

Penerapan Algoritma A* untuk Optimasi Jalur Rotasi Jungler pada Game Mobile Legends: Bang Bang dengan Pemodelan Satu Map Penuh

Muh. Hartawan Haidir - 13524147

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10 Bandung

E-mail: haidirmuhhartawan@gmail.com , 13524147@std.stei.itb.ac.id

Abstract—Mobile legends: Bang Bang (MLBB) adalah game MOBA yang populer di kalangan anak-anak, remaja (pelajar/mahasiswa), hingga orang dewasa muda (millennial). Selain itu, game ini juga paling populer di Asia Tenggara. Peran jungler dalam game ini bertugas melakukan farming monster jungle serta rotasi ke lane lain untuk membantu rekan tim, termasuk melakukan invade ke jungle musuh dan memperebutkan objektif bersama seperti Turtle dan Lord. Makalah ini mengusulkan pemodelan peta mlbb secara penuh sebagai graf berbobot yang mencakup seluruh wilayah strategis untuk menentukan jalur rotasi jungler yang optimal. Graf terdiri dari 17 node yang merepresentasikan seluruh titik penting di map, dengan bobot edge berupa waktu tempuh dan faktor risiko invade. Algoritma A* dipilih karena mampu menemukan jalur optimal secara efisien melalui fungsi evaluasi $f(n) = g(n) + h(n)$ dengan heuristik jarak Euclidean yang bersifat admissible. Pemodelan dua sisi map membuka tiga skenario utama: farming jungle sendiri, invade jungle musuh, dan perebutan objektif bersama (Turtle/Lord). Hasil analisis menunjukkan A* mampu menangani ketiga skenario dengan kompleksitas waktu $O(E \log V)$ dan secara konsisten menghasilkan jalur yang lebih efisien dibandingkan BFS maupun DFS.

Keywords—A*; algoritma pencarian; graf berbobot; heuristik; invade; jungler; Mobile Legends; objektif bersama; rotasi.

I. PENDAHULUAN

Mobile Legends: Bang Bang (MLBB) adalah game bergenre Multiplayer Online Battle Arena (MOBA) yang dikembangkan oleh Moonton dan merupakan salah satu game mobile paling populer di Asia Tenggara, terutama Indonesia [1]. Dua tim yang beranggotakan lima pemain bertarung dengan tujuan menghancurkan markas lawan melalui tiga jalur utama (lane) yang dihubungkan oleh area hutan (jungle) di kedua sisi peta. Selain itu, setiap player mempunyai role masing-masing, yaitu jungler, gold lane, exp lane, mid lane, dan support sehingga tugas yang dimiliki oleh setiap player juga berbeda.

Peran jungler memiliki tanggung jawab unik: bergerak di area jungle untuk membunuh monster yang memberikan experience (exp) dan gold, kemudian melakukan rotasi untuk membantu rekan tim. Lebih dari itu, jungler juga dapat menyerang jungle milik lawan (invade) untuk mencuri sumber daya atau mengganggu ritme rotasi atau farming musuh, serta memperebutkan objektif bersama berupa Turtle dan Lord yang memberikan keuntungan massal bagi tim [2].

Keputusan jungler menyangkut tiga pilihan besar setiap saat: (1) melanjutkan farming jungler itu sendiri, (2) melakukan rotasi ke lane untuk membantu teman, (3) memperebutkan atau mencuri objektif di wilayah musuh. Setiap pilihan memberikan biaya (waktu dan risiko) serta manfaat yang berbeda-beda dan keputusan yang salah dapat membalikkan jalannya pertandingan [3].

Masalah ini secara alami dapat dimodelkan sebagai masalah pencarian jalur pada graf berbobot yang mencakup seluruh peta-bukan hanya satu sisi jungle. Algoritma A* (A-star), yang diperkenalkan oleh Hart, Nilsson, dan Raphael [4], merupakan algoritma pencarian jalur yang menggabungkan keoptimalan Uniform Cost Search (UCS) dengan efisiensi Greedy Best First Search melalui penggunaan heuristik. Makalah ini bertujuan: (1) memodelkan seluruh peta MLBB sebagai graf berbobot dengan 17 node, (2) mendefinisikan bobot edge yang mencakup faktor risiko invade, (3) menerapkan A* pada tiga skenario utama, pergerakan jungler, serta (4) menganalisis keunggulan A* untuk masalah ini dibandingkan algoritma lain.

II. DASAR TEORI

A. Graf Berbobot dan Pencarian Jalur

Graf berbobot $G = (V, E, w)$ terdiri dari himpunan simpul V , himpunan sisi E , dan fungsi bobot $w: E \rightarrow R^+$ yang memetakan setiap sisi ke nilai biaya positif. Masalah pencarian jalur optimal bertujuan menemukan jalur dari simpul asal s ke simpul tujuan t dengan total bobot minimum [5].

Pada lingkungan game MOBA, peta permainan secara alami membentuk graf setiap titik penting (monster, objektif, posisi lane) adalah node, jalur yang dapat dilalui karakter adalah edge, dan biaya perjalanan adalah bobotnya. Kompleksitas, keputusan jungler – yang mencakup dua sisi peta dan objektif bersama – menuntut pemodelan graf yang menyeluruh [6].

B. Algoritma A*

A* (A-star) mempertahankan sebuah open list (priority queue) berisi node kandidat diurutkan berdasarkan fungsi evaluasi:

$$f(n) = g(n) + h(n)$$

Dengan $g(n)$ adalah biaya aktual dari node awal ke node n , dan $h(n)$ adalah estimasi heuristik biaya dari n ke tujuan. A^* mengekskansi node dengan nilai $f(n)$ terkecil, menjamin ditemukannya jalur optimal jika $h(n)$ bersifat admissible – tidak pernah melebihi-lebihkan biaya sebenarnya [4].

Pseudocode A^* secara ringkas adalah sebagai berikut: inisialisasi open list dengan node awal, selama open list tidak kosong, ambil node n dengan $f(n)$ terkecil. Jika n adalah tujuan, rekonstruksi jalur, jika tidak, maka ekspansi semua tetangga n dan perbarui nilai g dan f mereka jika jalur yang ditemukan lebih baik.

C. Fungsi Heuristik

Heuristik yang digunakan adalah jarak Euclidean antara posisi koordinat node n dan node tujuan t :

$$h(n) = \sqrt{(x_t - x_n)^2 + (y_t - y_n)^2}$$

Heuristik ini admissible karena jarak lurus selalu lebih kecil atau sama dengan jarak tempuh nyata. Heuristik juga konsisten (monoton) sehingga A^* tidak perlu mengunjungi node yang sama dua kali, memastikan efisiensi pencarian [4][8].

III. PEMODELAN MASALAH

A. Peta MLBB sebagai Graf Berbobot Penuh

Berbeda dengan model sederhana yang hanya mencakup satu sisi jungle, makalah ini memodelkan seluruh peta MLBB sebagai satu graf terpadu $G = (V, E, w)$ dengan $|V| = 17$ node yang mencakup semua wilayah strategis. Hal ini diperlukan karena pergerakan jungler sesungguhnya tidak dibatasi oleh garis imajiner–jungler dapat dan sering harus memasuki wilayah musuh



Gambar I. Map Mobile Legends: Bang Bang [9]

Tabel I menyajikan seluruh node beserta nilai sumber daya yang diperoleh Tanda (*) pada Turtle dan Lord menunjukkan bahwa gold didistribusikan ke seluruh anggota tim, bukan hanya jungler.

TABLE I. DAFTAR NODE SELURUH MAP MLBB

ID	Nama Node	Wilayah	EXP	Gold
v0	Base (tim)	Tim sendiri	-	-
v1	Blue Buff	Tim sendiri	100	50

v2	Red Buff	Tim sendiri	80	50
v3	Lithowanderer	Tim sendiri	60	40
v4	Crab (bawah)	Tim sendiri	30	20
v5	Gold Lane	Tim sendiri	-	-
v6	Exp Lane	Tim sendiri	-	-
v7	Mid Lane	Tengah	-	-
v8	Turtle	Tengah (bersama)	0	200*
v9	Lord	Tengah (bersama)	0	300*
v10	Blue Buff (M)	Musuh	100	50
v11	Red Buff (M)	Musuh	80	50
v12	Litho (M)	Musuh	60	40
v13	Crab (M)	Musuh	30	20
v14	Gold Lane (M)	Musuh	-	-
v15	Exp Lane (M)	Musuh	-	-
v16	Base (M)	Musuh	-	-

Graf yang digunakan pada penelitian ini merupakan graf berbobot tak berarah yang merepresentasikan seluruh area penting pada peta Mobile Legends: Bang Bang. Setiap simpul (node) merepresentasikan lokasi strategis yang dapat dikunjungi oleh jungler, sedangkan sisi (edge) merepresentasikan jalur perpindahan yang memungkinkan antar lokasi tersebut. Node pada graf mencakup area jungle tim sendiri, area netral, objektif bersama, serta area jungle musuh. Dengan pemodelan ini, seluruh keputusan jungler dapat direpresentasikan dalam satu graf terpadu tanpa perlu memisahkan area tim sendiri dan area musuh.

TABLE II. KOORDINAT NODE PADA PETA

NODE	NAMA	X	Y
v0	Base (tim)	10	10
v1	Blue Buff	20	20
v2	Red Buff	35	25
v3	Lithowanderer	50	40
v4	Crab (bawah)	60	20

v5	Gold Lane	80	30
v6	Exp Lane	30	80
v7	Mid Lane	50	50
v8	Turtle	60	60
v9	Lord	70	80
v10	Blue Buff (M)	80	80
v11	Red Buff (M)	75	65
v12	Litho (M)	65	55
v13	Crab (M)	80	45
v14	Gold Lane (M)	90	70
v15	Exp Lane (M)	70	95
v16	Base (M)	95	95

Red Buff	Turtle	12	Tinggi	Objektif
Turtle	Lord	15	Tinggi	Objektif
Blue Buff	Blue Buff(M)	14	Sangat Tinggi	Invade
Red Buff	Red Buff(M)	12	Sangat Tinggi	Invade
Mid Lane	Litho(M)	11	Tinggi	Invade

B. Bobot Edge dan Faktor Risiko Invade

Kebaruan model ini dibandingkan versi sebelumnya adalah dimasukkannya faktor risiko (r) pada bobot edge yang mengarah ke wilayah musuh atau objektif bersama. Bobot efektif didefinisikan sebagai :

$$w_eff(u,v) = w(u,v) + r(u,v) * k$$

Di mana $w(u, y)$ adalah waktu tempuh dalam detik, $r(u,v) \in \{0, 1, 2\}$ adalah tingkat risiko (rendah tinggi sangat tinggi), dan k adalah konstanta penalti yang dapat disetel (default $k = 5$). Dengan formulasi ini, A* secara otomatis memprioritaskan jalur aman kecuali jika jalur berisiko memberikan sumber daya yang jauh lebih besar.

TABLE III. BOBOT EDGE UTAMA PADA GRAF MLBB

Dari	Ke	W(e) detik	Risiko	Tipe
Base	Blue Buff	8	Rendah	Farming
Blue Buff	Red Buff	5	Rendah	Farming
Red Buff	Lithowan d.	6	Rendah	Farming
Lithowan d.	Gold Lane	7	Rendah	Rotasi
Blue Buff	Exp Lane	10	Rendah	Rotasi

C. Fungsi Biaya dan Heuristik

Tujuan jungler adalah memaksimalkan nilai sumber daya total dalam interval waktu tertentu. Fungsi biaya dari u ke v didefinisikan ulang sebagai :

$$c(u,v) = w_eff(u,v) - R(v)$$

Di mana $R(v)$ adalah nilai sumber daya ternormalisasi dari node tujuan v . Nilai $R(v)$ dihitung sebagai $(exp + gold) / 10$ agar satuan sebanding dengan bobot waktu dalam detik. Dengan formulasi ini, node bernilai tinggi (seperti Lord dengan $R = 60$) secara efektif “menarik” jalur A* ke arahnya jika jarak dan risikonya sebanding

Koordinat setiap node didefinisikan pada sistem koordinat peta MLBB yang dinormalisasi (0, 0) di sudut kiri-bawah hingga (100, 100) di sudut kanan-atas, Heuristik Euclidean dihitung berdasarkan koordinat ini dan tetap admissible karena peta dua dimensi

Heuristik yang digunakan pada penelitian ini adalah jarak Euclidean antara node saat ini dan node tujuan. Heuristik tersebut dipilih karena mampu memberikan estimasi jarak sederhana namun efektif pada lingkungan dua dimensi.

Secara matematis :

$$h(n) = \sqrt{(x_t - x_n)^2 + (y_t - y_n)^2}$$

Heuristik Euclidean bersifat admissible karena nilai yang dihasilkan tidak pernah melebihi biaya sebenarnya untuk mencapai tujuan. Hal ini disebabkan jalur aktual yang ditempuh hero selalu memiliki panjang yang sama atau lebih besar dibandingkan jarak garis lurus antara dua titik.

Dengan demikian berlaku :

$$h(n) \leq h^*(n)$$

Untuk setiap node n , dengan $h^*(n)$ merupakan biaya sebenarnya menuju tujuan. Karena memenuhi sifat admissible dan konsisten (*consistent heuristic*), algoritma A* tetap menjamin ditemukannya solusi optimal

D. Contoh Perhitungan Fungsi Evaluasi

Sebagai ilustrasi, misalkan jungler baru saja meninggalkan Base (v0) dan berada pada node Blue Buff (v1) dengan tujuan Gold Lane (v5).

Diketahui :

$$g(v1) = 8 \text{ (bobot edge Base} \rightarrow \text{Blue Buff)}$$

Koordinat :

$$v_1 = (20, 20)$$

$$v_5 = (80, 30)$$

Maka nilai Heuristik :

$$h(v_1) = \sqrt{(80-20)^2 + (30-20)^2}$$

$$= \sqrt{3600+100}$$

$$= \sqrt{3700}$$

$$\approx 60.83$$

Sehingga fungsi evaluasi A* adalah :

$$f(v_1) = g(v_1) + h(v_1)$$

$$f(v_1) = 8 + 60.83$$

$$f(v_1) = 68.83$$

Sebagai perbandingan, A* juga mempertimbangkan Exp Lane (v6) dengan koordinat (30, 80) yang dapat dicapai langsung dari Base dengan $g(v6) = 18$:

$$h(v_6) = \sqrt{(80 - 30)^2 + (30 - 80)^2}$$

$$\approx 70.71$$

$$f(v_6) = g(v_6) + h(v_6)$$

$$f(v_6) = 88.71$$

Karena $f(v1) = 68.83 < f(v6) = 88.71$, A* memilih mengekskansi Blue Buff (v1) terlebih dahulu. Ini menunjukkan bahwa heuristik Euclidean berhasil mengarahkan pencarian ke node yang secara geometris dan biaya aktual lebih menguntungkan menuju Gold Lane. Node dengan nilai $f(n)$ terkecil akan selalu dipilih untuk diekspansi pada setiap iterasi berikutnya.

IV. PENERAPAN ALGORITMA A* PADA TIGA SKENARIO JUNGLER

A. Gambaran Umum Tiga Skenario

Dengan model graf satu map penuh, terdapat tiga skenario utama pergerakan jungler yang semuanya dapat ditangani oleh satu instansi A* dengan node tujuan yang berbeda:

- 1) *Skenario Farming dan Rotasi Lane*: jungler memulai dari Base, melakukan farming camp jungle sendiri secara efisien, lalu rotasi ke lane yang membutuhkan bantuan. Node tujuan adalah posisi lane (Gold, Exp, atau Mid)
- 2) *Skenario Perebutan Objektif Bersama*: Jungler menuju Turtle atau Lord untuk diperebutkan. Node tujuan adalah v8 (Turtle) atau v9 (Lord). Bobot edge ke objektif ini lebih tinggi karena risiko bertemu tim musuh.
- 3) *Skenario Invade*: Jungler memasuki jungle musuh untuk mencari buff atau camp. Node tujuan adalah v10 - v13 (camp jungle musuh). Bobot edge invade mengandung penalti risiko tinggi $k = 2$.

B. Skenario 1 : Farming dan Rotasi ke Gold Lane

Skenario: Jungler memulai pertandingan dari iBase (v0) dengan tujuan mencapai Gold Lane (v5) sambil mengambil semaksimal mungkin sumber daya jungle di sepanjang jalur. Tabel III menunjukkan trace eksekusi A*.

TABLE IV. TRACE A* – SKENARIO 1: BASE KE GOLD LANE

Iter	Node Ekspansi	g(n)	h(n)	f(n)	Open List (ringkas)
1	Base (v0)	0	72.8	72.8	{Blue Buff: $f=8+60.8=68.8$, Exp Lane: $f=18+70.7=88.7$ }
2	Blue Buff (v1)	8	60.8	68.8	{Red Buff: $f=13+45.3=58.3$, Exp Lane: $f=18+70.7=88.7$ }
3	Red Buff (v2)	13	45.3	58.3	{Litho: $f=19+31.6=50.6$, Turtle: $f=25+42=67$ }
4	Lithowand. (v3)	19	31.6	50.6	{Gold Lane: $f=26+0=26$ }
5	Gold Lane (v5) ✓	26	0	26	SELESAI — jalur optimal ditemukan

Jalur optimal yang ditemukan Base $\rightarrow$ Blue Buff $\rightarrow$ Red Buff $\rightarrow$ Lithowanderer $\rightarrow$ Gold Lane, dengan total waktu $g = 26$ detik dan total sumber daya yang dikumpulkan : 240 exp +

140 gold. A* tidak mengeksplorasi turtle ($f = 40$) maupun Exp lane ($f = 88.7$) karena nilai f -nya lebih besar.

C. Skenario 2: Perebutan Objektif Turtle

Skenario: Pada menit ke-4, Turtle (v8) hampir bisa dibunuh Jungler saat ini di Red Buff (v2) dan perlu memutuskan apakah langsung ke Turtle atau kembali ke Base dulu A* dijalankan dengan node asal v2 dan tujuan v8.

TABLE V. SKENARO 2: RED BUFF KE TURTLE

Iter	Node Ekspansi	g(n)	h(n)	f(n)	Open List (ringkas)
1	Red Buff (v2)	13	42.1	55.1	Red Buff lebih dekat ke Turtle daripada Litho
2	Turtle (v8) ✓	25	0	25	SELESAI — via Base→Blue→Red →Turtle (25 dtk)

Jalur Optimal: Red Buff → Turtle langsung (12 detik, $w_{eff} = 12 + 1*5 = 17$ dengan risiko tinggi). A* memilih jalur ini karena lebih cepat daripada kembali ke Blue Buff dahulu (total $5+12=17$ detik, f lebih besar). Ini menunjukkan A* mampu menangani konteks mid-game di mana jungler tidak selalu memulai dari Base.

D. Skenario 3: Invade Jungle Musuh

Skenario: Tim lawan memiliki jungler yang lebih lemah di early game. Jungler kita ingin steal Blue Buff musuh (v10). A* dijalankan dari Base (v0) ke v10 dengan bobot invade menggunakan penalti $k=5$, $r=2$, sehingga $w_{eff}(v1 \rightarrow v10) = 14 + 2*5 = 24$.

TABLE VI. SKENARIO 3: INVADE BLUE BUFF MUSUH

Iter	Node Ekspansi	g(n)	h(n)	f(n)	Open List (ringkas)
1	Base (v0)	0	98.9	98.9	Tujuan = Blue Buff Musuh (v10)
2	Blue Buff (v1)	8	84.9	92.9	Jalur sendiri lebih murah dari invade langsung
3	Red Buff (v2)	13	70.7	83.7	Mendekati perbatasan map
4	Blue Buff Musuh ✓	27	0	27	SELESAI — $w_{eff}=14+5*2=24$, via Red→invade→Blue(M)

A* menemukan jalur invade: Base → Blue Buff → Red Buff → Blue Buff Musuh (total $g = 27$ detik termasuk penalti risiko). Menariknya, A* memilih melewati camp sendiri (Blue

Buff dan Red Buff) sebelum invade karena ini memberikan sumber daya tambahan dengan risiko nol, sedangkan invade langsung dari Base ($g = 0 + 24 = 24$ lebih kecil) tidak melalui node perantara yang memberi nilai. Hal ini menunjukkan fleksibilitas model fungsi biaya $c(u,v)$ yang telah memasukkan nilai sumber daya.

V. ANALISIS

A. Kompleksitas Waktu dan Ruang

Dengan $|V| = 17$ node dan $|E| \approx 28$ edge menggunakan binary heap sebagai priority queue, kompleksitas waktu A* adalah $O(E \log V) = O(28 \log 17) \approx O(115)$ operasi per pencarian. Ini sangat efisien untuk lingkungan real-time game yang membutuhkan respons cepat (umumnya < 16 ms per frame).

Kompleksitas ruang adalah $O(V)$ untuk menyimpan open list dan closed list. Dalam implementasi nyata pada game, overhead memori ini sangat kecil—graph 17 node hanya membutuhkan beberapa kilobyte.

B. Perbandingan dengan Algoritma Lain

Tabel VII di bawah merangkum perbandingan A* terhadap algoritma pencarian lain untuk masalah rotasi jungler pada graf satu map penuh :

TABLE VII. PERBANDINGAN ALGORITMA PENCARIAN

ALGO RITMA	OPTIM AL?	BOBOT EDGE?	HEURISTIK?	KOMPLEKSITAS	COCOK UNTUK MLBB?
BFS	YA (TANPA BOBOT)	TIDAK	TIDAK	$O(V+E)$	TIDAK — ABAIKAN BOBOT WAKTU
DFS	TIDAK	TIDAK	TIDAK	$O(V+E)$	TIDAK — TIDAK OPTIMAL
UCS/ DIJKSTRA	YA	YA	TIDAK	$O(E \log V)$	BISA, TAPI KURANG EFISIEN
GREEDY BFS	TIDAK	TIDAK	YA	$O(E \log V)$	TIDAK — TIDAK OPTIMAL
A*	YA	YA	YA	$O(E \log V)$	YA — OPTIMAL DAN EFISIEN

A* unggul karena merupakan satu-satunya algoritma yang sekaligus: (1) menjamin optimalitas pada graf berbobot, (2) menggunakan heuristik untuk mengarahkan pencarian sehingga lebih sedikit node yang dieksplorasi dibandingkan UCS, dan (3) secara alami menangani bobot edge heterogen termasuk penalti risiko invade.

C. Keunggulan Model Graf Satu Map Penuh

Model satu map penuh yang diusulkan memiliki beberapa keunggulan dibandingkan model parsial yang hanya mencakup satu sisi jungle. Pertama, model ini lebih realistis

karena mencerminkan kebebasan gerak jungler sesungguhnya di dalam game. Kedua, objektif bersama (Turtle dan Lord) dapat dimodelkan secara alami sebagai node dengan bobot edge tinggi dari kedua sisi, tanpa perlu pemodelan terpisah. Ketiga, keputusan invade dapat dievaluasi secara kuantitatif melalui parameter penalti risiko k yang dapat dikalibrasi sesuai patch game terkini.

Keterbatasan model ini adalah asumsi bahwa bobot edge bersifat statik, padahal dalam game nyata posisi musuh terus berubah sehingga risiko invade bervariasi secara dinamis. Pengembangan ke depan dapat menggunakan pendekatan Dynamic A* (D*) atau replanning A* untuk menangani ketidakpastian ini [7].

D. Pengaruh Faktor Risiko Invade

Besarnya nilai penalti risiko k sangat memengaruhi keputusan yang diambil oleh algoritma A*. Semakin besar nilai k , semakin mahal biaya efektif untuk memasuki wilayah musuh.

TABLE VIII. PENGARUH VARIASI NILAI PENALTI

Nilai k	Jalur yang Dipilih
0	Invade langsung
2	Invade masih dipilih
5	Farming sendiri lebih disukai
10	Invade hampir tidak pernah dipilih

Hasil tersebut menunjukkan bahwa parameter risiko dapat digunakan sebagai mekanisme untuk menyesuaikan gaya permainan jungler. Pada kondisi permainan yang aman, nilai k dapat diperkecil sehingga A* lebih agresif melakukan invade. Sebaliknya, ketika tim sedang tertinggal atau kehilangan vision map, nilai k dapat diperbesar agar algoritma lebih konservatif.

VI. KESIMPULAN

Makalah ini telah mengusulkan dan menganalisis penerapan algoritma A* untuk optimasi jalur rotasi jungler pada game Mobile Legends: Bang Bang dengan pemodelan yang mencakup seluruh peta permainan. Kontribusi utama makalah ini adalah: (1) perluasan model graf dari satu sisi jungle menjadi satu map penuh dengan 17 node yang mencakup jungle kedua tim serta objektif bersama Turtle dan Lord, (2) definisi bobot edge yang menyertakan faktor risiko invade sehingga A* dapat secara otomatis mempertimbangkan trade-off antara keuntungan dan risiko, serta (3) analisis tiga skenario utama pergerakan jungler yang semuanya dapat diselesaikan secara optimal oleh satu instansi A*.

Hasil analisis teoritis menunjukkan A* adalah pilihan terbaik untuk masalah ini karena menjamin optimalitas, bekerja pada graf berbobot, dan efisien secara komputasional. Sebagai pengembangan lebih lanjut, implementasi simulasi program dapat dilakukan untuk memvalidasi model secara empiris, serta eksplorasi Dynamic A* untuk menangani perubahan risiko secara real-time sesuai posisi musuh yang terus berubah dalam game.

REFERENCES

- [1] Moonton, "Mobile Legends: Bang Bang," [Game]. Google Play Store & App Store, 2016.
- [2] Moonton, "Mobile Legends: Bang Bang — Patch Notes dan Role Guide," 2026. [Online]. Available: <https://m.mobilelegends.com> [Diakses: Juni 2026].
- [3] D. Foad, A. Ghifari, M. B. Kusuma, N. Hanafiah, and E. Gunawan, "A Systematic Literature Review of A* Pathfinding," *Procedia Computer Science*, vol. 179, pp. 507–514, 2021.
- [4] P. E. Hart, N. J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 4th ed. Cambridge, MA: MIT Press, 2022.
- [6] M. Buro, "Real-Time Strategy Games: A New AI Research Challenge," in *Proc. IJCAI*, 2003, pp. 1534–1535.
- [7] A. Stentz, "The Focussed D* Algorithm for Real-Time Replanning," in *Proc. IJCAI*, 1995, pp. 1652–1659.
- [8] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Hoboken, NJ: Pearson, 2020.
- [9] R. Firmansyah, "Simak Penjelasan Map dan Lane Terbaru Mobile Legend: Jungle, Roaming, Gold, Exp, dan Mid Lane," *Kabar Lumajang - Pikiran Rakyat*, 2022. [Online]. Available: <https://kabarlumajang.pikiran-rakyat.com/ipetek/pr-42781656/simak-penjelasan-map-dan-lane-terbaru-mobile-legend-jungle-roaming-gold-exp-dan-mid-lane>. [Diakses: Juni 2026].

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, 19 Juni 2026



Muh. Hartawan Haidir 13524147