

PENENTUAN JALUR PERGERAKAN DAN PERILAKU PERANG KARAKTER BUKAN PEMAIN MENGGUNAKAN ALGORITMA A* DAN METODE LOGIKA FUZZY PADA GAME SIMULATOR TANK

Wali Muhammad¹, Vincent Suhartono², M. Arief Soeleman³
^{1,2,3}Pascasarjana Teknik Informatika Universitas Dian Nuswantoro

Abstract

Simulation is the process of implementation of the model into a computer program (software) with / without electronic circuits, as well as executes the software that imitates or resembles the behavior of real systems specific. Tank Simulator games are one type of simulation is to be a vehicle for the train crew in the operation and practice of war using tanks combat vehicles. Tank Simulator game is enabled to mimic the movement of tanks and or any type of vehicle involved in this simulation is run on terrain that resembles the actual situation and assisted with visualization software. Tank Simulator Game will be good when it is applied Artificial Intelligence (AI) engine well on his game. AI in a game simulator generally consists of the calculation of the movement path selection detects and moves according to the presence of an opponent and do battle tactics in accordance with the existing circumstances and conditions. It is very important in Tank Simulator Game is the application of AI in the tank instead of the player character or a non-playing character (NPC). To determine the selection of tank track instead of the player will use the path is as real as the actual field exercises that are used in the clinic. In terms of tactics behave war, tank instead of the player having a specific strategy such as that tank piloted by humans. Intended strategy is a strategy for the opponent, patrolling the area alone, silent, attack, defend, and run.

*Keywords: Tank Simulator Game, A *, Fuzzy logic, NPC.*

1. PENDAHULUAN

1.1 Latar Belakang

Game Simulator Tank (GST) adalah salah satu teknik pembelajaran yang menggabungkan beberapa unsur yang ada pada *game* perang dan sistem simulasi untuk mendapatkan hasil yang lebih baik. GST ini dapat berfungsi untuk menirukan gerakan tank asli dengan peralatan, tombol dan komponen didalamnya yang menyerupai barang aslinya dan dapat digunakan untuk pembelajaran pengemudian dan pengoperasian tank baik untuk operasi biasa maupun untuk pengoperasian dengan setrategi perang. Disamping itu pada GST dalam melakukan peperangan menggunakan strategi perang dan pola tempur yang baku, seperti dalam melakukan peperangan perlu diperhitungkan jalur terbaik yang harus dilalui, juga kondisi fisik tank, jumlah amunisi yang tersedia, jumlah lawan dan jumlah kawan. Pada sistem pengoperasiannya GST dibantu oleh visualisasi *software*.

Algoritma A* adalah algoritma pencarian pada graf yang mencari jalur dari satu verteks (simpul) awal ke sebuah verteks akhir yang telah ditentukan. Algoritma A* menggunakan pendekatan heuristik yang memberikan peringkat ke tiap-tiap verteks yang direpresentasikan dengan titik, dengan cara memperkirakan rute terbaik yang dapat dilalui dari verteks tersebut. Setelah itu tiap-tiap verteks tersebut dicek satu-persatu berdasarkan urutan yang dibuat dengan pendekatan heuristik tersebut. Algoritma A* sukses besar dalam menemukan jalur jalan pada sebuah *game*, karena itu banyak peneliti yang menaruh harapan untuk mengoptimalkan proses pencarian menggunakan algoritma A* untuk memenuhi perubahan kebutuhan *game*.

Logika fuzzy merupakan salah satu dari *Artificial Intelligent (AI)* yang memiliki kemampuan untuk menyelesaikan masalah perilaku sistem yang kompleks dan memiliki ketidak pastian, sehingga apabila

diterapkan pada sebuah sistem, logika fuzzy dapat mengatasi proses dari suatu sistem yang dinamik, kompleks dan memiliki ketidak pastian yang disebabkan oleh kemampuan mengolah informasi numerik dari variabel yang diukur. Dalam hal perilaku perang, Karakter Bukan Pemain (KBP) harus mempunyai strategi khusus seperti halnya para ahli perang bertempur secara riil di medan perang. Strategi yang dimaksud adalah berupa strategi mencari lawan, menyerang lawan, menghindari lawan, bertahan dalam arti berpatroli di area sendiri dan baru membalas bila diserang, diam, dan lari. Dalam pengambilan keputusan untuk berperilaku perang ini digunakan metode logika fuzzy.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah disebutkan dalam uraian sebelumnya maka dapat dirumuskan beberapa permasalahan sebagai berikut.

a. Masalah Umum

Beberapa teknik pembelajaran perang khususnya dengan menggunakan kendaraan tempur tank yang ada banyak yang belum memenuhi sasaran, lokasi dan situasi latihan tidak menyerupai dengan keadaan sebenarnya seperti bila peperangan dilakukan di medan tempur nyata.

b. Masalah Spesifik

- 1) Penentuan pemilihan jalur (path) pada pergerakan KBP belum memperhitungkan jalur jalan riil sesuai pada medan latihan sebenarnya, seperti jalan berlumpur, ada genangan air/sungai.
- 2) Kurang akuratnya penerapan strategi perang yang dipakai, untuk menentukan respon perilaku terhadap setiap perubahan dalam memberikan rekomendasi langkah- langkah yang harus dilakukan bagi Karakter Bukan Pemain dalam menghadapi situasi dan kondisi yang sedang terjadi.
- 3) Beberapa penelitian terkait dalam melakukan strategi perang belum memperhitungkan jumlah kawan dan/atau lawan.

1.3 Tujuan Penelitian

a. Tujuan Umum

Untuk lebih mempertajam hasil dari sebuah pembelajaran di bidang Militer khususnya dalam hal pengoperasian dan latihan perang menggunakan tank tiruan yang dilakukan di medan perang yang menyerupai keadaan riil agar memenuhi sasaran pembelajaran.

b. Tujuan Spesifik

- 1) Agar dalam melakukan pemilihan jalur pergerakan, Karakter Bukan Pemain sudah memperhitungkan semua ruang keadaan (state) jalur jalan dan medan tempur yang ada.
- 2) Menentukan algoritma yang paling optimal untuk diterapkan pada *Game Simulator Tank* dalam hal penentuan jalur pergerakan dan perilaku taktis peperangan.
- 3) Memperhitungkan jumlah kawan dan lawan dalam menentukan setiap pergerakan dan strategi taktis dalam peperangan, karena dalam perang modern seperti sekarang ini teknologi yang dipakai serba canggih.

1.4 Manfaat Penelitian

a. Manfaat bagi TNI dalam Pelatihan

Penerapan dari hasil penelitian ini nantinya adalah untuk Simulator Terintegrasi Tank yang akan digunakan oleh TNI angkatan darat dalam melatih para kru kendaraan tempur tank untuk meningkatkan kemampuannya dalam mengoperasikan tank baik untuk situasi damai maupun situasi

perang dalam rangka menjaga keamanan dan kesatuan wilayah Negara Kesatuan Republik Indonesia.

b. Manfaat bagi Ilmu Pengetahuan dan Teknologi

- 1) Untuk pengembangan teori yang berkaitan dengan perhitungan pemilihan jalur pergerakan (pathfinding) karakter (kendaraan tempur) bukan pemain menggunakan algoritma A*.
- 2) Sebagai salah satu model penentuan perilaku perang Karakter Bukan Pemain menggunakan metode Logika Fuzzy.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian lain yang terkait dengan penelitian ini antara lain adalah:

1. *Xiao Cui and Hao Shi Journal.*
“A*-based Pathfinding in Modern Computer Games”.
School of Engineering and Science, Victoria University, Melbourne, Australia, Januari 2011.
2. *Yifan Li, Petr Musilek and Loren Wyard-Scott Journal.*
“Fuzzy logic in Agent-Based Game Design”.
Department of Electrical and Computer Engineering University of Alberta Edmonton, AB T6G 2V4, Canada, 2004.
3. Jurnal: Yunifa Miftachul Arif, Mochamad Hariadi, Supeno Mardi S, N
“Strategi Menyerang pada Game FPS Menggunakan Hierarchy Finite State Machine dan Logika Fuzzy”.
Pasca Sarjana Teknik Elektro ITS, Surabaya 60111, 2011.
4. *Yung-Ping Yung-Ping Fang, I-Hsien Ting, Journal*
“Applying Reinforcement Learning fo Game AI in Tank-Battle Game”.
Department, Department of Information Management National University of Kaohsiung, No. 700, Kaohsiung University Road, Kaohsiung, 811 Taiwan, 2010.

2.2. Landasan Teori

2.2.1 Algoritma A* Pathfinding

Algoritma A* merupakan salah satu algoritma pencarian *graph* terbaik yang mampu menemukan jalur dengan biaya pengeluaran paling sedikit dari titik permulaan yang diberikan sampai ke titik tujuan yang diharapkan. Algoritma A* dapat diimplementasikan lebih efisien karena tidak ada simpul yang diproses berulang-ulang. Algoritma ini memeriksa simpul dengan menggabungkan $g(n)$, yaitu *cost* (biaya) yang dibutuhkan untuk mencapai sebuah simpul dan $h(n)$ yaitu *cost* yang didapat dari simpul ke tujuan. Sehingga dapat dirumuskan sebagai:

$$f(n) = g(n) + h(n) \quad (1)$$

Keterangan :

$f(n)$ adalah total biaya (cost) sebuah jalur (path) dari simpul awal ke simpul tujuan (goal) melalui simpul n .

$g(n)$ adalah biaya (cost) yang dibutuhkan oleh sebuah jalur (path) untuk mencapai simpul n dari simpul awal.

$h(n)$ adalah biaya (cost) sebuah jalur (path) .

2.2.2 Fungsi Heuristic

Ada beberapa fungsi heuristic umum yang bisa dipakai untuk algoritma A* ini. Salah satunya adalah yang dikenal dengan istilah Manhattan Distance. Fungsi heuristic ini digunakan untuk kasus dimana pergerakan pada peta hanya lurus (horizontal atau vertikal), tidak diperbolehkan pergerakan diagonal.

Perhitungan nilai heuristic untuk node ke-n menggunakan Manhattan Distance adalah sebagai berikut.

$$h(n) = (\text{abs}(n.x - \text{goal}.x) + \text{abs}(n.y - \text{goal}.y)) \quad (2)$$

Keterangan: $h(n)$ adalah nilai *heuristic* untuk *node* n, dan *goal* adalah *node* tujuan.

Jika pergerakan diagonal pada peta diperbolehkan, maka digunakan fungsi heuristic selain Manhattan Distance. Untuk mendekati kenyataan, *cost* untuk perpindahan node secara diagonal dan orthogonal dibedakan. *Cost* diagonal adalah 1,4 kali *cost* perpindahan secara orthogonal.

Maka fungsi heuristic yang digunakan adalah sebagai berikut.

$$\begin{aligned} h_{\text{diagonal}}(n) &= \min(\text{abs}(n.x - \text{goal}.x) + \text{abs}(n.y - \text{goal}.y)) \\ h_{\text{orthogonal}}(n) &= (\text{abs}(n.x - \text{goal}.x) + \text{abs}(n.y - \text{goal}.y)) \\ h(n) &= d2 * h_{\text{diagonal}}(n) + d1 * (h_{\text{orthogonal}}(n) - (2 * h_{\text{diagonal}}(n))) \end{aligned} \quad (3)$$

Keterangan:

- $h(n)$: biaya (cost) heuristic untuk node n
- $h_{\text{diagonal}}(n)$: banyaknya langkah diagonal yang bisa diambil untuk mencapai goal dari node n
- $h_{\text{orthogonal}}(n)$: banyaknya langkah lurus yang bisa diambil untuk mencapai goal dari node n.
- $n.x$ & $n.y$: biaya koordinat x & y pada node n
- $\text{goal}.x$: biaya koordinat x pada node tujuan
- $\text{goal}.y$: biaya koordinat y pada node tujuan
- $d1$: biaya perpindahan node secara orthogonal
- $d2$: biaya perpindahan node secara diagonal

2.2.3 Jarak Euclid dalam Algoritma A*

Dalam mengaplikasikan Algoritma A* pada penelitian ini, fungsi heuristic yang digunakan adalah jarak Euclid. Heuristik ini akan menghitung jarak berdasarkan panjang garis yang dapat ditarik dari dua buah titik. Perhitungannya dapat dituliskan sebagai berikut :

$$h(n) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (4)$$

2.2.4 Logika Fuzzy (*Fuzzy logic*)

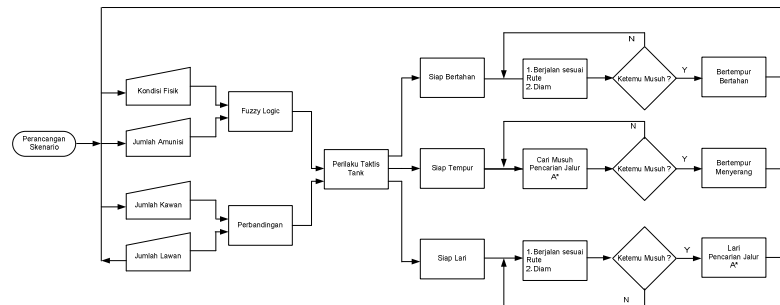
Logika Fuzzy bukan merupakan suatu ilmu yang baru, konsep tentang fuzzy itu sendiri sudah ada sejak lama namun ilmu tentang logika fuzzy modern dan metodis baru ditemukan beberapa tahun yang lalu. Logika Fuzzy adalah metode pemecahan masalah dengan beribu – ribu aplikasi dalam pengendali yang tersimpan dan pemrosesan informasi. Logika Fuzzy menyediakan cara sederhana untuk menggambarkan kesimpulan pasti dari informasi yang ambigu, samar-samar, atau tidak tepat. Sedikit banyak Logika Fuzzy menyerupai pembuatan keputusan pada manusia dengan kemampuannya untuk bekerja dari data yang ditafsirkan dan mencari solusi yang tepat.

Dalam sistem Logika Fuzzy terdapat beberapa tahapan operasional yang meliputi:

- a. Fuzzyfikasi
- b. Aturan Dasar (*Rule Based*)
- c. Komposisi Aturan
- d. Defuzzyfikasi

2.3. Penentuan Jalur Pergerakan dan Perilaku Perang Karakter Bukan Pemain Menggunakan Algoritma A* dan Metode Logika Fuzzy pada *Game Simulator Tank*

Perancangan sistem yang dipakai secara umum adalah berawal dari permasalahan dimana pengembangan perangkat lunak khususnya untuk *Game Simulator Tank* masih belum optimal dalam penerapan algoritma sistem cerdas. Kerangka pemikiran dalam hal penggunaan algoritma pada pengembangan perangkat lunak secara garis besar konsep desainnya dapat ditunjukkan pada gambar 1.



Gambar 1. Diagram Penerapan Algoritma pada *Game Simulator Tank*

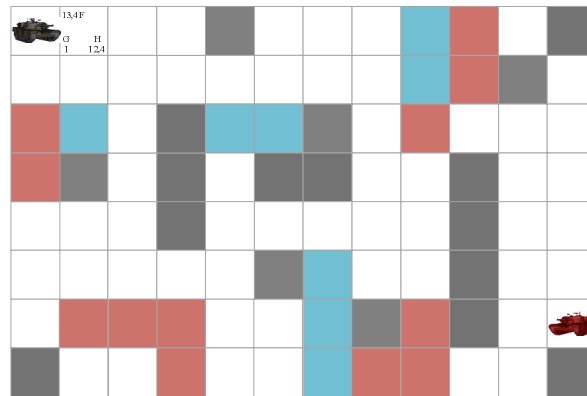
Dari gambar 1 di atas didapat bahwa keputusan metode logika fuzzy diambil dari beberapa hal, seperti perhitungan jumlah kawan, jumlah lawan, kesehatan tank, dan jumlah amunisi Tank. Keputusan tersebut akan menghasilkan sebuah kondisi yang akan digunakan sebagai masukan dalam melaksanakan aksi selanjutnya, yaitu bertahan (sambil mencari bantuan atau siap lari), siap tempur dan lari. Kondisi tersebut akan menghasilkan suatu tindakan yang berbeda bila tank bukan pemain akan bertemu dengan musuhnya. Apabila tank bukan pemain berada pada kondisi siap tempur, maka tank tersebut akan menggunakan algoritma A* untuk mencari jalur yang terbaik untuk mencari musuhnya dengan memperhitungkan kondisi jalan yang akan dilalui berdasarkan peta ruang keadaan yang ada.

3. METODE PENELITIAN DAN PENGEMBANGAN

3.1. Analisis Pergerakan A* dengan Memperhatikan Kondisi Lingkungan







Pada kasus ini akan dianalisis penerapan Algoritma A* pada jalan dengan memperhitungkan kondisi lingkungan seperti lumpur, sungai, bangunan dan pohon untuk pencarian jalan terbaik dan/atau waktu tercepat menuju tujuan dengan menghitung jarak langkah dan simpul yang diperiksa Ada berbagai macam rintangan yang ditemukan dalam sebuah medan peperangan yang digambarkan pada gambar 2. Kemampuan KBP untuk dapat melewatinya pun juga harus diperhitungkan. Untuk itu ada mekanisme penghitungan beban yang harus diperkirakan, agar biaya yang didapat adalah optimal. Dalam hal ini biaya perpindahan secara diagonal dan orthogonal dibedakan.

Sebagai contoh, pada rancangan peta berbentuk kotak berikut mempunyai berbagai rintangan yang berbeda-beda yang akan dianalisis bagaimana pengaruh pergerakan KBP dalam memilih jalur yang akan dilewati. Peta berikut mempunyai ordo 8 x 12, dimana pergerakan sumbu X adalah 8 dan pergerakan sumbu Y adalah 12. Untuk lebih jelasnya, gambaran peta yang terdapat beberapa macam lingkungan dapat dilihat pada gambar 2 berikut.



Gambar 2. Peta dengan Berbagai Kondisi Lingkungan

Keterangan gambar:

	Kotak Putih	:	Tanpa rintangan	Heuristic = 1
	Kotak Coklat	:	Lumpur	Heuristic = 1.5
	Kotak Biru	:	Sungai	Heuristic = 2
	Kotak Hitam	:	Bangunan, Pohon, Kend. lain	Tidak dapat dilewati
	Posisi Awal			
	Posisi Tujuan			

Pada gambar 2 di atas terdapat kotak berisi tank warna hitam di sebelah kiri atas adalah titik awal, dan kotak berisi tank warna merah di sebelah kanan bawah adalah titik tujuan. Angka-angka dalam kotak masing-masing adalah nilai f (kiri atas), g (kiri bawah) dan h (kanan bawah). Untuk pergerakan horizontal dan vertikal, beayanya (cost) adalah 1, sedangkan untuk pergerakan diagonal beayanya adalah 1,4 yang berasal dari $\sqrt{2}$ dengan pembulatan 1 angka di belakang desimal.

Pada kotak di sebelah kanan simpul awal, g bernilai 1 dan h bernilai 12,4. Perhitungannya cukup sederhana, yaitu g bernilai 1 sebab dari simpul awal ke kotak tersebut hanya perlu bergerak horizontal satu kali. h bernilai 12,4 didapat dari banyaknya langkah yang diperlukan untuk mencapai tujuan dari kotak tersebut. f bernilai 13,4 adalah merupakan jumlah dari g (1) dan h (12,4). Selanjutnya, algoritma ini akan memeriksa kotak mana yang bernilai f-nya terkecil dan mengembangkan jalan di sekitar kotak tersebut.

Pada iterasi pertama ada tiga jalur yang ditemukan, yaitu pada koordinat (1,7), (2,7) dan (2,8). Di sini dihitung nilai f(n), g(n) dan h(n) untuk masing-masing jalur. Namun disini hanya akan dibahas satu jalur saja, sedangkan perhitungan jalur lainnya tidak dibahas karena dianggap penghitungannya similar.

Koordinat (2, 8)

Mekanisme perhitungan h(n):

$$\begin{aligned}
 h_{\text{diagonal}}(n) &= \min(\text{abs}(n.x - \text{tujuan}.x), \text{abs}(n.y - \text{tujuan}.y)) \\
 &= \min(\text{abs}(2-12), \text{abs}(8-2)) \\
 &= \min(10, 6) \\
 &= 6 \\
 h_{\text{orthogonal}}(n) &= (\text{abs}(n.x - \text{tujuan}.x) + \text{abs}(n.y - \text{tujuan}.y)) \\
 &= \text{abs}(2-12) + \text{abs}(8-2) \\
 &= 10 + 6
 \end{aligned}$$

$$= 16$$

$$\begin{aligned}h(n) &= d2 * h_diagonal(n) + d1 * (h_orthogonal(n) - 2 * h_diagonal(n)) \\&= 1,4 * 6 + 1 * (16 - 2 * 6) \\&= 8,4 + 4 \\&= 12,4\end{aligned}$$

Mekanisme Perhitungan f(n):

$$\begin{aligned}f(n) &= g(n) + h(n) \\&= 1 + 12,4 \\&= 13,4\end{aligned}$$

Iterasi kedua

Koordinat (1, 6):

Karena g(n) sebelumnya pada koordinat (2, 7) dengan nilai 1,4, kemudian karena contoh yang dihitung adalah secara diagonal dan lumpur, maka *cost* yang dipakai adalah 2,9, yang berasal dari penjumlahan 1,5 (cost lumpur) + 1,4 (cost diagonal).

$$\begin{aligned}g(n) &= g(n) \text{ sebelumnya} + 2,8 \\&= 1,4 + 2,8 \\&= 4,2\end{aligned}$$

Mekanisme perhitungan h(n):

$$\begin{aligned}h_diagonal(n) &= \min(\text{abs}(n.x - \text{tujuan}.x), \text{abs}(n.y - \text{tujuan}.y)) \\&= \min(\text{abs}(1-12), \text{abs}(6-2)) \\&= \min(11, 4) \\&= 4\end{aligned}$$

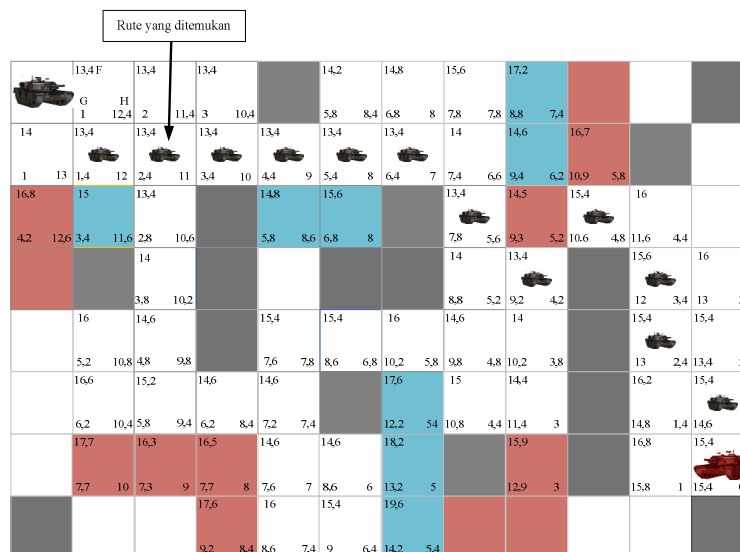
$$\begin{aligned}h_orthogonal(n) &= \text{abs}(n.x - \text{tujuan}.x) + \text{abs}(n.y - \text{tujuan}.y) \\&= \text{abs}(1-12) + \text{abs}(6-2) \\&= 11 + 4 \\&= 15\end{aligned}$$

$$\begin{aligned}h(n) &= d2 * h_diagonal(n) + d1 * (h_orthogonal(n) - 2 * h_diagonal(n)) \\&= 1,4 * 4 + 1 * (15 - 2 * 4) \\&= 5,6 + 7 \\&= 12,6\end{aligned}$$

Mekanisme Perhitungan f(n):

$$\begin{aligned}f(n) &= g(n) + h(n) \\&= 4,2 + 12,6 \\&= 16,8\end{aligned}$$

Dengan cara seperti ini, setelah dilakukan iterasi ke tigapuluh dua proses pencarian jalur selesai karena telah ditemukan jalur menuju ke target dengan biaya 15,4. Karena jalur paling optimal dari awal (1, 8) dan tujuan (12, 2) sudah ditemukan, KBP akan bergerak menuju tujuan seperti terlihat pada gambar 3.



Gambar 3. Proses hasil akhir pencarian A*

3.2. Analisis dan Perancangan Status Kesehatan Tank dengan Logika Fuzzy

Status Kesehatan Tank (SKT) merupakan cara untuk memantau status berdasarkan kesiapan tank untuk terus melakukan peperangan, khususnya yang berkaitan dengan kondisi fisik tank dan jumlah amunisi yang masih tersedia. Penggunaan Status kesehatan tank hanya berlaku untuk tank dan tidak untuk kendaraan tempur lain seperti truk pengangkut pasukan, ranpur, dan kendaraan selain tank lainnya.

Penulis mengasumsikan bahwa status kesehatan tank dibagi menjadi 5 (lima) kategori, kategori ini dicapai bila tank yang bersangkutan telah terkena tembakan pada area yang bukan vital (area depan dan samping), sekali terkena tembakan kesehatan tank akan berkurang 20%, sehingga maksimal dari tembakan yang boleh mengenai tank adalah 5 kali. Adapun kalau area yang terkena tembakan adalah bagian vital yang dalam hal ini diwakili dengan titik pusat tank atau area belakang yang merupakan tempat penyimpanan bahan bakar, maka kesehatan tank tersebut akan langsung berubah menjadi 0 alias rusak. Kategori status kesehatan tank bisa dilihat pada Tabel 1 berikut.

Tabel 1. Kategori Status Kesehatan Tank

Nilai SKT	Kategori	Status Kesehatan Tank
80 - 100	Kesehatan tank sangat baik	Baik
60 - 80	Kesehatan tank cukup baik	Cukup Baik
40 - 60	Kesehatan tank masih standar/normal	Normal
20 - 40	Kesehatan tank rusak tingkat ringan	Agak Rusak
1 - 20	Kesehatan tank rusak tingkat berat	Rusak

Untuk mendapatkan *output* berupa Status Kesehatan Tank, diperlukan empat tahapan sebagai berikut.

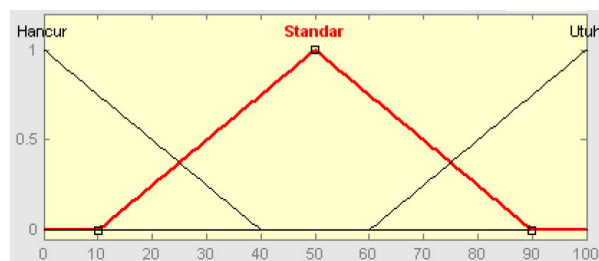
3.2.1 Pembentukan Himpunan Fuzzy

Pada Metode Mamdani, baik variabel *input* maupun variabel *output* dibagi menjadi satu atau lebih himpunan fuzzy. Dalam penentuan Status Kesehatan Tank (SKT), variabel *input* dibagi menjadi dua yaitu variabel Kondisi Fisik dan Jumlah Amunisi.

Himpunan fuzzy beserta fungsi keanggotaan dari variabel Kondisi Fisik, Jumlah Amunisi, dan hasil keluaran Status Kesehatan Tank direpresentasikan sebagai berikut:

1. Himpunan Fuzzy Variabel Kondisi Fisik

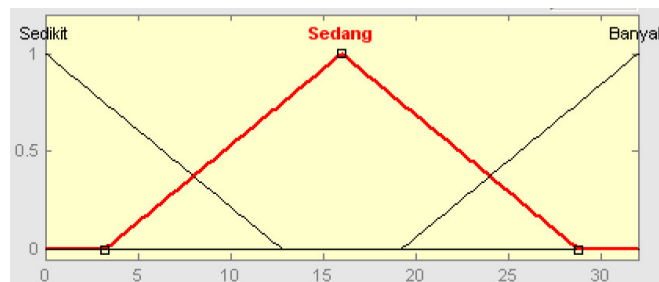
Pada variabel Kondisi Fisik didefinisikan tiga himpunan fuzzy, yaitu HANCUR, STANDAR, dan UTUH. Untuk merepresentasikan variabel Kondisi Fisik digunakan bentuk kurva bahu kiri untuk himpunan fuzzy HANCUR, bentuk kurva segitiga untuk himpunan fuzzy STANDAR, dan bentuk kurva bahu kanan untuk himpunan fuzzy UTUH. Gambar himpunan fuzzy untuk variabel Kondisi Fisik ditunjukkan pada gambar 4.



Gambar 4. Himpunan fuzzy Kondisi Fisik

2. Himpunan Fuzzy Variabel Jumlah Amunisi

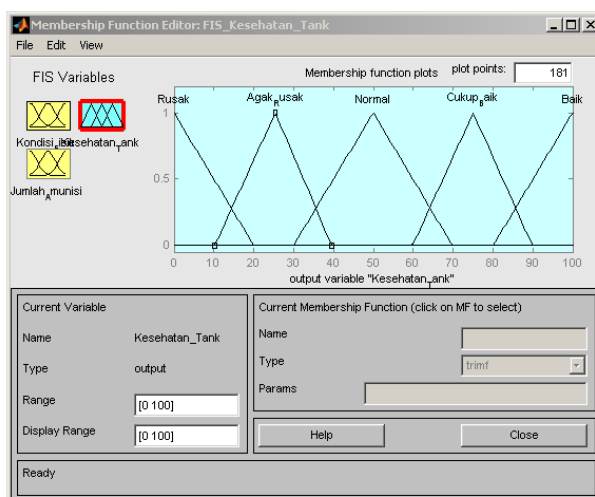
Pada variabel Jumlah Amunisi didefinisikan tiga himpunan fuzzy, yaitu SEDIKIT, SEDANG, dan BANYAK. Untuk merepresentasikan variabel Jumlah Amunisi digunakan bentuk kurva bahu kiri untuk himpunan fuzzy SEDIKIT, bentuk kurva segitiga untuk himpunan fuzzy SEDANG, dan bentuk kurva bahu kanan untuk himpunan fuzzy BANYAK. Representasi himpunan fuzzy untuk variabel Jumlah Amunisi ditunjukkan pada gambar 5.



Gambar 5. Himpunan Fuzzy: Jumlah Amunisi

3. Himpunan Fuzzy Variabel Status Kesehatan Tank

Pada variabel Status Kesehatan Tank didefinisikan lima himpunan fuzzy, yaitu RUSAK, AGAK RUSAK, NORMAL, CUKUP BAIK dan BAIK. Untuk merepresentasikan variabel Status Kesehatan Tank digunakan bentuk kurva bahu kiri untuk himpunan fuzzy RUSAK, bentuk kurva segitiga untuk himpunan fuzzy AGAK RUSAK, NORMAL dan CUKUP, dan bentuk kurva bahu kanan untuk himpunan fuzzy BAIK. Representasi himpunan fuzzy untuk variabel Status Kesehatan Tank ditunjukkan pada gambar 6.



Gambar 6. Representasi Variable Status Kesehatan Tank

Setelah pembentukan fungsi keanggotaan pada masing–masing variabel, *input* yang berupa nilai crisp akan diubah ke dalam *fuzzy input* yaitu dengan menentukan derajat keanggotaan nilai *input* pada sebuah himpunan fuzzy, proses ini disebut fuzzyfikasi.

3.2.2 Aplikasi Implikasi Fungsi

Setelah pembentukan himpunan fuzzy, maka dilakukan pembentukan aturan fuzzy. Aturan-aturan dibentuk untuk menyatakan relasi antara *input* dan *output*. Tiap aturan merupakan suatu implikasi. Operator yang digunakan untuk menghubungkan antara dua *input* adalah operator AND, dan yang memetakan antara *input-output* adalah IF-THEN. Proposisi yang mengikuti IF disebut anteseden, sedangkan proposisi yang mengikuti THEN disebut konsekuen.

Berdasarkan kategori dalam SKT, maka dapat dibentuk aturan–aturan seperti terlihat pada Tabel 2 berikut.

Tabel 2. Aturan–aturan dalam Penentuan Status Kesehatan Tank

		Kondisi Fisik		
		Rusak	Standar	Utuh
Jumlah Amunisi	Sedikit	Rusak	Agak Rusak	Normal
	Sedang	Agak Rusak	Normal	Cukup Baik
	Banyak	Normal	Cukup Baik	Baik

Setelah penentuan fungsi keanggotaan variabel, maka dilakukan pembentukan aturan logika fuzzy. Berdasarkan data yang ada, dapat dibentuk aturan-aturan sebagai berikut :

[R1]: Jika Kondisi Fisik adalah HANCUR dan Jumlah Amunisi adalah SEDIKIT maka Status Kesehatan Tank adalah RUSAK.

[R2]: Jika Kondisi Fisik adalah HANCUR dan Jumlah Amunisi adalah SEDANG maka Status Kesehatan Tank adalah AGAK RUSAK.

[R3]: Jika Kondisi Fisik adalah HANCUR dan Jumlah Amunisi adalah BANYAK maka Status Kesehatan Tank adalah NORMAL.

[R4]: Jika Kondisi Fisik adalah STANDAR dan Jumlah Amunisi adalah SEDIKIT maka Status Kesehatan Tank adalah AGAK RUSAK.

[R5]: Jika Kondisi Fisik adalah STANDAR dan Jumlah Amunisi adalah SEDANG maka Status Kesehatan Tank adalah NORMAL.

[R6]: Jika Kondisi Fisik adalah STANDAR dan Jumlah Amunisi adalah BANYAK maka Status Kesehatan Tank adalah CUKUP BAIK.

[R7]: Jika Kondisi Fisik adalah UTUH dan Jumlah Amunisi adalah SEDIKIT maka Status Kesehatan Tank adalah NORMAL.

[R8]: Jika Kondisi Fisik adalah UTUH dan Jumlah Amunisi adalah SEDANG maka Status Kesehatan Tank adalah CUKUP BAIK.

[R9]: Jika Kondisi Fisik adalah UTUH dan Jumlah Amunisi adalah BANYAK maka Status Kesehatan Tank adalah BAIK.

Setelah aturan dibentuk, maka dilakukan aplikasi fungsi implikasi. Pada Metode Mamdani, fungsi implikasi yang digunakan adalah MIN, yang berarti tingkat keanggotaan yang didapat sebagai konsekuen dari proses ini adalah nilai minimum dari variabel Kondisi Fisik dan Jumlah Amunisi. Sehingga didapatkan daerah fuzzy pada variabel Status Kesehatan Tank untuk masing-masing aturan.

3.2.3 Komposisi Aturan

Pada metode Mamdani, komposisi antarfungsi implikasi menggunakan fungsi MAX yaitu dengan cara mengambil nilai maksimum dari *output* aturan kemudian menggabungkan daerah fuzzy dari masing-masing aturan dengan operator OR.

$$\mu_{sf}[x] = \max (\mu_{kf1}[x], \mu_{kf2}[x], \mu_{kf3}[x], \mu_{kf4}[x], \mu_{kf5}[x], \mu_{kf6}[x], \mu_{kf7}[x], \mu_{kf8}[x], \mu_{kf9}[x])$$

Keterangan:

$\mu_{sf}[x]$: nilai keanggotaan solusi fuzzy sampai aturan ke-*i*; $\mu_{kfi}[x]$

$\mu_{kf}[x]$: nilai keanggotaan konsekuen fuzzy setiap aturan ke-*i*; $i = 1, 2, \dots, 6$.

3.2.4 Penegasan (Defuzzyfikasi)

Input dari proses defuzzyfikasi adalah suatu himpunan fuzzy yang diperoleh dari komposisi aturan fuzzy, sedangkan *output* yang dihasilkan merupakan suatu bilangan tegas (*crisp*) pada domain himpunan fuzzy tersebut. Sehingga jika diberikan suatu himpunan fuzzy dalam range tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai *output*. Defuzzyfikasi yang digunakan dalam menentukan nilai SKT adalah dengan metode *centroid*. Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat (Z_0) daerah fuzzy. Secara umum dirumuskan:

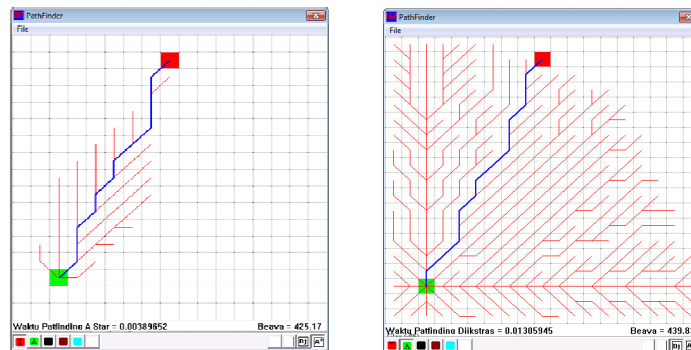
$$Z_0 = \frac{\int_a^b Z_b \mu_{(z)} dz}{\int_a^b \mu_{(z)} dz}$$

untuk domain kontinyu, dengan Z_0 adalah nilai hasil defuzzyfikasi dan $\mu_{(z)}$ adalah derajat keanggotaan titik tersebut, sedangkan Z adalah nilai domain ke-*i*.

4. HASIL DAN PEMBAHASAN

4.1. Pengujian Algoritma A* untuk Pencarian Jalur (Pathfinding)

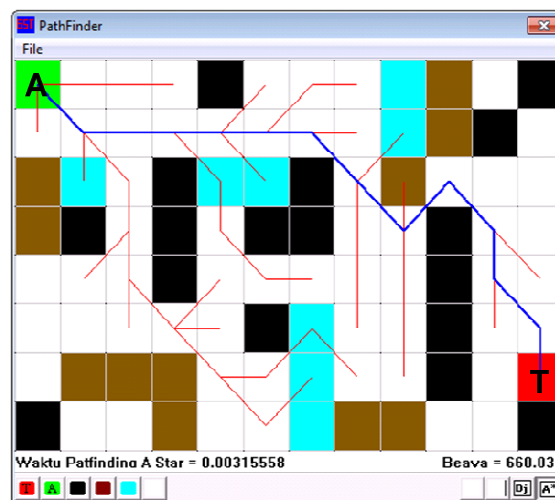
Pengujian yang akan digunakan untuk menemukan jalur yaitu dengan menggunakan penghalang dan tanpa penghalang. Pertama setelah pengguna menempatkan posisi awal dan posisi tujuan yang diinginkan, dan sudah menempatkan penghalang pada medan yang digunakan maka pencarian jalur dapat dilakukan dengan memilih submenu A* pada menu Algoritma. Hasil jalan yang ditemukan dengan waktu pencarian dapat dilihat pada gambar 6a.



Gambar 6 a. Jalur algoritma A* b. Jalur algoritma *Dijkstra*

Pengujian pencarian jalur dengan algoritma A* dapat dicapai dengan waktu 3.896 ms dan biayanya adalah 425.17. Sebagai perbandingan pencarian jalur dengan algoritma Dijkstra dicapai dengan waktu 13.059 ms dan biayanya adalah 439.83, dimana hasil tersebut dapat dilihat pada gambar 6b.

Gambar 7 adalah hasil pencarian jalur dengan algoritma A* dengan 3 macam penghalang, yaitu lumpur (kotak warna coklat), genangan air/sungai (kotak warna hijau kebiruan) dan bangunan/pohon/kendaraan lain (kotak warna hitam/abu-abu).



Gambar 7. Jalur A* dengan Berbagai Macam Kondisi Lingkungan

Perbandingan hasil pengujian performansi untuk algoritma A* dan Dijkstra dengan beberapa variasi titik awal dan titik tujuan pada peta dengan ordo (19, 19) dapat dilihat pada tabel 3.

Tabel 3. Perbandingan Hasil Pengujian antara A* dan Dijkstra

No.	Posisi Awal	Posisi Tujuan	A*		Dijkstra		Halangan
			Waktu	Beaya	Waktu	Beaya	
1	(0,0)	(18,18)	1.468	638.36	23.257	638.36	Tidak ada
2	(0,0)	(0,18)	0.937	431.05	14.909	14.909	Tidak ada
3	(0,0)	(18,0)	0.966	470.84	187.58	187.58	Tidak ada
4	(1,4)	(17,15)	6.121	520.90	20.647	20.647	Tidak ada
5	(1,4)	(17,15)	6.570	554.15	16.741	16.741	Ada
6	(0,0)	(18,18)	9.211	711.56	17.748	17.748	Ada
7	(0,0)	(0,18)	4.849	546.22	20.423	20.423	Ada

Dengan hasil uji yang telah dilakukan dapat dilihat hasil pencarian dengan menggunakan algoritma A* yaitu waktu pencarian, simpul yang diperiksa dan jarak rute yang dihasilkan. Pada algoritma A* jarak rute yang dihasilkan pendek, dengan waktu yang jauh lebih cepat dibandingkan Dijkstra.

4.2. Metode Logika Fuzzy dalam Menentukan Keputusan Peilaku Perang

Proses uji coba untuk metode Logika Fuzzy dilakukan dengan memasukkan sejumlah data ke dalam program yang telah dibuat. Hasil keputusan diambil dari perhitungan fuzzy kondisi fisik, jumlah amunisi, jumlah tank kawan dan jumlah tank lawan. Masing-masing kondisi merupakan hasil perhitungan fuzzy dari data yang bersifat dinamis (bisa berubah setiap saat).

4.2.1 Status Kesehatan Tank

Hasil dari algoritma pembelajaran menggunakan perangkat lunak untuk pengujian penggunaan metode logika fuzzy sudah dapat dihasilkan sebuah keputusan berdasarkan data *input* yang dimasukkan. Hasil program setelah *input* dimasukkan untuk Status Kesehatan Tank dapat dilihat pada *output* pada tabel 2 di bawah ini.

Tabel 2. Hasil pengujian Status Kesehatan Tank

Kondisi Fisik	Jumlah Amunisi	Status Kesehatani Tank
Utuh	Sedang	Baik
Utuh	Sedikit	Agak Rusak
Sedang	Sedang	Normal
Sedang	Banyak	Cukup Baik
Hancur	Cukup	Rusak
Hancur	Banyak	Agak Rusak
Utuh	Banyak	Baik

4.2.2 Perhitungan Jumlah Kawan dan Lawan

Pada perang modern seperti sekarang ini teknologi yang dipakai serba canggih, baik jumlah kawan maupun jumlah lawan harus bisa diketahui atau diprediksi dengan baik. Agar GST ini bisa mengikuti strategi perang modern, maka harus dapat memperhitungkan jumlah kawan dan lawan dalam menentukan setiap pergerakan dan strategi taktis dalam peperangan. Teknik dalam menentukan seberapa besar komposisi jumlah kawan dan lawan berpengaruh terhadap strategi perang adalah dengan cara membandingkan berapa jumlah kawan dibandingkan dengan jumlah lawan. KBP memiliki kemampuan jarak pandang tertentu dengan sudut 360 derajat. KBP akan mengenali adanya lawan atau kawan dalam jarak tersebut. KBP memerlukan kalkulasi untuk mengetahui keseimbangan kondisi antara kawan dan lawan. KBP akan mempertimbangkan SKT.

4.2.3 Penentuan Status Perilaku Perang

Dalam menentukan Status Perilaku Perang kapan KBP akan siap tempur, bertahan atau bahkan melarikan diri (menghindar) didapat dari pengolahan Status Kesehatan Tank dengan perbandingan jumlah kawan dan lawan. Bila perbandingan jumlah kawan dengan lawan lebih besar atau sama dengan 1.5, maka apapun Status Kesehatan Tanknya semua KBP Status Perilaku Perangnya adalah TEMPUR. Untuk lebih jelasnya kondisi Status Perilaku Perang KBP yang diolah berdasarkan Status Kesehatan Tank dan perbandingan jumlah kawan dan lawan dapat dilihat pada tabel 3.

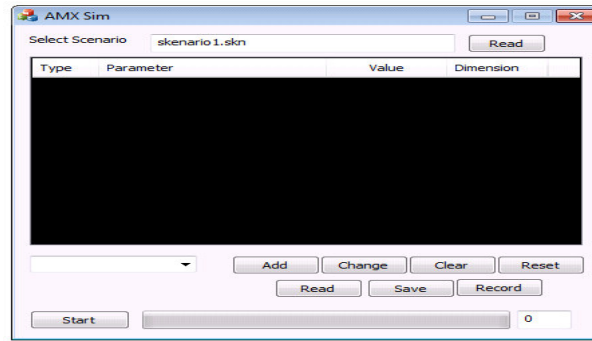
Tabel 3. Kondisi Status Perilaku Perang

Perbandingan Jumlah Kawan-Lawan	Status Kesehatan Tank	Status Perilaku Perang
$\geq 2:1$	Semua kondisi	Tempur
$1:1 - 1.5:1$	Cukup Baik, Baik	Tempur
	Normal , Agak Rusak	Bertahan
	Rusak	Lari
$\leq 1:1$	Cukup Baik, Baik	Bertahan
	Normal , Agak Rusak, Rusak	Lari

4.3. Implementasi pada *Game Simulator Tank*

Setelah dilakukan pengujian terhadap algoritma-algoritma yang digunakan, yaitu A* dan logika fuzzy, selanjutnya algoritma tersebut diimplementasikan pada aplikasi perangkat lunak *Game Simulator Tank* (GST).

Untuk menjalankan GST ini dilakukan dengan menjalankan file eksekusi yang bernama ATS_IGd.exe terlebih dahulu, lalu kemudian disusul dengan menjalankan file eksekusi Host_IGd.exe. Pada layar akan terlihat tampilan antarmuka sebagaimana terlihat pada gambar 8. Untuk memulai melakukan permainan *Game Simulator Tank* tekan tombol Start, namun sebelum itu pengguna harus memilih skenario terlebih dahulu tentang medan latihan yang dipakai, macam dan jumlah kendaraan yang akan digunakan, lokasi kendaraan, dan sebagainya. Pemilihan skenario dilakukan dengan memasukkan nama skenario yang akan dipakai setelah itu menekan tombol read. Namun begitu edit skenario masih bisa dilakukan sewaktu-waktu saat permainan masih berjalan.



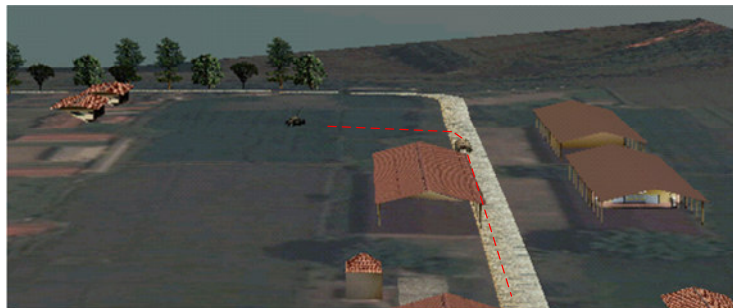
Gambar 8. Tampilan Awal *Game Simulator Tank*

Setelah beberapa waktu berjalan, perjalanan yang dilalui oleh tank dapat dilihat pada gambar 9 berikut ini.



Gambar 9. Jalur yang Dilalui Tank dengan Algoritms A*

Sebagai perbandingan bila tank tidak menggunakan algoritma A* untuk pencarian jalurnya, maka Tank berjalan sesuai dengan rute yang telah ditetapkan oleh program seperti terlihat pada gambar 10.



Gambar 10. Jalur yang Dilalui Tank Tanpa Algoritms A*

5. PENUTUP

5.1. Kesimpulan

Beberapa kesimpulan yang dapat diambil dari proses penelitian ini adalah sebagai berikut.

- a. Aplikasi *Game Simulator Tank* ini menggunakan situasi yang menyerupai keadaan sebenarnya, seperti medan latihan perang diambil dari lokasi Riil, yaitu di Padalarang dan jalur jalan dan kontur medan yang ada menyerupai Kondisi di lapangan.

- b. Pemilihan jalur sudah memperhitungkan kondisi jalan riil dengan memperhitungkan bila jalan tersebut berpenghalang tidak dapat dilewati atau dengan nilai heuristic tertentu sehingga kalau dilewati akan memperhambat jalan.
- c. Penerapan strategi perang semakin akurat, respon perilaku karakter bukan pemain akan selalu memperhitungkan kondisi kesehatan kendaraan waktu itu.
- d. Aplikasi sudah memperhitungkan jumlah kawan dan lawan.

5.2. Saran

Penerapan algoritma A* dan metode Logika Fuzzy sudah cukup baik dalam membuat sebuah media pembelajaran berbasis simulasi berupa *Game Simulator Tank* untuk memenuhi sasaran sesuai dengan peruntukannya, semua proses bisa dilakukan menyerupai dengan kondisi riil di lapangan. Namun hal yang sudah ada akan bisa menjadi lebih hidup dan lebih baik lagi bila juga menggunakan beberapa metode lain bersamaan dengan penerapan kedua metode yang menjadi obyek penelitian ini. Metode tersebut adalah penentuan gerak mengikuti pergerakan lawan secara lebih baik yaitu menggunakan *Finite State Machine* dan metode *Reinforcement Learning*. Penerapan kedua Metode tambahan tersebut pada *Game Simulator Tank* akan menjadikan kegunaan media pembelajaran tersebut menjadi lebih sempurna.

PERNYATAAN ORIGINALITAS

“Saya menyatakan dan bertanggung jawab dengan sebenarnya bahwa Artikel ini adalah hasil karya saya sendiri kecuali cuplikan dan ringkasan yang masing-masing telah saya jelaskan sumbernya” [Wali Muhammad – P31.2011.01016]

DAFTAR PUSTAKA

- [1] Xiao Cui and Hao Shi Jurnal : *A*-based Pathfinding in Modern Computer Games*. School of Engineering and Science, Victoria University, Melbourne, Australia, Januari 2011.
- [2] Yifan Li, et al. Jurnal : *Fuzzy logic in Agent-Based Game Design*. Department of Electrical and Computer Engineering University of Alberta Edmonton, AB T6G 2V4, Canada, 2004.
- [3] Millington and John Funge - 2009: *Artificial Intelligence for Games Second Edition*. Morgan Kaufmann Publishers is an imprint of Elsevier. 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA, 2009.
- [4] T. Sutojo, S.Si.,M.Kom., Edy Mulyanto, S.Si.,M.Kom., Dr. Vincent Suhartono – 2011: *Kecerdasan Buatan*. Andi Publisher, Jogjakarta.
- [5] Yunifa Miftachul Arif, et al. Jurnal : *Strategi Menyerang pada Game FPS Menggunakan Hierarchy Finite State Machine dan Logika Fuzzy*. Pasca Sarjana Teknik Elektro ITS, Surabaya 60111
- [6] Bambang Sridadi, 2009. *Pemodelan dan Simulasi Sistem : Teori, Aplikasi dan Contoh Program dalam Bahasa C*. Informatika, Bandung,
- [7] Yung-Ping Fang, et al. Jurnal: *Applying Reinforcement Learning for the AI in aTank-Battle Game*. Department of Information Management National University of Kaohsiung, No. 700, Kaohsiung University Road, Kaohsiung, 811 Taiwan, Desember 2010.
- [8] Pedro F. Felzenszwalb et al. Jurnal : *The Generalized A* Architecture*. Department of Computer Science University of Chicago Chicago, IL 60637, Juni 2007.
- [9] <http://www.delta3d.org>. diakses 7 Juni 2013.
- [10] Krzysztof R. Apt Lectures In Game Theory For Computer Scientists. Centrum Wiskunde En Informatica (Cwi), Amsterdam, 2011.
- [11] Lester, Patrick. (2003), *A* Pathfinding for Beginners*.
- [12] Xiaowei, S. (2006). *An intelligent recommendation system based on fuzzy logic*. Informatics in Control, Automation and Robotics I , 105-109.

- [13] Kusumadewi, S. & Purnomo, H., 2004, *Aplikasi Logika Fuzzy untuk Pendukung Keputusan*, Graha Ilmu. Yogyakarta.
- [14] Jin Hyuk Hong dan Sung-Bae Cho. *Evolving Reactive NPCs for the Real-Time Simulation Game*. CIG, 2005.
- [15] Wijaya, Surya Adi. 2009. *Fuzzy State Machine Untuk Menghasilkan Variasi Respon Npc (Non-Playable Character) Pada Game*. Tesis. Surabaya: Institut Teknologi Surabaya.
- [16] Iryanto. 2003. *Pengantar Teori dan Aplikasi Graph*. Medan: USU Press.
- [17] Riftadi, Mohammad. (2007), *Variasi Penggunaan Fungsi Heuristik dalam Pengaplikasian Algoritma A**, Makalah IF2251, Teknik Informatika ITB, Bandung.
- [18] Noor, T. Ricky . (2009), *Efisiensi Waktu Algoritma A* untuk Penentuan Jalur Optimal Pada Peta Geografis*, Program Studi Teknik Elektro, Universitas Pancasila.
- [19] Russell, S. J. Norvig, P. (2003), *Artificial Intelligence: A Modern Approach*.
- [20] Adipranata, Rudy, Andreas Handojo & Happy Setiawan (2007). *Aplikasi Pencari Rute Optimum Pada Peta*. Universitas Kristen Petra, Surabaya