. *Fuzzy State Machine* bisa dikembangkan dengan penggabungan dengan metode kecerdasan yang lain dalam model *non-deterministic FSM* untuk mendapatkan variasi respon yang lebih realistis.
2. Ditambahkan beberapa NPC dengan perilaku yang beragam, sehingga game menjadi lebih variatif.
3. Akan lebih baik jika simulasi menggunakan *game engine* yang sesuai dan tidak membutuhkan komputasi yang tinggi.

Daftar Pustaka

- [1] Millington, Ian. 2006. *Artificial Intelligence for Games*. Morgan Kaufmann Publishers, San Francisco.
- [2] Fidaner, I ik Bari . 2005. *Artificial Intelligence in*

Computer Games. CMPE 540 Principles of Artificial Intelligence, page 1-11.

- [3] Doherty, Darren. 2006. *Evolving Tactical Teams for Shooter Games using Genetic Programming*. Department Of Information Technology National University Of Ireland, Galway.
- [4] Tozour, Paul; Austin, Ion Storm. 2004. *First Person Shooter AI Architecture*, dalam: *AI Game Programming Wisdom*. Charles River Media. Hingham, Massachussets.
- [5] Schwab, Brian. 2004. *AI Game Engine Programming*. Charles River Media, Hingham, Massachussets.
- [6] Seland, Jørgen Havsberg. 2007. *A Visual Programming Language for Hierarchical Finite State Machines in Game AI*. Department of Computer Science, University of Copenhagen.
- [7] McCloskey, John; Miller, Jeffrey; Prasad, Amish; & Linden, Lars. 2009. *AI in First-Person Shooter Games*. McGill University, Montreal, QC H3A 2A7, Canada.
- [8] Suyanto, S.T, M.Sc. 2007. *Artificial Intelligence: Searching, Reasoning, Planning and Learning*. Penerbit Informatika, Bandung, Indonesia
- [9] Carr, Dave ; Shearer, Jeff. 2007. *Nonlinear Control and Decision Making Using Fuzzy Logic in Logix*. Publication 9324-WP006A-EN-P-August 2007. Rockwell Automation, Milwaukee.

Biodata Penulis

Ady Wicaksono, memperoleh gelar Sarjana Pendidikan (S.Pd), Program Studi Pendidikan Seni Rupa UM, lulus tahun 2003, Memperoleh gelar Sarjana Seni (S.Sn), Program Studi Film dan Televisi IKJ, lulus tahun 2007. Saat ini sebagai staf pengajar program DKV Universitas Negeri Malang. **Mochamad Hariadi, S.T., M.Sc., Ph.D., Supeno Mardi Susiki Nugroho, S.T. M.T.**, saat ini sebagai staf pengajar di Program Magister Jurusan Teknik Elektro, Bidang Keahlian Jaringan Cerdas Multimedia, Institut Teknologi Sepuluh Nopember Surabaya.