Jurnal Ilmiah KURSOR
Menuju Solusi Teknologi Informasi

Vol. 6, No. 1, Januari2011 **ISSN 0216 - 0544**

PERILAKU TAKTIS UNTUK *NON-PLAYER CHARACTERS* DI *GAME* PEPERANGAN MENIRU STRATEGI MANUSIA MENGGUNAKAN FUZZY *LOGIC* DAN *HIERARCHICAL FINITE STATE MACHINE*

[a]Supeno Mardi Susiki Nugroho, [b]Yunifa Miftachul Arif , [c]Mochamad Hariadi, [d]Mauridhi H Purnomo

[a,c,d]Jurusan Teknik Elektro, ITS
Jl. Raya ITS, Kampus ITS, Sukolilo, Surabaya, 60111
[b]STMIK ASIA
Jl Borobudur 21, Malang, 61234
E-Mail: [a]mardi@ee.its.ac.id

Abstrak

Saat ini, perkembangan teknologi game khususnya agen *Non-Player Character*(*NPC*) yang mempunyai perilaku taktis mirip manusia semakin populer. Perilaku itu memerlukan aturan-aturan dasar yang komprehensif yang dapat menanggapi perilaku pemain. Untuk mengatasi kebutuhan itu, diusulkan *NPC* yang berperilaku adaptif dengan logika *fuzzy*, dimana didefinisikan empat jenis perilaku manuver, yaitu menyerang dengan brutal, menyerang, bertahan dan melarikan diri; yang tergantung pada kesehatan, amunisi *NPC*, dan jarak dengan musuh. Empat jenis perilaku itu digunakan oleh dua agen otonom *NPC* menggunakan *Hierarchy Finite State Machine* untuk bermanuver secara adaptif saat adegan tempur. Lalu *NPC* diuji pada simulasi pertempuran dan diamati perilakunya.Menggunakan *First Person Game Engine*, kinerja *NPC* dengan perilaku berbasis *fuzzy* dibandingkan dengan *NPC* tanpa perilaku fuzzy. Hasil penelitian menunjukkan kinerja *NPC* dengan perilaku *fuzzy* mengungguli *NPC* tanpa perilaku *fuzzy* 80% lebih baik.

Kata kunci:*Non-Player Character*, Manuver Adaptif, Logika Fuzzy, *Hierarchical Finite State Machine.*

Abstract

Nowadays, the proliferation of game technology especially in intelligent human-like NPCs (Non-Player Characters) leads to more adaptive behavior of NPCs maneuvers. Providing smoother behaviors require comprehensive rules base which can respond to players behaviors. Addressing this requirement we propose NPCs which had tactical behaviors based on fuzzy logic. The fuzzy logic defines four type behaviors for the NPCs, which depend on NPC health, ammo, and distance of the enemy.Those behaviors implemented on two intelligent agents employed Hierarchical Finite State Machine to express the maneuver actions of NPC during combat scenes. Using First Person Game Engine, the performance of NPCs with fuzzy behavior compared with NPC without fuzzy behavior. The results of experiment showed the performance of the NPC with Fuzzy behavior outperform 80% better than the NPC without fuzzy behavior.

Key words: Non-Player Characters, Adaptive Maneuver, Fuzzy Logic, Hierarchical Finite State Machine.

PENDAHULUAN

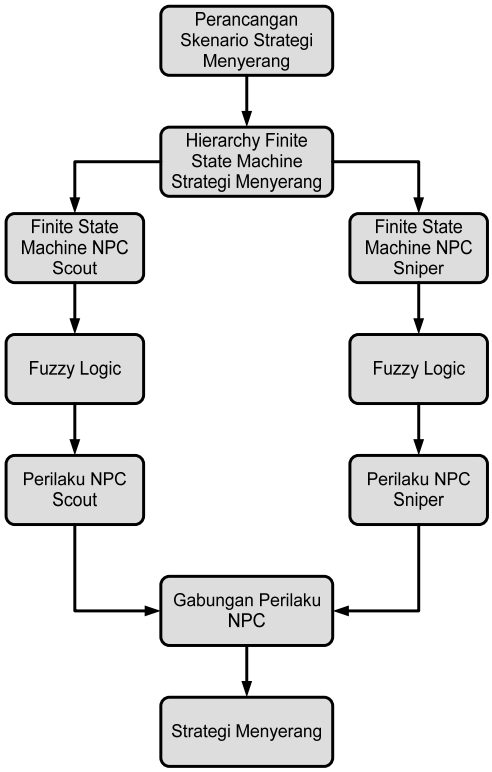
Penggunaan agen sebagai alat pengendali dalam sebuah sistem cerdas telah berkembang dalam dekade ini. Di riset bidang system informasi, agen cerdas digunakan untuk memonitor stok perusahaan berbasis sistem *Reorder Point* [1]. Pada permainan komputer, perilaku agen berupa *Non-PlayerCharacter* (*NPC*) hingga saat ini terus dikembangkan dengan menggunakan kecerdasan buatan [2].

Untuk *genre* permainan komputer tertentu misalnya*First Person Shooter*(*FPS*) dan *Real-Time Strategy*(*RTS*), perilaku *NPC* merujuk pada kemampuan permainan untuk menantang pemain pada tingkat taktis dan strategis [3].

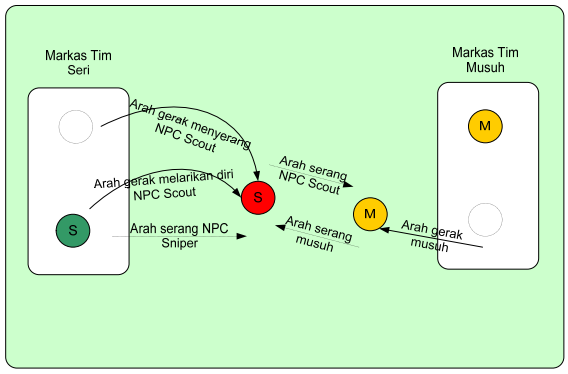
Khusus untuk berperang *NPC* juga diharapkan mempunyai strategi-strategi khusus seperti halnya manusia [4]. Salah satu strategi adalah bagaimana membuat perilaku taktis dalam bermanuver untuk mengalahkan musuh. Misalnya jika *NPC* bergerak menghindar dari serangan musuh dan mengarah ke arah tim, maka *NPC* akan mendapatkan bantuan serangan dari rekan satu tim. Ketika musuh mengejar dan bergerak mendekati persembunyian tim, *NPC* temannya dapat menyergap atau menyerang dengan tiba-tiba. Penelitian ini melakukan percobaan pemodelan manuver agen menghindar dari dari gerakan semula yaitu menyerang.

Model pergerakan agen *NPC* yang sederhana dapat dibuat dengan *Finite State Machine* (*FSM*). Untuk *basicFSM*, *state* tersusun lebih sederhana dan berurutan, namun memiliki banyak kelemahan karena sistem yang paling praktis memiliki jumlah *state* dan transisi yang banyak sehingga representasi dan analisis menjadi sulit [5]. Song mengusulkan *action-selection-mechanism* (*ASM*) yang terkoneksi *Finite State Machine* untuk untuk menangani state perilaku berurutan, yang diterapkan pada robot, yang dapat diterapkan pada perintah berhirarki, tetapi belum dicobakan pada permainan komputer[6].

Penelitian ini melakukan pemodelan dua agen *NPC* pada permainan komputer, dengan menggunakan *Hierarchy Finite State Machine* (*HFSM*), dan memasang logika *fuzzy* didalamnya. Model ini diujicobakan pada *NPC* pengumpan (*scout*) dan *NPCsniper* pada game perang untuk dapat berubah perilaku secara adaptif.



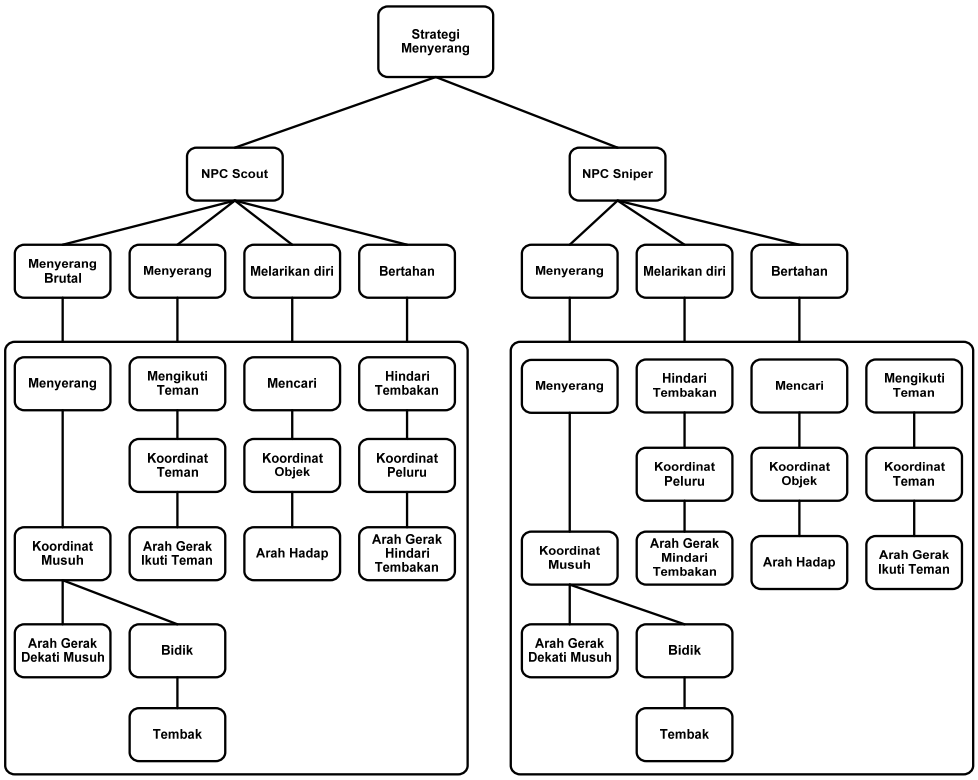
Gambar 1. Blok Diagram Penelitian 2 Agen Menggunakan *Hierarchy Finite State Machine* untuk Bermanuver Secara Adaptif.



Gambar 2. Skenario Manuver *NPCScout* dan *NPCSniper*.

Hierarchy Finite State Machine untuk bermanuver secara adaptif

Gambar 1 menunjukkan tahapan-tahapan dalam penelitian yang meliputi perancangan skenario menyerang bertahan dan melarikan diri, perancangan *Hierarchy Finite State Machine*, *Finite State MachineNPCScout* dan *Sniper*, serta logika *fuzzy*.



Gambar 3. *Hierarchy Finite State Machine* Strategi Menyerang.

Skenario Manuver *NPC* scout dan *NPCSniper N*

Pada penelitian ini dibuat sebuah skenario *game* perang menggunakan karakter tank. Seperti pada Gambar 2, terdapat dua tim yang terlibat peperangan dalam *game* ini, yaitu tim seri dan tim musuh. Tim seri yang dimaksud adalah *NPCScout* dan *NPCSniper*. Tim seri yang merupakan obyek dalam penelitian ini mempunyai beberapa perilaku dalam melakukan manuver penyerangan. Perilaku yang dimiliki *NPCScout* adalah menyerang brutal, menyerang, melarikan diri dan bertahan. *NPCScout* cenderung lebih banyak menyerang, karena tugas utamanya adalah memancing serangan musuh. Sedangkan perilaku yang dimiliki *NPCSniper* adalah menyerang, bertahan dan melarikan diri. Fungsi utama dari *NPCSniper* adalah melakukan serangan jarak jauh sebagai salah satu bentuk *back up* serangan yang dilakukan oleh *NPCScout*.

***Hierarchy Finite State Machine* untuk Strategi Menyerang**

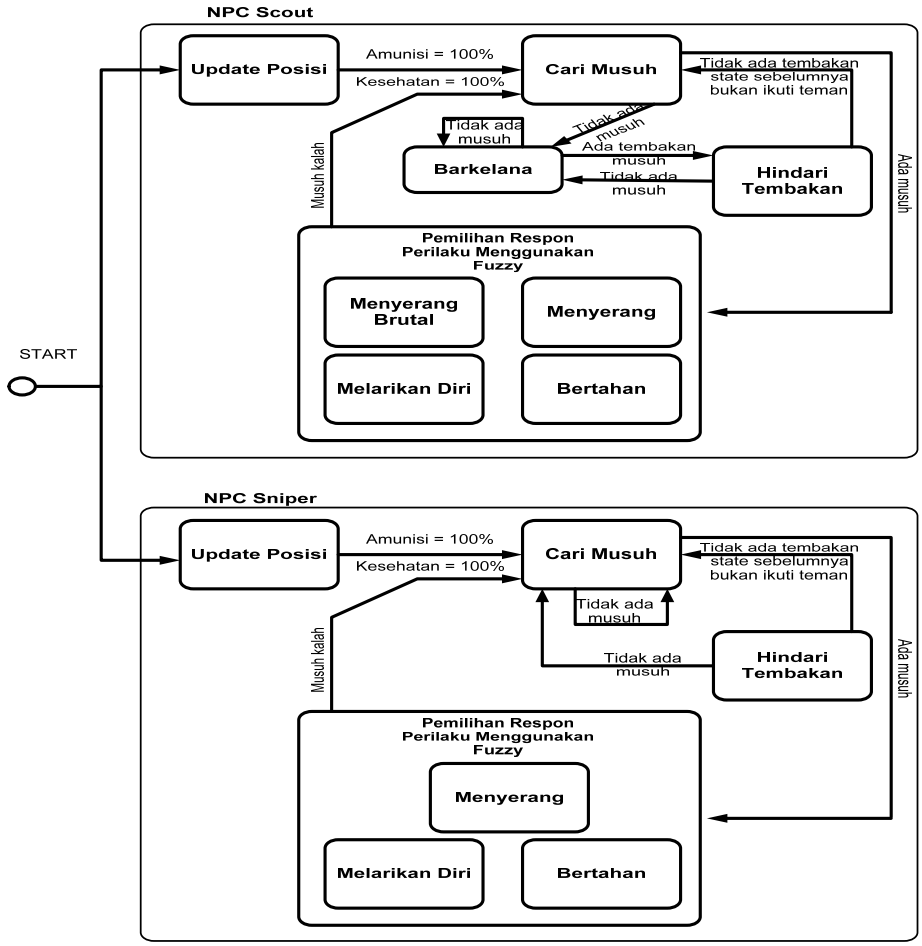
Secara garis besar hirarki strategi penyerangan terbagi menjadi dua bagian, yaitu hirarki manuver untuk *NPCScout* dan hirarki manuver untuk *NPCSniper*, seperti digambarkan pada gambar 3. Dengan menggunakan *Hierarchy Finite State Machine*, *NPCScout* mempunyai perilaku menyerang brutal, menyerang, melarikan diri dan bertahan; sedangkan *NPCSniper* mempunyai perilaku menyerang, melarikan diri dan bertahan, yang kemudian dijabarkan pada *Top Level Finite State Machine*.

Top Level Finite State Machine

Top level finite state machine untuk gerak menghindar pada *NPCScout* dan *NPCSniper* memiliki *FSM* yang sama untuk *update* posisi, cari musuh dan hindari tembakan, sebagaimana digambarkan pada Gambar 4. Perbedaan yang ada pada*FSM* berkelana dan *FSM*perubahan perilaku *NPC* yang memakai logika *fuzzy*.

Logika *Fuzzy*

Logika *fuzzy* dalam penelitian ini digunakan untuk menentukan variasi perilaku yang dilakukan oleh *NPC*, baik *NPCScout* maupun *NPCSniper*. Dengan adanya logika *fuzzy* tersebut masing-masing *NPC* dapat merubah perilaku berbasis perubahanvariabel masukan menjadi perilaku yang sudah dirancang menggunakan *HFSM*.

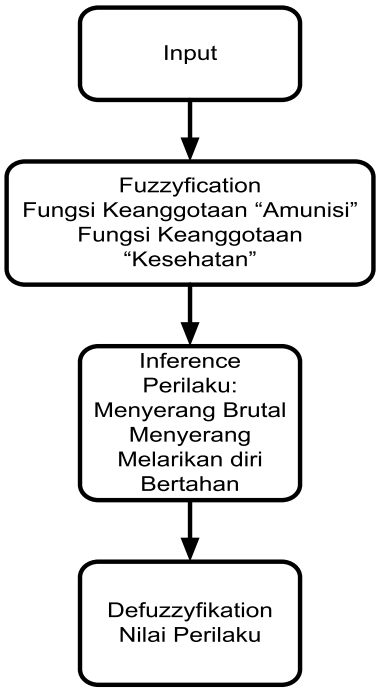


Gambar 4. *Top Level Finite State Machine* Strategi Menyerang.

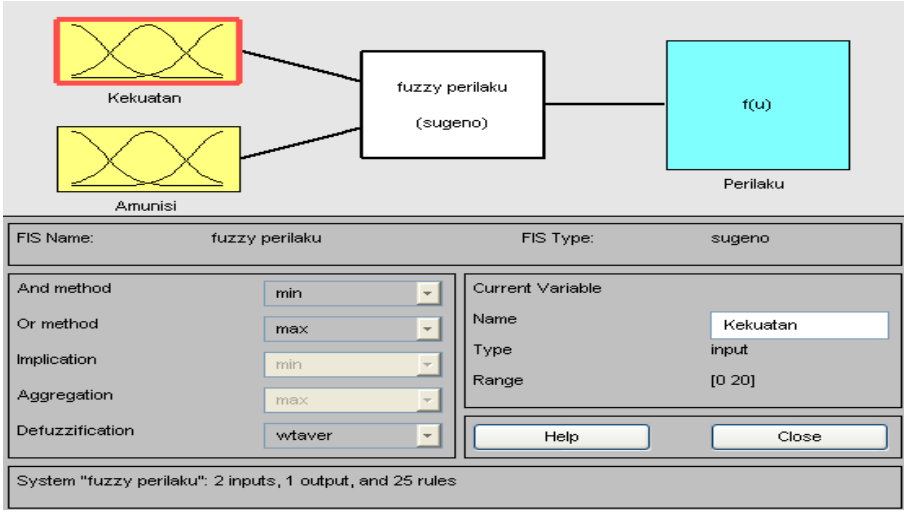
Metode *fuzzy* yang digunakan adalah metode Sugeno, karena metode ini menghasilkan keluaran yang berupa konstanta tegas, sehingga dapat mewakili nilai perilaku yang sudah dirancang sebelumnya.

Rancangan *Fuzzy NPCScout*

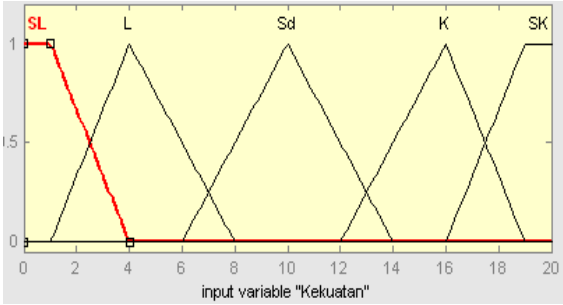
Untuk menghasilkan perilaku pada NPC Scout ada dua variabel yang digunakan, yaitu jumlah amunisi (sangat sedikit, sedikit, sedang, banyak, sangat banyak) dan kesehatan (sangat lemah, lemah, sedang, kuat, sangat kuat). Dengan menggunakan dua variabel diharapkan dapat menentukan variasi perilaku yang akan dilakukan. Untuk menjadi otonom maka digunakan aturan sebab akibat antara perilaku dengan atribut variabel yang menempel pada NPC. Misalnya ketika amunisi sedikit dan kesehatan lemah maka NPC cenderung melakukan perilaku melarikan diri sesuai dengan hasil defuzzyfikasi.



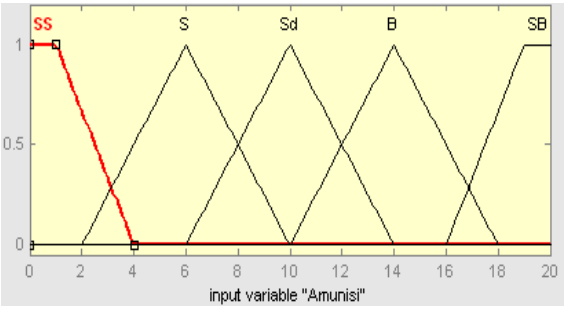
Gambar 5. Logika *Fuzzy* untuk Menghasilkan Perilaku.



Gambar 6. Rancangan*Fuzzy*untuk Menghasilkan Perilaku *NPCScout.*



Gambar 7. Derajat Keanggotaan Masukan Kekuatan *NPCScout.*



Gambar 8. Derajat Keanggotaan Masukan Amunisi *NPCScout.*

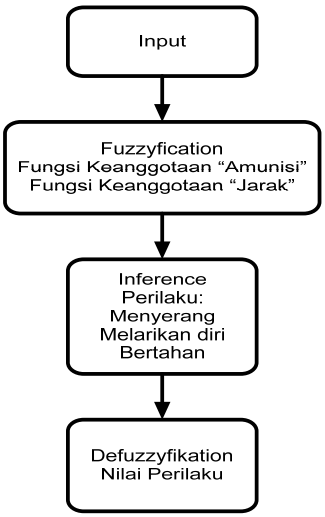
Tabel 1. Aturan *Fuzzy* untuk Menghasilkan Perilaku *NPC Scout.*

		Kekuatan				
	Variabel	SL	L	Sd	K	SK
Amunisi	SS	B	B	B	B	B
	S	B	B	L	L	MN
	Sd	B	L	MN	MN	MB
	B	L	MN	MN	MN	MB
	SB	L	MN	MN	MB	MB

Gambar 5 menunjukkan bahwa dua atribut yaitu amunisi dan kesehatan, digunakan dalam logika *fuzzy* masing-masing menggunakan gabungan-gabungan fungsi keanggotaan segitiga dan trapesium untuk membentuk perilaku *NPC*. Sedangkan rancangan*fuzzy* untuk menghasilkan perilaku *NPC* Scout dapat dilihat pada Gambar 6.

Pada Gambar 7 terlihat derajat keanggotaan masukan kekuatan *NPCScout* yang mempunyai interval antara 0 sampai 20. Variabel kekuatan mempunyai variabel linguistik: *sangat lemah, lemah, sedang, kuat, sangat kuat.* Variabel *sangat lemah (SL)* mempunyai interval antara 0 sampai 4. Untuk variabel*lemah (L)* mempunyai interval antara 1 sampai 8. Variabel *sedang (Sd)* mempunyai interval antara 6 sampai 14. Untuk variabel *kuat (K)* mempunyai interval antara 12 sampai 19. Sedangkan untuk variabel *sangat kuat (SK)* mempunyai interval antara 16 sampai 20.

Derajat keanggotaan variabel amunisi untuk *NPCScout* tergambar pada Gambar 8. Variabel amunisi mempunyai nilai linguistik:*sangat sedikit, sedikit, sedang, banyak, sangat banyak.* Interval variabel ini ditetapkan antara 0 sampai 20. Dimana untuk *sangat sedikit (SS)* mempunyai interval antara 0 sampai 4, untuk *sedikit (S)* mempunyai interval antara 2 sampai 10, untuk *sedang (Sd)* mempunyai interval antara 6 sampai 14. Sedangkan untuk *banyak (B)* mempunyai interval antara 12 sampai 18 dan untuk *sangat banyak(SB)* intervalnya adalah antara 16 sampai 20.



Gambar 9. Logika *Fuzzy* untuk Menghasilkan Perilaku *NPC Sniper*.

Tabel 2. Aturan *Fuzzy* Untuk Menghasilkan Perilaku *NPCSniper*.

		Jarak		
	Variabel	D	Sd	J
Amunisi	S	L	L	B
	Sd	B	B	B
	B	B	MN	MN

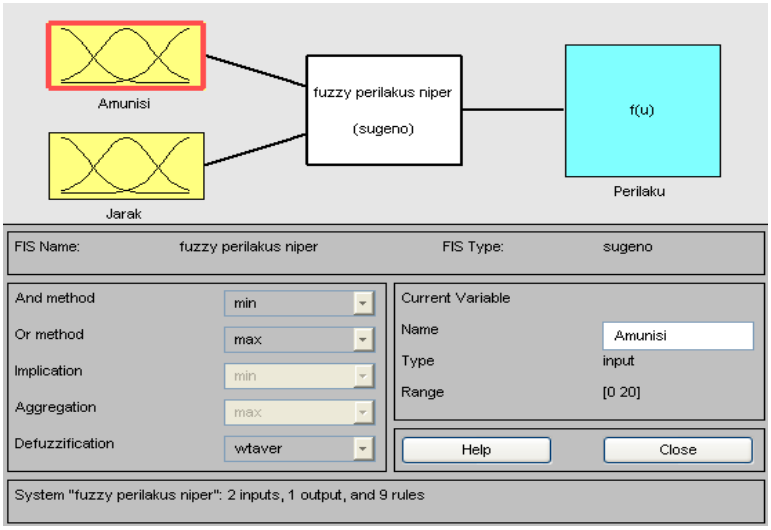
Untuk keluaran perilaku *NPCScout*, nilai linguistiknya dibagi menjadi empat, yaitu *menyerang brutal (MB), menyerang (MN), melarikan diri (L)*, dan *bertahan (B)*. *MB* mempunyai nilai 3, *MN* mempunyai nilai 2, *L* mempunyai nilai 1, dan *B* mempunyai nilai 0. Aturan *fuzzy* untuk perilaku *NPC* diperlihatkan pada Tabel 1.

Rancangan Fuzzy *NPCSniper*

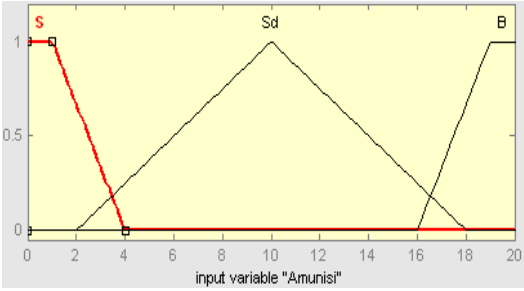
Untuk menghasilkan perilaku pada *NPC* Sniper ada dua variabel yang digunakan, yaitu jumlah amunisi *(sedikit, sedang, banyak)* dan jarak musuh *(dekat, sedang, jauh)*. Dengan menggunakan dua variabel diharapkan dapat menentukan variasi perilaku yang akan dilakukan. Untuk menjadi otonom maka digunakan aturan sebab akibat antara perilaku dengan atribut variabel yang menempel pada *NPC*. Misalnya ketika amunisi banyak dan *jarak musuh* jauh maka *NPC* cenderung melakukan perilaku menyerang sesuai dengan hasil *defuzzyfikasi*.

Gambar 9 menunjukkan bahwa dua atribut yaitu amunisi dan jarak, digunakan sebagai masukan dalam logika *fuzzy*. Masing-masing atribut menggunakan gabungan fungsi keanggotaan segitiga dan trapezium. Rancangan *fuzzy* untuk menghasilkan perilaku *NPCSniper* dapat dilihat pada Gambar 10.

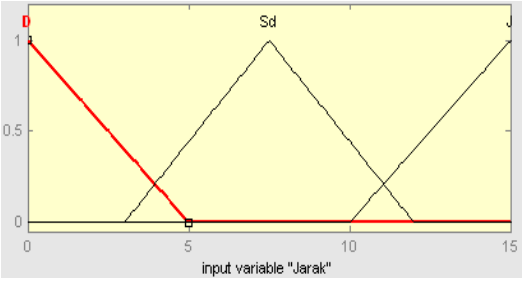
Pada Gambar 11 terlihat derajat keanggotaan masukan amunisi *NPCSniper* yang mempunyai interval antara 0 sampai 20. Variabel *amunisi* mempunyai variabel linguistik: *sedikit, sedang,*dan *banyak*. Variabel *sedikit(S)* mempunyai interval antara 0 sampai 4, sedangkan *sedang(Sd)* mempunyai interval antara 2 sampai 18, dan *banyak(B)* mempunyai interval antara 16 sampai 20.



Gambar 10.Rancangan*Fuzzy* untuk Menghasilkan Perilaku *NPCSniper*.



Gambar 11. Derajat Keanggotaan Masukan Amunisi *NPCSniper*.



Gambar 12. Derajat Keanggotaan Masukan Jarak *NPCSniper*.

Derajat keanggotaan variabel jarak untuk *NPCSniper* tergambar pada Gambar 12. Variabel jarak mempunyai nilai linguistik: *dekat*, *sedang*, dan *jauh*. Untuk *dekat(D)* mempunyai interval antara 0 sampai 5, *sedang (Sd)* mempunyai interval antara 3 sampai 12. Sedangkan untuk *jauh(J)* mempunyai interval antara 10 sampai 15.

Untuk keluaran perilaku *NPC Sniper* nilai linguistiknya dibagi menjadi tiga, yaitu: *menyerang (MN)*, *melarikan diri (L)*, dan *bertahan (B)*. Aturan *fuzzy* untuk perilaku *NPC Sniper* diperlihatkan pada Tabel 2.

HASIL DAN PEMBAHASAN

Pengujian Sistem

Pengujian diawali dengan pengujian logika *fuzzy* yang digunakan untuk menentukan variasi perilaku masing-masing *NPC* yang sudah dibangun menggunakan *HFSM* berdasarkan variabel masukan yang dimiliki. Selanjutnya hasil sistem *fuzzy* yang diperoleh, diujicobakan pada *game* menggunakan *Torque Game Engine* (www.torquepowered.com).

Pengujian *Fuzzy* Perilaku *NPCScout*

Sesuai variabel yang diberikan untuk menghasilkan perubahan perilaku *NPCScout*, maka beberapa parameter diujikan untuk mengetahui variasi perilaku yng dihasilkan. Parameter yang diuji` diantaranya adalah beberapa nilai variabel amunisi mulai dari minimum hingga maksimum dan juga beberapa nilai variabel kesehatan mulai dari minimum hingga maksimum.

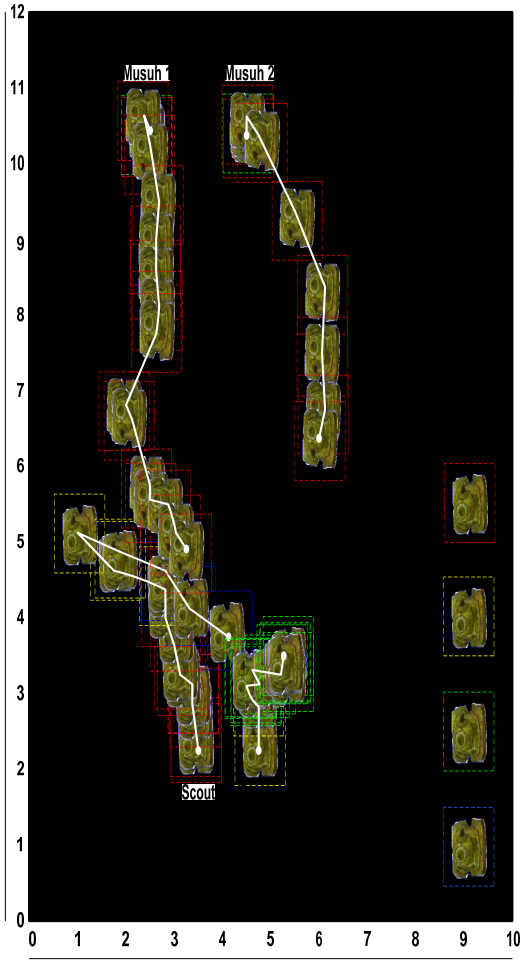
Tabel 3. Hasil Pengujian Perilaku *NPC Scout* dengan Menggunakan Variabel Parameter Masukan yang Berbeda.

		AMUNISI																				
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
KEKUATAN	0	0	0	0	0	0	0	0	0	0	0	0	0,25	0,5	0,75	1	1	1	1	1	1	1
	1	0	0	0	0	0	0	0	0	0	0	0	0,25	0,5	0,75	1	1	1	1	1	1	1
	2	0	0	0	0	0	0	0	0,167	0,167	0,222	0,333	0,722	0,9	1,06	1,33	1,33	1,4	1,5	1,33	1,33	1,33
	3	0	0	0	0	0	0	0	0,167	0,167	0,444	0,667	0,944	1,1	1,28	1,67	1,67	1,6	1,5	1,67	1,67	1,67
	4	0	0	0	0	0	0	0	0,25	0,25	0,75	1	1,25	1,5	1,75	2	2	2	2	2	2	2
	5	0	0	0	0	0	0	0	0,25	0,25	0,75	1	1,25	1,5	1,75	2	2	2	2	2	2	2
	6	0	0	0	0	0	0	0	0,333	0,5	0,667	1	1,33	1,5	1,67	2	2	2	2	2	2	2
	7	0	0	0	0,25	0,5	0,5	0,5	1	1	1	1,5	1,75	1,75	1,75	2	2	2	2	2	2	2
	8	0	0	0	0,429	1	1	1	1,33	1,5	1,67	2	2	2	2	2	2	2	2	2	2	2
	9	0	0	0	0,429	1	1	1	1,25	1,5	1,75	2	2	2	2	2	2	2	2	2	2	2
	10	0	0	0	0,429	1	1	1	1,25	1,5	1,75	2	2	2	2	2	2	2	2	2	2	2
	11	0	0	0	0,429	1	1	1	1,25	1,5	1,75	2	2	2	2	2	2	2	2	2	2	2
	12	0	0	0	0,429	1	1	1	1,33	1,5	1,67	2	2	2	2	2	2	2	2	2	2	2
	13	0	0	0	0,5	1	1	1	1,5	1,5	1,5	2	2	2	2	2	2	2	2,25	2,5	2,5	2,5
	14	0	0	0	0,429	1	1	1	1,33	1,5	1,67	2	2	2	2	2	2	2	2,57	3	3	3
	15	0	0	0	0,429	1	1	1	1,25	1,5	1,75	2	2	2	2	2	2	2	2,57	3	3	3
	16	0	0	0	0,429	1	1	1	1,25	1,5	1,75	2	2	2	2	2	2	2	2,57	3	3	3
	17	0	0	0	0,643	1,4	1,33	1,33	1,72	1,9	2,06	2,33	2,39	2,4	2,39	2,33	2,33	2,4	2,79	3	3	3
	18	0	0	0	0,643	1,6	1,67	1,94	1,94	2,1	2,28	2,67	2,61	2,6	2,61	2,67	2,67	2,6	2,79	3	3	3
	19	0	0	0	0,857	2	2	2	2,25	2,5	2,75	3	3	3	3	3	3	3	3	3	3	3
	20	0	0	0	0,857	2	2	2	2,25	2,5	2,75	3	3	3	3	3	3	3	3	3	3	3

Tabel 4. Hasil Pengujian *Fuzzy* Perilaku *NPCSniper* dengan Menggunakan Variabel Masukan yang Berbeda.

JARAK MUSUH \ AMUNISI	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,727	1	1	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,727	1	1	1
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,727	1	1	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,727	1	1	1
4	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,526	0,814	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	1	1	1	1,19	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47	1,47
12	1	1	1	1,27	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
13	1	1	1	1,27	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
14	1	1	1	1,27	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
15	1	1	1	1,27	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Gambar 9. Simulasi Pergerakan *NPC* Ketika Melawan 1 *NPC* Musuh.



Gambar 10. Simulasi Pergerakan *NPC* Ketika Melawan 2 *NPC* Musuh.

Dari variasi variabel masukan yang digambarkan pada Tabel 3 dapat diperoleh keluaran perilaku *NPCscout* yang variatif, perilaku-perilaku tersebut selanjutnya dikelompokkan menjadi empat model perilaku, yaitu bertahan(nilai 0 sampai 0,5), melarikan diri (nilai 0,5 sampai 1,5), menyerang(1,5 sampai 2,5) dan menyerang brutal (nilai 2,5 ke atas). Tabel 4 menyatakan keluaran perilaku *NPCSniper*. Pada pengujian *fuzzy* perilaku *NPCSniper*, ada dua variabel masukan yang diujikan, yaitu amunisi dan jarak musuh. Variabel amunisi diujikan mulai dari *sedikit*, *sedang*, hingga *banyak*. Sedangkan variabel jarak musuh diujikan mulai dari *dekat*, *sedang* hingga *jauh*. Perilaku *NPCSniper* dikelompokkan menjadi empat model perilaku yaitu bertahan(nilai 0 sampai 0,5), melarikan diri (nilai 0,5 sampai 1,5), menyerang(1,5 sampai 2,5) dan menyerang brutal (nilai 2,5 ke atas).

Percobaan Perilaku *NPCScout* dan *NPC Sniper* Melawan Musuh

Percobaan ini dilakukan untuk melihat apakah perilaku masing-masing *NPC* sudah sesuai dengan perilaku yang sudah dirancang menggunakan *HFSM* dan respon perubahan perilaku berdasarkan variabel masukan menggunakan logika *fuzzy*.

Dalam percobaan ini juga digunakan untuk menguji perubahan perilaku *NPC* berdasarkan variabel masukan yang dimiliki apabila hanya melawan satu *NPC* musuh saja, dan melawan dua musuh. Simulasi pergerakan dan perilaku *NPC* dapat dilihat pada Gambar 9 dan 10.

Dari hasil sepuluh kali percobaan yang dilakukan untuk menguji tingkat kemenangan ketika melawan dua *NPC* musuh tanpa menggunakan logika *fuzzy* didapatkan hasil seperti pada Tabel 5. Dimana tim *Scout* dan *Sniper* mendapatkan kemenangan sebanyak 8 kali sedangkan tim Musuh mendapatkan dua kali, sehingga tingkat kemenangan *NPCScout* dan *Sniper* adalah 80%.

SIMPULAN

Dari hasil uji coba penelitian ini dapat diperoleh beberapa simpulan, yaitu: *Hierarchy Finite State Machine* digunakan untuk merancang perilaku manuver masing-masing *NPC* dalam merancang strategi menyerang, aturan *fuzzy* dapat diterapkan untuk menghasilkan perilaku *NPC* yang bervariasi sesuai dengan variabel masukan yang dimiliki, dan tingkat kemenangan strategi menyerang dengan logika *fuzzy* pada penelitian ini mencapai 80% jika melawan musuh yang mempunyai perilaku menyerang tanpa logika *fuzzy*.

Tabel 5. Hasil Tim *NPCScout* dan *NPCSniper* melawan *NPC* Musuh.

Percobaan	Tim *NPCScout* dan *NPCSniper*	Tim *NPC* Musuh 1 dan *NPC* Musuh 2
1	Menang	Kalah
2	Menang	Kalah
3	Kalah	Menang
4	Menang	Kalah
5	Menang	Kalah
6	Kalah	Menang
7	Menang	Kalah
8	Menang	Kalah
9	Menang	Kalah
10	Menang	Kalah

DAFTAR PUSTAKA

[1] Yunitarini R, Rancang Bangun Sistem Agen Cerdas Monitoring Stok Perusahaan, Jurnal KURSOR, 5 (1) : 45-58, 2009.

[2] Hon JH, and Cho S, Evolving Reactive NPCs for the Real-Time Simulation Game. *IEEE Symposium on Computational Intelligence and Games*, pp. 86-93, 2005.

[3] Mclean A, Hunting Down the Player in a Convincing Manner, Pivotal Games, *AI Game Programming Wisdom 2*, Massachusset, Charles River Media, 2003.

[4] McPartland M,and Gallagher M, Creating a Multi-Purpose First Person Shooter Bot with Reinforcement Learning, *IEEE*

Symposium on Computational Intelligence and Games, pp. 143-150,2008.

[5] Girault A, Lee B, and Lee EA, Fellow. Hierarchical Finite State Machines with Multiple Concurrency Models. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 18(6): 742-760, 1999.

[6] Song W, Cho K, and Um K, Motivation-based Hierarchical Behavior Planning, *Journal of Korea Game Society*, pp. 79-90,2008.