

Penerapan Algoritma A Star dalam Menentukan Jalur Penyerangan pada Permainan Counter-Strike Global Offensive

Vigor Akbar / 13515031¹

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
¹13515031@std.stei.itb.ac.id

Abstrak—Algoritma A Star merupakan salah satu algoritma yang digunakan untuk menentukan pencarian jalur di dalam ruang solusi menggunakan fungsi evaluasi. Algoritma A Star ini dapat diterapkan pada permainan Counter-Strike Global Offensive untuk menentukan jalur penyerangan paling optimal yang akan digunakan.

Keywords—Counter-Strike, A Star, Ruang Solusi, Fungsi Evaluasi, Rute Terpendek.

I. PENDAHULUAN

Counter-Strike: Global Offensive adalah permainan yang bergenre *multiplayer first-person shooter*, yaitu sebuah permainan tembak-menembak dimana pemain memainkannya dalam sudut pandang orang pertama dan dimainkan secara online bersama-sama dengan pemain lain dari berbagai negara. Counter-Strike: Global Offensive dikembangkan oleh perusahaan pengembang game terkemuka yaitu Hidden Path Entertainment dan Valve Corporation. Permainan ini dirilis untuk platform Microsoft Windows, OS X, Xbox 360, dan PlayStation 3 pada tahun 2012 dan dirilis untuk platform Linux pada tahun 2014, namun permainan ini kebanyakan dimainkan pada komputer PC atau Laptop karena membutuhkan keyboard dan mouse untuk mendapatkan akurasi tembakan yang tinggi serta pergerakan yang cepat.

Permainan Counter-Strike: Global Offensive adalah permainan yang dimainkan oleh dua tim, yaitu tim *terrorist* dan tim *counter-terrorist*. Masing-masing tim terdiri dari lima pemain. Terdapat berbagai macam mode permainan yang dapat dipilih untuk dimainkan dalam permainan ini, diantaranya adalah mode *Bomb Mission*, mode *Death-Match*, dan mode *Hostage Rescue*. Mode *Bomb Mission* adalah mode dimana tim *terrorist* bertujuan untuk menanam bom pada *bomb site* yang terdapat pada peta. Sedangkan tim *counter-terrorist* bertujuan untuk menggagalkan tim lawannya untuk meledakan bom pada *bomb site*. Salah satu tim akan memenangkan suatu babak jika semua anggota tim lawannya sudah mati. Mode *Death-Match* adalah mode dimana masing-masing player berlomba-lomba untuk mendapatkan *kills* paling banyak. Pemain yang dapat membunuh lawannya dengan jumlah terbanyak adalah pemenangnya. Mode *Hostage Rescue* adalah mode dimana tim

counter-terrorist bertujuan membebaskan sandera yang ada di suatu tempat tertentu dengan cara membawanya ke *Rescue Zone*. Sedangkan tim *terrorist* bertujuan untuk menggagalkan rencana tim lawannya. Salah satu tim akan memenangkan suatu babak jika semua anggota tim lawannya sudah mati.

Pada makalah ini, akan dibahas mengenai jalur penyerangan yang dalam hal ini adalah bagaimana tim *terrorist* menentukan jalur penyerangan dalam mode *Bomb Mission* pada map *de_dust2*. Pemilihan jalur penyerangan yang tepat akan meningkatkan kemungkinan tim *terrorist* dapat memenangkan permainan.



Gambar 1. Tampilan permainan Counter Strike: Global Offensive mode *Death-Match*.

Penyerangan dapat dilakukan ke bom site A atau bom site B. Penyerangan yang dilakukan oleh tim *terrorist* dibatasi oleh waktu sehingga jarak jalur yang diambil juga harus diperhitungkan. Selain jarak juga terdapat faktor-faktor lain yang dapat menentukan berhasil atau tidaknya penyerangan yang dilakukan seperti jumlah uang yang dimiliki oleh anggota-anggota tim, jumlah musuh pada posisi tertentu yang sudah diketahui, *grenade* yang dimiliki dan lain-lain.

Algoritma A* (A Star) dapat digunakan untuk menentukan penentuan jalur penyerangan yang akan diambil.

Gambar 2. Jenis-Jenis Graf

(Sumber : http://mathworld.wolfram.com/images/eps-gif/GraphsSimple_800.gif)

II. LANDASAN TEORI

A. Graf

1. Definisi Graf

Graf G didefinisikan sebagai pasangan himpunan $G = (V, E)$, dalam hal ini:

$$V = \{v_1, v_2, v_3, \dots, v_n\}$$

$$E = \{e_1, e_2, e_3, \dots, e_n\}$$

V merupakan himpunan tidak kosong simpul-simpul, sedangkan E adalah himpunan sisi yang menghubungkan sepasang simpul.

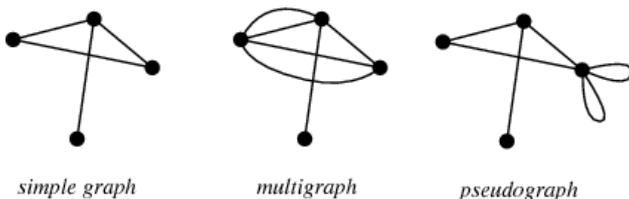
E boleh saja kosong sedangkan V tidak boleh kosong. Hal ini dapat dihubungkan dengan masalah transportasi di perkotaan, tidak mungkin ada jalan dibuat tanpa memiliki tempat yang ingin dihubungkan. Tetapi dapat saja suatu tempat tidak memiliki akses ke tempat lainnya.

Jenis-jenis graf dapat dikelompokkan berdasarkan ada atau tidaknya gelang atau sisi ganda pada suatu graf, dan berdasarkan orientasi arah pada sisi yang dimiliki oleh suatu graf.

2. Jenis-Jenis Graf

Berdasarkan ada atau tidaknya gelang atau sisi pada suatu graf, graf dapat digolongkan menjadi graf sederhana (*simple graph*) dan graf tidak sederhana (*unsimple graph*). Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi ganda, sedangkan graf tidak sederhana mengandung gelang atau sisi ganda. Graf tidak sederhana terbagi lagi menjadi graf ganda (*multigraph*) dan graf semu (*pseudograph*). Graf ganda dapat mengandung sisi ganda, sedangkan graf semu dapat memiliki gelang maupun sisi ganda.

Berdasarkan orientasi arah yang dimiliki oleh sisi-sisi pada graf, graf dapat digolongkan menjadi graf tak-berarah dan graf berarah. Graf berarah memiliki sisi yang berbentuk panah. Simpul pada graf berarah terbagi dua menjadi simpul asal (*initial vertex*) dan simpul terminal (*terminal vertex*). Graf berarah dapat memiliki sisi gelang, graf ganda berarah dapat memiliki sisi ganda maupun sisi gelang. Graf tidak berarah adalah graf biasa, yaitu graf yang sisi-sisinya hanya berupa garis yang tidak punya penunjuk.



B. Algoritma A* (A Star)

Algoritma A* adalah algoritma pengembangan dari algoritma pencarian BFS (Breadth First Search), yaitu mencari jalan dengan biaya terkecil ke tujuan tertentu dengan jumlah tujuan bisa lebih dari satu tujuan. Perbedaan yang menjadikan A* lebih baik daripada BFS adalah adanya fungsi heuristik yang digunakan dalam menentukan cost saat pembangkitan suatu simpul pada ruang solusi. Fungsi heuristik ini adalah sebuah perkiraan untuk suatu simpul, apakah simpul itu semakin baik untuk menuju suatu tujuan. Rumus lengkap dari algoritma ini adalah sebagai berikut.

$$F(x) = g(x) + h(x)$$

Jadi untuk setiap simpul yang dibangkitkan akan dihitung cost nya dengan cara menambahkan fungsi $g(x)$ dan $h(x)$, dimana $g(x)$ adalah jarak atau biaya dari suatu simpul ke simpul lain, biasanya berupa jarak atau waktu tempuh atau dapat juga berbentuk lainnya. Kemudian $h(x)$ adalah fungsi heuristik yang merupakan perkiraan seberapa baik kecenderungan suatu simpul untuk menuju ke simpul tujuan yang optimal. Sehingga dengan adanya fungsi evaluasi g -generate suatu simpul yang tidak menuju ke solusi permasalahan.

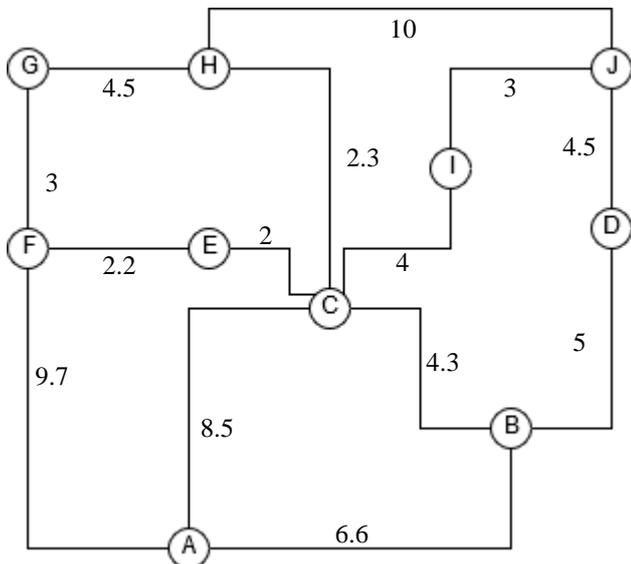
Setelah mendapatkan cost-cost hingga mendapatkan kandidat-kandidat solusi, jalur dengan cost paling optimal akan dipilih sebagai solusi dari suatu permasalahan yang ingin diselesaikan dengan algoritma A*.

C. Aim and Positioning

Aim dan *positioning* merupakan faktor penting dalam permainan Counter Strike: Global Offensive. Jika sudah menguasai teorinya dengan benar, kemungkinan pemain memenangkan pertandingan akan semakin besar.

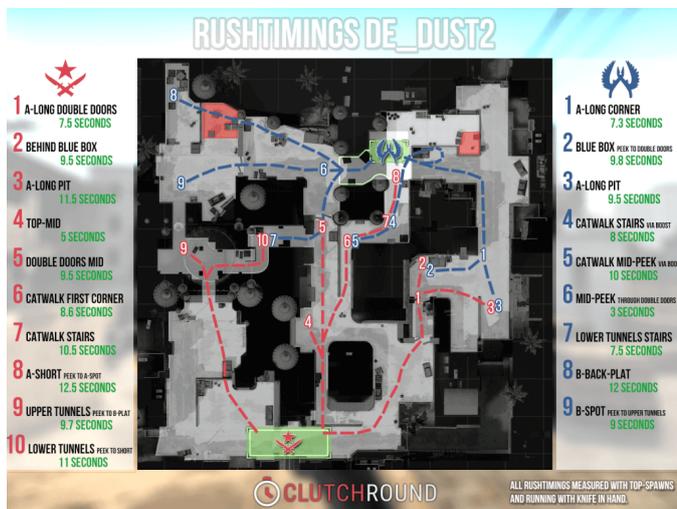
Aim adalah bagaimana kita membidik menggunakan *crosshair* yang ada pada layar computer saat kita bermain Counter-Strike: Global Offensive (CSGO). Kualitas *aim* seorang pemain CSGO adalah gabungan dari memori otot (*muscle memory*), reaksi (*reaction*), dan ketenangan (*calmness*).

Memori otot dapat terbentuk setelah bermain CSGO beberapa waktu dan akan semakin kuat seiring lamanya bermain. Reaksi dapat dibentuk dengan mempelajari aturan sederhana tentang penempatan posisi dan sudut gertak (*angle snapping*). Kemudian yang paling sulit untuk dikuasai adalah ketenangan saat melakukan aim ketika berhadapan dengan musuh. Kebanyakan pemain akan cenderung panik ketika bertemu dengan musuh sehingga cenderung akan melakukan aim yang salah sehingga peluru yang ditembakkan akan menyasar ke berbagai arah. Senjata pada permainan CSGO masing-masing memiliki *spray pattern*—persebaran peluru



Gambar 4. Graf dari map de_dust2

Waktu tempuh dari masing-masing simpul dengan cara berlari menggunakan pisau (kecepatan lari 250 unit per detik) didapat dari gambar berikut serta percobaan langsung penulis.

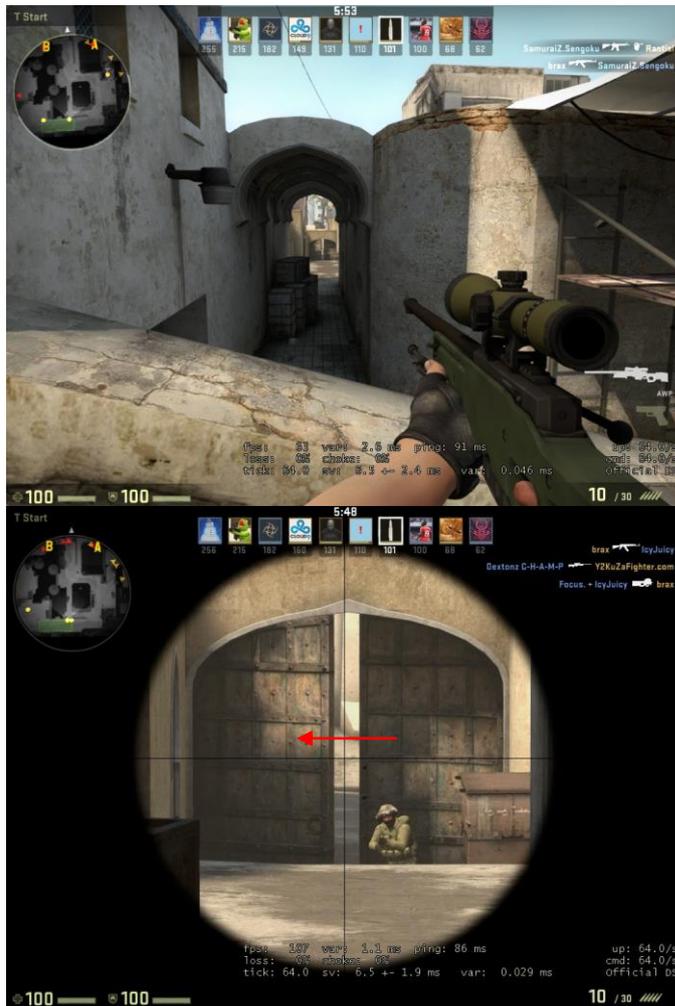


Gambar 5. Waktu tempuh pada map de_dust2

(Sumber: http://clutchround.com/csgo-map-mastery-de_dust2/)

Menggunakan *evaluation function* dari A^* , yaitu $f(n) = g(n) + h(n)$, $g(n)$ yang digunakan oleh penulis dalam menentukan jalur penyerangan adalah waktu tempuh, sedangkan $h(n)$ atau fungsi heuristik yang digunakan oleh penulis dalam makalah ini adalah gabungan perkiraan jumlah musuh yang dapat terlihat pada suatu spot tertentu serta apakah ada sniper disuatu spot tersebut. Satu musuh tanpa sniper diberi bobot 10, untuk musuh yang menggunakan sniper diberi bobot 25 jika kita tidak menggunakan sniper, diberi bobot 10 jika kita menggunakan sniper juga, jika tidak diketahui semua simpul yang dibangkitkan diberi bobot 5.

Salah satu cara mengetahui ada berapa musuh yang menjaga area bom A atau B adalah dengan melihat ada berapa musuh yang berlari dari *ct spawn* melewati *mid doors*. Jika ada dua orang yang terlihat melintas, maka area bom B ada dua orang yang menjaga dan seterusnya. Namun cara ini tidak dapat dilakukan jika pada penglihatan ke *mid door* dihalangi *smoke grenade*. Untuk lebih jelasnya dapat dilihat pada gambar berikut.



Gambar 6 dan 7. Cara memperkirakan posisi tim *counter-terrorist* dengan melihat musuh yang melintas.

Cara lainnya adalah dengan mendapatkan informasi dari teman satu tim yang melakukan *peeking* atau mengintip suatu tempat secara cepat untuk mengetahui posisi musuh serta senjata yang digunakannya.

IV. PENENTUAN JALUR PENYERANGAN BERDASARKAN ALGORITMA A^*

Berdasarkan gambar 4 dan perhitungan fungsi heuristik, akan ditentukan jalur penyerangan yang paling optimal dipilih untuk mencapai simpul G atau J, dengan posisi awal di simpul A.

Misalkan terjadi scenario sebagai berikut:

1. Ada dua orang musuh ke *bomb site* B (simpul G), satu di mid (simpul H), dua di *bomb site* A.
2. Player tidak menggunakan sniper.
3. Ada musuh yang menggunakan sniper di *bomb site* B (simpul G) dan di mid (simpul H).

Pilih A sebagai simpul-E.

Simpul-E	Simpul hidup	f(n)	f(n) terkecil
A	A-B	$6.6 + 2*(10) = 26.6$	A-B
	A-C	$8.5 + 25 = 33.5$	
	A-F	$9.7 + 10 + 25 = 44.7$	

Pilih A-B sebagai simpul-E

Simpul-E	Simpul hidup	F(n)	F(n) terkecil
A-B	A-C	$8.5 + 25 = 33.5$	A-B-D
	A-F	$9.7 + 10 + 25 = 44.7$	
	A-B-C	$10.9 + 25 = 35.9$	
	A-B-D	$11.6 + 20 = 31.6$	

Pilih A-B-D sebagai simpul-E

Simpul-E	Simpul hidup	F(n)	F(n) terkecil
A-B-D	A-C	$8.5 + 25 = 33.5$	A-C
	A-F	$9.7 + 10 + 25 = 44.7$	
	A-B-C	$10.9 + 25 = 35.9$	
	A-B-D-J	$16.1 + 20 = 36.1$ (kandidat solusi)	

Pilih A-C sebagai simpul-E

Simpul-E	Simpul hidup	F(n)	F(n) terkecil
A-C	A-F	$9.7 + 10 + 25 = 44.7$	A-C-E
	A-B-C	$10.9 + 25 = 35.9$	

	A-B-D-J	$16.1 + 20 = 36.1$ (kandidat solusi)	
	A-C-E	$10.5 + 5 = 15.5$	
	A-C-H	$10.8 + 25 = 35.8$	
	A-C-I	$12.5 + 20 = 32.5$	

Pilih A-C-E sebagai simpul-E

Simpul-E	Simpul hidup	F(n)	F(n) terkecil
A-C-E	A-F	$9.7 + 10 + 25 = 44.7$	A-C-I
	A-B-C	$10.9 + 25 = 35.9$	
	A-B-D-J	$16.1 + 20 = 36.1$ (kandidat solusi)	
	A-C-H	$10.8 + 25 = 35.8$	
	A-C-I	$12.5 + 20 = 32.5$	
	A-C-E-F	$12.7 + 10 + 25 = 47.7$	

Pilih A-C-I sebagai simpul-E

Simpul-E	Simpul hidup	F(n)	F(n) terkecil
A-C-I	A-F	$9.7 + 10 + 25 = 44.7$	
	A-B-C	$10.9 + 25 = 35.9$	
	A-B-D-J	$16.1 + 20 = 36.1$ (kandidat solusi)	
	A-C-H	$10.8 + 25 = 35.8$	
	A-C-E-F	$12.7 + 10 + 25 = 47.7$	

		solusi)	
--	--	---------	--

Karena tidak ada lagi simpul hidup yang memiliki $f(n)$ lebih kecil dari kandidat solusi A-C-I-J, maka jalur penyerangan optimal adalah melalui mid, lalu ke a-short, lalu ke bomb site A, yaitu melewati simpul A-C-I-J karena merupakan jalur terpendek dan relatif aman dibandingkan jika pergi ke bomb site B yang memiliki sniper.

V. KESIMPULAN

Algoritma A* (A Star) dapat digunakan untuk menjelaskan penentuan jalur penyerangan paling optimal dari tim *terrorist* pada permainan Counter-Strike: Global Offensive. Penentuan rumus heuristik dari algoritma A* sangat mempengaruhi penentuan jalur, sehingga semakin banyak faktor-faktor yang dimasukkan kedalam fungsi heuristik, akan semakin akurat perkiraan jalur optimal yang dapat dipilih dalam melakukan penyerangan. Penentuan jalur dengan jarak tempuh tersingkat juga menentukan apakah jalur tersebut optimal dalam penyerangan dikarenakan penyerangan tim *terrorist* dibatasi oleh waktu sehingga dengan memanfaatkan waktu semaksimal mungkin, penyerangan dapat dilancarkan dengan lebih optimal.

VI. UCAPAN TERIMA KASIH

Penulis mengucapkan syukur dan terima kasih kepada Allah SWT karena makalah ini telah terselesaikan. Penulis juga mengucapkan terimakasih kepada dosen pembimbing mata kuliah IF2211 Strategi Algoritma Dr. Ir. Rinaldi Munir, MT., Dr. Nur Ulfa Maulidevi, S.T., M.Sc., dan Dr. Masayu Leylia Khodra, S.T., M.T. atas bimbingannya. Serta terimakasih kepada pihak lain yang turut membantu dalam pembuatan makalah ini.

REFERENCES

- [1] Bone, 2014, "Advanced CSGO Theory", <http://steamcommunity.com/sharedfiles/filedetails/?id=313351628>, diakses tanggal 18 Mei 2017.
- [2] Pinsof, Allistair, 2012, "Review: Counter-Strike: Global Offensive", <https://www.destructoid.com/review-counter-strike-global-offensive-233724.phtml>, diakses tanggal 18 Mei 2017.
- [3] Froosh, 2013, "Map Call-Outs (Competitive Maps)", <https://steamcommunity.com/sharedfiles/filedetails/?id=157442340>, diakses tanggal 18 Mei 2017.
- [4] Geeksforgeeks.org, 2016, "A* Search Algrprithm", <http://www.geeksforgeeks.org/a-search-algorithm/>, diskases tanggal 19 Mei 2017.
- [5] Munir, Rinaldi, 2009, *Matematika Diskrit*, Bandung: Informatika.
- [6] Munir, Rinaldi, 2009, *Diktat Kuliah IF2211 Strategi Algoritma*, Bandung: Informatika.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 18 Mei 2017



Vigor Akbar
13515031