

Implementasi Algoritma A* (A-Star) dan Greedy Dalam Penentuan Routing Pada Wide Area Network (WAN)

Alfry Aristo Jansen Sinlae^{1,*}, Rini Nuraini², Dedy Alamsyah³, Sampurna Dadi Riskiono⁴

¹Fakultas Teknik, Ilmu Komputer, Universitas Katolik Widya Mandira, Kupang, Indonesia

²Fakultas Teknologi Komunikasi dan Informatika, Informatika, Universitas Nasional, Jakarta, Indonesia

³Fakultas Teknik, Teknik Informatika, Universitas Muhammadiyah Tangerang, Tangerang, Indonesia

⁴Fakultas Teknik dan Ilmu Komputer, Teknik Elektro, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

Email: ^{1,*}alfry.aj@unwira.ac.id, ²rini.nuraini@civitas.unas.ac.id, ³dedy.alamsyah@umt.ac.id, ⁴sampurna.go@teknokrat.ac.id

Email Penulis Korespondensi: alfry.aj@unwira.ac.id

Abstrak—Perkembangan terus-menerus dari internet telah menyebabkan perubahan dalam kompleksitas routing. Dalam konteks jaringan area luas (WAN), optimisasi aliran jaringan menjadi isu utama dalam pemilihan rute tercepat, yang melibatkan protokol routing. Algoritma-algoritma pada protokol routing dirancang untuk menentukan jalur terpendek. Maka dari itu, dibutuhkan kajian terkait bagaimana membangun algoritma routing yang dapat dimanfaatkan untuk mendapatkan jalur terpendek dalam setiap prosesnya. Penelitian ini bertujuan untuk mengurangi beban kerja jaringan dan membandingkan komputasi yang terlibat dalam penentuan jalur terpendek menggunakan algoritma Greedy dan algoritma A*. Hasil simulasi menunjukkan bahwa algoritma A* lebih unggul daripada greedy. Hasil pengukuran menunjukkan bahwa biaya akhir algoritma Greedy yaitu 44 dan algoritma algoritma A* yaitu 38. Dengan mempertimbangkan nilai heuristik untuk setiap node, algoritma A* lebih unggul dari pada algoritma Greedy pada penentuan jalur terpendek.

Kata Kunci: Algoritma A*; Algoritma Greedy; Protokol Routing; Wide Area Network; WAN

Abstract—The continuous development of the internet has led to changes in routing complexity. In the context of a wide area network (WAN), network flow optimization becomes a major issue in the selection of the fastest route, which involves routing protocols. Algorithms in routing protocols are designed to determine the shortest path. Therefore, studies are needed regarding how to build a routing algorithm that can be used to get the shortest path in each process. This study aims to reduce network workload and compare the computation involved in determining the shortest path using the Greedy algorithm and the A* algorithm. The simulation results show that the A* algorithm is superior to greedy. The measurement results show that the final cost of the Greedy algorithm is 44 and the A* algorithm is 38. Taking into account the heuristic value for each node, the A* algorithm is superior to the Greedy algorithm in determining the shortest path.

Keywords: A* Algorithm; Greedy Algorithm; Routing Protocol; Wide Area Networks; WAN

1. PENDAHULUAN

Teknologi Wide Area Network (WAN) memungkinkan penyediaan layanan internet di berbagai wilayah geografis, yang difasilitasi oleh beberapa penyedia layanan telekomunikasi. Jaringan WAN adalah jenis jaringan yang dapat dimanfaatkan oleh bisnis atau pemerintah untuk menghubungkan kantor atau lokasi yang berbeda [1], [2]. Perangkat seperti router, bridges, serat optik dan infrastruktur jaringan lainnya diperlukan dalam jaringan WAN untuk menghubungkan segmen atau wilayah yang berbeda. Memperluas arsitektur jaringan akan berpengaruh pada kompleksitas tautan komunikasi jaringan. Jalur komunikasi yang kompleks ditentukan oleh berbagai faktor seperti kebutuhan akan bandwidth, desain jaringan fisik, seberapa cepat setiap perangkat memproses paket, cara memilih jalur terbaik untuk pengiriman paket dan banyak lagi faktor yang lain. Salah satu tantangan utama dalam menentukan jalur terpendek adalah optimalisasi aliran jaringan [3]. Ide mendasar di balik pemilihan rute terdekat pada jaringan yaitu bagaimana membuat model jalur yang terdekat agar memperoleh waktu yang minimal dari sebuah protokol routing.

Router adalah komponen jaringan komputer yang bertugas untuk mengirimkan paket data ke alamat tujuan dengan menggunakan protokol routing [4]. Routing beroperasi pada lapisan ke-2 atau ke-3 dari model jaringan, yang bertanggung jawab untuk mengirimkan data di antara jaringan agar dapat terhubung. Selama pembaruan status jaringan, tanggung jawab utama router adalah memberi tahu kondisi terakhir terhadap neighboring nodes dijaluannya. Protokol routing harus dapat menjelaskan elemen-elemen kunci, seperti bagaimana menawarkan pembaruan, melakukan updating pengetahuan, kapan menyediakan pembaruan, dan bagaimana mentransfer informasi ke router selanjutnya untuk menentukan jalurnya [5]. Manajemen jaringan yang tepat dan teknik perutean yang efisien diperlukan untuk mentransfer komunikasi dari sumber ke komputer target. Manajemen menjadi semakin menantang dan kompleks seiring pertumbuhan jaringan. Pada dasarnya routing merupakan suatu proses untuk pemilihan jalur. Melalui desain statis atau dinamis, protokol routing berusaha agar dapat memberikan pengaturan dalam memodelkan perutean. Operasi ini bertujuan agar mempercepat dan meningkatkan ketepatan proses routing untuk sampai ke jalur pengiriman pesan [6], [7]. Untuk itu pada proses routing memerlukan algoritma pencarian agar dapat berjalan secara efektif. Beberapa algoritma yang dapat digunakan dalam permasalahan pencarian yakni algoritma A* dan algoritma Greedy. Penentuan pendekatan yang tepat akan berpengaruh pada kinerja dari model yang dibangun [8]. Algoritma Greedy merupakan teknik untuk memperoleh jalur terbaik yang didasari pada penentuan titik terkecil melalui perhitungan jalur yang terlewat [9]. Pendekatan Greedy memperhitungkan node pada tiap jalur dengan memperhatikan tahapan pada masing-masing

node yang dilewati dimulai dari node awalnya [10]. Berbeda dengan algoritma A-Star (A*) yang merupakan penggabungan dari algoritma Greedy Best First dan Algoritma Uniform Cost [11], dimana algoritma ini memperkirakan cost secara heuristik dan menentukan cost yang paling rendah dari node pertama hingga node terakhir. Pendekatan A* memperhitungkan estimasi jarak berdasarkan rute terdekat menuju tujuan secara heuristik kemudian digunakan sebagai pertimbangan untuk menentukan beberapa rute agar tercapai rute terbaik [12].

Terdapat beberapa penelitian terdahulu yang menerapkan algoritma A* dan Greedy untuk memperoleh jalur terdekat dalam permasalahan pencarian. Diantaranya penelitian tersebut yaitu penelitian mengenai penerapan pendekatan A* untuk mendapatkan jalur terpendek pada Game Rogue-like [13], pencarian jalur terbaik dengan algoritma A* untuk angkutan umum [14], penerapan algoritma Greedy Best First untuk mendapatkan rute yang efektif menuju lokasi instansi penting di Kota Argamakmur [15] dan pencarian Masjid di Samarinda menggunakan Greedy Best First agar didapatkan ruter terbaik.

Perbedaan penelitian ini dengan penelitian terdahulu yakni algoritma pencarian seperti A* dan Greedy pada penelitian diatas diterapkan untuk pembangunan sistem pencarian rute terdekat, sedangkan untuk penelitian ini algoritma pencarian tersebut diimplementasikan untuk protocol routing agar dapat diperoleh jalur yang efektif dan maksimal. Hal ini disebabkan karena teknik dalam menentukan jalur terpendek seperti algoritma A* dan algoritma Greedy menghasilkan graph sehingga dapat digunakan pada protokol routing dalam jaringan WAN. Tujuan utama dari penelitian ini adalah untuk melakukan simulasi dan membandingkan efisiensi algoritma graf A* dan Greedy dalam protokol routing. Tujuannya adalah untuk menentukan keefektifannya dalam mencapai algoritma pencarian yang efisien untuk tujuan perutean. Melalui simulasi protokol routing yang dilakukan dengan menerapkan algoritma A* dan algoritma Greedy dapat digunakan sebagai acuan optimalisasi aliran jaringan, khususnya dalam proses komputasi penentuan rute dengan jarak terpendek yang diperlukan untuk menempuh perjalanan ke tujuan dan untuk mengurangi beban kerja jaringan.

2. METODOLOGI PENELITIAN

s2.1 Tahapan Penelitian

Pada penelitian ini, akan dibangun simulasi model routing pada jaringan WAN dengan mengembangkan topologi menggunakan algoritma A* dan Greedy. Selain itu, penelitian ini juga melibatkan pengukuran kinerja setiap pendekatan yang dibandingkan melalui skema perutean yang telah direncanakan. Tahapan atau fase penelitian yang diterapkan divisualisasikan dalam Gambar 1.



Gambar 1. Alur Penelitian

Pada Gambar 1 terlihat fase-fase menggambarkan alur penelitian yang akan dilakukan. Untuk rincian dari fase-fase tersebut adalah sebagai berikut:

- 1) Studi literatur
Pada tahapan ini, dilakukan peninjauan literatur yang relevan agar memperoleh informasi yang cukup untuk pelaksanaan penelitian.
- 2) Perancangan dan Simulasi
Perancangan dalam penelitian ini melibatkan pemodelan sistem berdasarkan analisis dan permasalahan yang ada [16]. Pada tahap ini, dilakukan perancangan model perutean yang akan dievaluasi menggunakan algoritma Greedy dan A*. Selanjutnya, dilakukan simulasi dan pengujian terhadap model yang telah berhasil dikembangkan untuk mendapatkan hasil yang lebih terperinci.
- 3) Analisis dan Evaluasi
Untuk tahapan ini, analisis dan evaluasi hasil dari prosedur evaluasi dilakukan. Evaluasi yang dilakukan melalui pengujian agar dapat melakukan pengukuran performa dari model yang dibangun [17]. Artinya pada

tahap ini tujuannya adalah untuk mendapatkan pengetahuan yang lebih mendalam mengenai penerapan algoritma A* dan Greedy.

4) Hasil dan Kesimpulan

Pada tahapan akhir, akan diuraikan temuan dari penelitian yang telah dikerjakan. Tujuannya, agar temuan penelitian yang telah dikerjakan dapat memberikan data untuk penelitian yang lebih lanjut.

2.2 Routing

Routing adalah proses penentuan jalur atau rute terbaik bagi paket data untuk melakukan perjalanan antara dua atau lebih node jaringan dalam jaringan komputer [18]. Dalam jaringan komputer, node jaringan dapat berupa komputer, server, router, atau perangkat jaringan lainnya yang dapat menerima, mengirim, atau merutekan paket data. Perutean adalah fungsi penting dalam jaringan komputer, karena memungkinkan transmisi data yang efisien dan andal antar perangkat di jaringan. Proses perutean melibatkan penentuan jalur terbaik untuk diambil paket data dari sumber ke tujuan berdasarkan sejumlah faktor seperti topologi jaringan, bandwidth yang tersedia, tingkat kemacetan, dan kondisi jaringan lainnya. Ada berbagai jenis protokol dan algoritma perutean yang digunakan dalam jaringan komputer, termasuk perutean statis, perutean dinamis, dan perutean hibrid. Perutean statis melibatkan konfigurasi rute jaringan secara manual, sedangkan protokol perutean dinamis menggunakan algoritme untuk secara otomatis menentukan jalur terbaik berdasarkan kondisi jaringan waktu nyata. Routing hybrid menggabungkan keuntungan dari protokol routing statis dan dinamis. Routing digunakan di berbagai jenis jaringan komputer, termasuk local area network (LAN), wide area network (WAN) dan internet [19]. Ini juga merupakan fungsi mendasar dalam pengoperasian internet, di mana paket data dialihkan antara beberapa node untuk mencapai tujuannya.

2.3 Algoritma A-Star (A*)

Algoritma A* merupakan algoritma pencarian jalur yang diaplikasikan dalam ilmu komputer untuk menemukan jalur terpendek antara dua titik pada grafik. Ini sangat berguna dalam permainan video dan robotika, di mana agen perlu menavigasi melalui lingkungan yang kompleks. Algoritma bekerja dengan memperluas simpul yang paling menjanjikan dalam grafik berdasarkan perkiraan heuristik jarak ke tujuan. Heuristik adalah fungsi yang memperkirakan sisa jarak dari node ke tujuan, dan algoritme A* menggunakan heuristik ini untuk memprioritaskan node mana yang akan dijelajahi selanjutnya. Ini menggabungkan fitur terbaik dari algoritme Dijkstra dan pencarian pertama yang rakus, menjadikannya efisien dan efektif.

Algoritma A* dimulai dari node awal dan menjelajahi grafik dengan cara yang meminimalkan jumlah biaya untuk mencapai node dan estimasi heuristik dari jarak yang tersisa ke tujuan. Itu terus menjelajahi node hingga mencapai node tujuan atau menentukan bahwa tidak ada jalur menuju tujuan. Salah satu manfaat dari algoritma A* adalah menjamin untuk menemukan jalur terpendek jika fungsi heuristik yang sesuai digunakan. Kinerja algoritma dapat dipengaruhi oleh pemilihan fungsi heuristik, tetapi dengan heuristik yang baik, dapat jauh lebih cepat daripada algoritma pencarian jalan lainnya [20]. Metode ini didasarkan pada persamaan (1).

$$f(n) = g(n) + h(n) \tag{1}$$

Keterangan :

$h(n)$ = taksiran cost pada titik n ke tujuannya

$g(n)$ = jalur cost atau rute

$f(n)$ = hasil taksiran cost terendah pada titik n dalam menuju tujuannya

Untuk nilai $h(n)$ dapat diperoleh berdasarkan persamaan Heuristic Euclidean Distance (HED) yang dapat dilihat pada persamaan (2).

$$d(a,b) = \sqrt{(xb - xa)^2 + (yb - ya)^2} \tag{2}$$

Keterangan :

$d(a,b)$ = jarak pada node a serta b

x = value absis pada node

y = value ordinat pada node

Algoritme A-Star dapat menghasilkan loop dengan akurasi yang lebih tinggi berkat heuristik yang lebih presisi, yang mempercepat perhitungan. Algoritma A* akan menghasilkan hasil yang lebih baik. Mungkin $O(p)$, dimana p merupakan total jalur yang tinggi jika heuristik yang digunakan akurat (yaitu, secara konsisten menetapkan jarak rute minimum antara dua node). Menetapkan apakah nilai heuristik setelah penerapan terlalu tinggi atau terlalu rendah dan apakah ini berdampak pada akurasi.

2.4 Algoritma Greedy

Algoritma Greedy merupakan pendekatan yang paling populer dalam mencapai kinerja optimal dalam permasalahan pencarian. Greedy jika diterjemahkan berarti rakus, tamak, atau serakah. Ini disebabkan karena algoritma ini cara kerjanya yaitu melakukan pengambilan dari apa bisa di dapatkan untuk memperoleh jarak

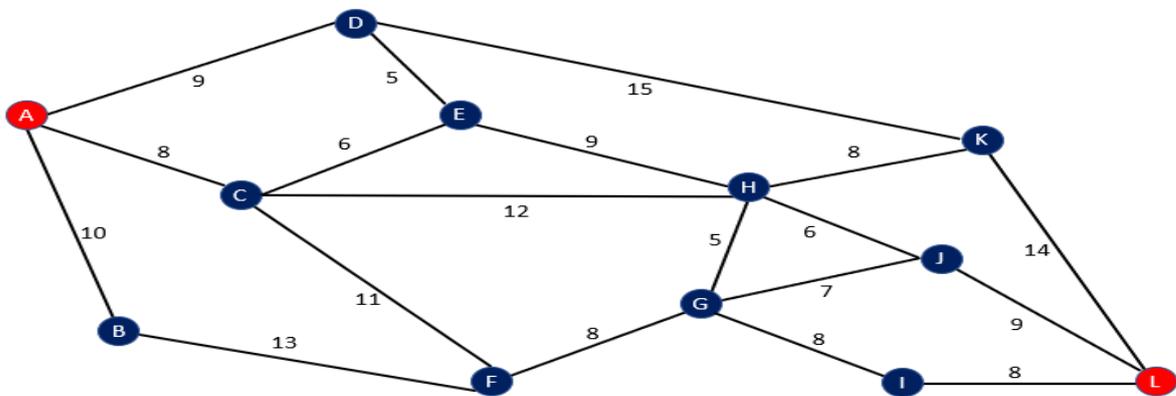
terpendek [21]. Algoritma Greedy mendapatkan rute terpendek dalam setiap langkahnya melalui pencapaian alternatif terbaik dari seluruh rute yang ada. Algoritma Greedy melakukan pemilihan alternatif terpendek dalam masing-masing tahapan dengan memperhitungkan dampak dari setiap tahap sebelumnya. Algoritma ini mendapatkan alternatif yang lebih baik dari alternatif sebelumnya, akan tetapi tidak ada jaminan alternatif terbaik secara keseluruhan. Algoritma Greedy melakukan pemilihan jalur terbaik dengan mengambil pada setiap node sepanjang perjalanan sampai menemukan jalur terpendek pada seluruh rute yang ada [22].

3. HASIL DAN PEMBAHASAN

Di bagian ini, algoritma Greedy dan A* akan dibandingkan dengan melihat masing-masing performanya. Performa akan dibentuk dengan menggunakan rute yang akan dipilih berdasarkan tarif termurah. Untuk mengukur kemampuan antara kedua algoritma yaitu A* dan Greedy, maka disusun pengembangan model rute untuk dilakukan pengujian. Dimana yang menjadi titik awal adalah node A, sedangkan node L merupakan tujuan akhir dari rute.

3.1 Algoritma Greedy

Untuk nilai jarak dari masing-masing node sudah ditentukan sebelumnya. Rancangan rute untuk pengujian menggunakan algoritma Greedy dapat dilihat pada Gambar 2.



Gambar 2. Rancangan rute untuk pengujian menggunakan algoritma Greedy Best First

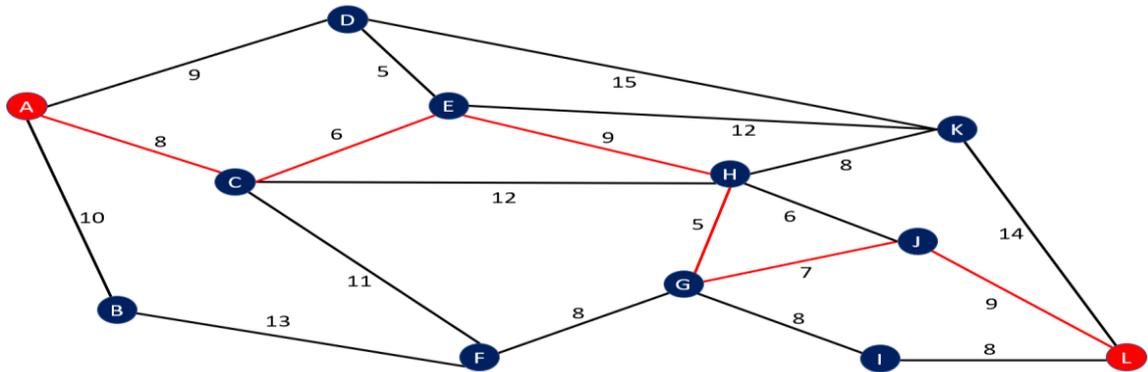
Pada Gambar 2, terlihat rancangan pengujian dengan jalur node A menuju node L yang akan dicari rute terdekatnya. Dari skema yang ada, node A berpotensi untuk memilih tiga rute, yaitu node B, C dan D. Dari sini algoritma Greedy akan mengevaluasi jarak terpendek yang tersedia. Sehingga node B akan dipilih sebagai tujuan berikutnya, hal ini didasarkan pada nilai jarak paling pendek yang didapatkan yaitu 8, sementara node yang lain memiliki nilai jarak yang lebih besar, yaitu 9 dan 10. Langkah ini akan terus dilakukan sampai menuju node akhir yaitu L. Tabel 1 menguraikan alur penentuan rute dengan algoritma Greedy.

Tabel 1. Alur komputasi algoritma Greedy

Proses Komputasi	Node		Nilai	Hasil	Urutan
	Asal	Tujuan			
1	A	B	10	10	3
		C	8	8	1
		D	9	9	2
2	C	E	6	14	4
		F	11	19	5
		H	12	20	6
3	E	H	9	23	7
		K	12	26	8
4	H	G	5	28	9
		J	6	29	10
		K	8	31	11
5	G	J	7	35	12
		I	8	36	13
6	J	L	9	44	14

Berdasarkan Tabel 1, dapat dilihat bahwa hasil kalkulasi dari algoritma Greedy, maka diperoleh jarak keseluruhan dari node A menuju ke node L adalah sebesar 44 dengan jumlah keseluruhan komputasi sebanyak 6 proses dalam menemukan node tujuan akhir, sebagaimana diperlihatkan pada Tabel 1. Rangkaian node yang

harus dilewati dari node A menuju node L adalah node C, E, H dan G. Jalur ini didapat dari proses 1 yaitu node A menuju node C, kemudian node C menuju ke node E, selanjutnya node E menuju ke node H, dilanjutkan node H menuju ke node G, berikutnya adalah node G menuju ke node J dan terakhir ditutup dengan node J menuju ke akhir node yaitu L. Hasil pemilihan jalur menggunakan Greedy tersaji pada Gambar 3.

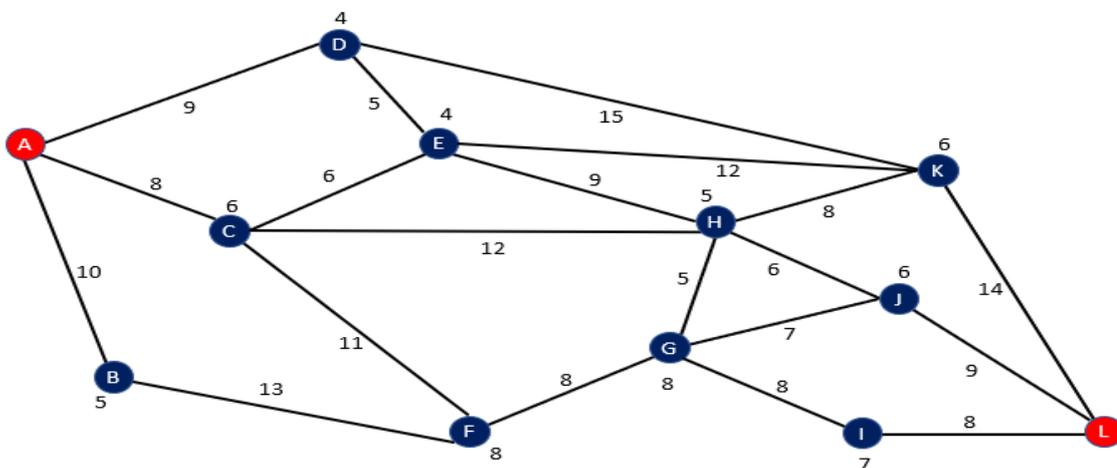


Gambar 3. Hasil pemilihan jalur menggunakan Greedy

Gambar 3 menunjukkan nilai yang didapatkan dari perhitungan algoritma Greedy bisa saja diperoleh rute yang tidak merepresentasikan total jarak yang terpendek, karena jarak yang menjadi pertimbangan bukanlah merupakan keseluruhan total jarak dengan nilai yang paling kecil. Namun algoritma greedy melakukan proses terhadap nilai node yang dianggapnya memiliki nilai yang paling kecil dari node yang akan dipilih sebagai node selanjutnya.

3.2 Algoritma A*

Berbeda dengan Algoritma Greedy, Algoritma A* mendasarkan pilihannya pada simpul berikutnya untuk dilalui tidak hanya pada jarak $g(n)$ tetapi juga pada perkiraan jarak $h(n)$ dari simpul berikutnya ke simpul tujuan sebagaimana ditentukan oleh Persamaan Ecludian Distance. Menarik garis lurus dari simpul n ke simpul tujuan dan menambahkannya ke total jarak dari simpul awal ke simpul n dalam menghasilkan perkiraan jarak. Untuk pengujian algoritma A*, digunakan model rute sebagaimana diperlihatkan pada Gambar 4.



Gambar 4. Model rute untuk algoritma A*

Pada Gambar 4, pengujian ini akan melakukan perhitungan jarak node A, yang akan terhubung ke node L. A dapat memilih antara tiga rute yang berbeda dari node awal, yaitu node B, C, dan D. Cara kerja algoritma A* berbeda dengan algoritma Greedy, dimana algoritma Greedy hanya akan memilih node jarak terpendek antara setiap rute yang layak. Nilai heuristik yang diberikan ke setiap node akan dipertimbangkan oleh algoritma A*. Untuk kejelasan dan penyederhanaan, kami membagi prosedur dalam penelitian ini menjadi dua kategori, terbuka dan tertutup, seperti yang ditunjukkan pada Tabel 2.

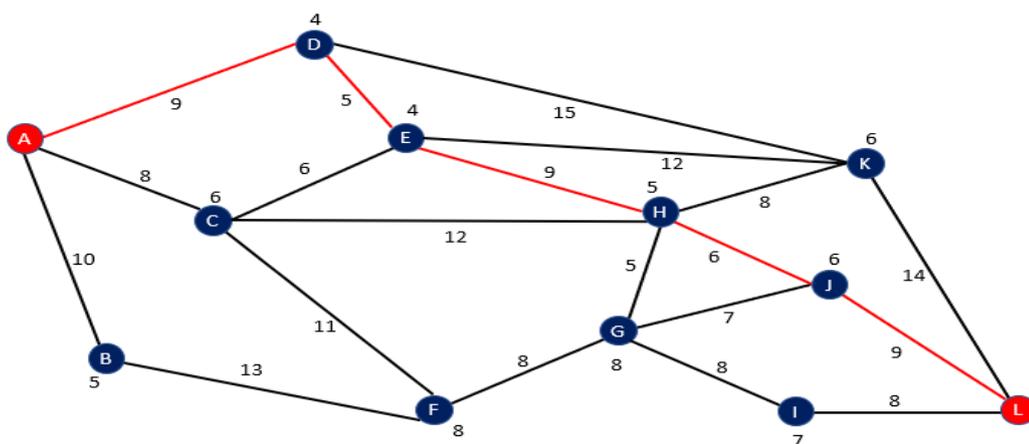
Tabel 2. Tahapan perhitungan algoritma A*

Langkah	Open List			Close List			
	Tujuan	Harga	Via	Tujuan	Harga	Via	Jarak
1	B	15	A	A	0	A	0
	C	14	A				
	D	13	A				

Langkah	Open List			Close List			
	Tujuan	Harga	Via	Tujuan	Harga	Via	Jarak
2	B	15		A	0	A	0
	C	14		D	13	A	9
3	E	18	D	A	20	A	0
	K	30	D	D	13	A	9
				E	18	D	5
4	H	19	E	A	20	A	0
	K	23	E	D	13	A	9
				E	18	D	5
				H	19	E	9
5	G	22	H	A	20	A	0
	J	21	H	D	13	A	9
	K	23	H	E	18	D	5
				H	19	E	9
				J	21	H	6
6	L	9	J	A	20	A	0
				D	13	A	9
				E	18	D	5
				H	19	E	9
				K	20	H	6
				L	14	J	9
						38	

Seperti yang terlihat pada Tabel 2, daftar kosong menggambarkan kemungkinan setiap kemungkinan rute. Tiga rute akan diperiksa pada langkah 1 dari prosedur penentuan awal, dan setiap simpul tujuan akan dinilai berdasarkan simpul yang akan dilaluinya setelah simpul sebelumnya. Algoritme A* selanjutnya akan mengevaluasi dan memberi label jarak yang terasa paling pendek dari sini. Nilai terkecil dari rute tersebut akan ditransfer dan ditempatkan di daftar terdekat setelah membandingkan nilai dari setiap jarak potensial. Hasil pengukuran nilai terkecil yang diterima dari open list dikunci dalam close list. Prosedur ini akan terus berjalan hingga node L sebagai tujuan akhirnya.

Berdasarkan pemilihan rute menggunakan Algoritma A*, maka total jarak yang ditempuh dari node A ke node L adalah 38, sebagaimana yang diperlihatkan pada Tabel 2. Urutan dari setiap node A, D, E, H, K, dan L. Hasil penelusuran dengan algoritma A* diperlihatkan oleh Gambar 5.



Gambar 5. Penelusuran jalur menggunakan algoritma A*

Terlepas dari posisi yang paling dekat dengan sebuah node, algoritma A* dapat memindahkannya ke posisi yang berbeda atau membuatnya tidak terkait dengan node lain. Algoritma A* ideal untuk topologi jaringan yang memprioritaskan pengiriman paket tunggal sesuai dengan informasi yang tersedia. Sangat cocok untuk jaringan yang berkonsentrasi pada satu tujuan. Sehingga untuk merutekan paket menggunakan algoritma A* ke beberapa tujuan, prosedur router harus diulang. Berdasarkan perbandingan keluaran algoritma Greedy dan A*, terdapat perbedaan nilai antara kedua algoritma tersebut saat memutuskan jalur mana yang harus diambil. Berdasarkan Tabel 3, algoritma A* mengungguli algoritma Greedy dalam pengujian ini, dengan nilai biaya akhir sebesar 38 dibandingkan dengan 44 untuk pendekatan Greedy.

Tabel 3. Hasil pemilihan rute dari kedua algoritma

Algoritma	Cost
Greedy	44
A*	38

Pada Tabel 3 memperlihatkan hasil pengujian pada protokol routing WAN dengan menggunakan algoritma pencarian Greedy dan A*, dimana algoritma A* lebih unggul. Hal ini disebabkan karena algoritma A* mendasarkan pilihannya pada simpul berikutnya untuk dilalui tidak hanya pada jarak $g(n)$ tetapi juga pada perkiraan jarak $h(n)$ dari simpul berikutnya ke simpul tujuan sebagaimana ditentukan oleh Persamaan Ecludian Distance. Sehingga algoritma A* ideal untuk topologi jaringan yang memprioritaskan pengiriman paket tunggal sesuai dengan informasi yang tersedia

4. KESIMPULAN

Dari hasil pengujian maka didapatkan algoritma A* lebih baik dari pada greedy. Hal ini dibuktikan dari hasil pengukuran terhadap kedua algoritma, yang mana biaya akhir untuk algoritma greedy adalah 44, sedangkan biaya akhir dari algoritma A* adalah 38. Dengan mempertimbangkan nilai heuristik yang ada pada setiap node, maka A* lebih unggul dalam menentukan jalan yang akan dilalui.

REFERENCES

- [1] M. Tahir, *Pengantar Jaringan Komputer Dasar*. Malang: CV Literasi Nusantara Abadi, 2023.
- [2] D. F. Waidah, D. D. Putra, and S. Syarifuddin, "Perancangan Sistem Jaringan dan Komunikasi Data PT. Wira Penta Kencana," *J. TIKAR*, vol. 2, no. 2, pp. 140–152, 2021.
- [3] R. D. Marcus, R. A. Saputro, and F. Y. Pamuji, "Optimasi Jaringan Routing Open Shortest Path First Dengan Menggunakan Multiprotocol Label Switching," *BRILIANT J. Ris. dan Konseptual*, vol. 5, no. 1, pp. 612–618, 2020.
- [4] S. Pujowati and B. B. Harianto, *Pengenalan Dasar Jaringan Komputer*. Magelang: Penerbit Pustaka Rumah C1nta, 2021.
- [5] M. F. Pratama, A. S. Y. Irawan, and A. Suharso, "Implementasi Routing Pada Jaringan Local Area Network Menggunakan Router di PT. Surya Baja Teknik Dan Surya Rasa (Studi kasus: PT. Surya Baja Teknik dan Surya Rasa)," *J. Ilm. Wahana Pendidik.*, vol. 6, no. 3, pp. 295–307, 2020, doi: 10.5281/zenodo.5267122.
- [6] S. D. Riskiono and D. Darwis, "Peran Load Balancing Dalam Meningkatkan Kinerja Web Server Di Lingkungan Cloud," *Krea-TIF*, vol. 8, no. 2, p. 1, 2020, doi: 10.32832/kreatif.v8i2.3503.
- [7] I. Ilahi, *Administrasi Infrastruktur Jaringan*. Surabaya: XP Solution Surabaya, 2020.
- [8] I. Komang and S. D. Riskiono, "Rancang Bangun Sistem Pengunci Loker Otomatis Dengan Kendali Akses Menggunakan RFID dan SIM 800L," *J. Ilm. Mhs. Kendali dan List.*, vol. 1, no. 1, pp. 33–41, 2020.
- [9] N. N. Sania and I. Sari, "Implementasi Rencana Perjalanan Wisata di Kota Bogor Menggunakan Algoritma Greedy Berbasis Website," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 2, pp. 114–130, 2019.
- [10] M. N. Santi and M. Mulyani, "Optimasi Biaya Jalur Tercepat Indarung-Unitas Menggunakan Algoritma Greedy," *Menara Ilmu*, vol. XIII, no. 11, pp. 60–69, 2019.
- [11] A. Hermawan and A. S. Tiwa, "Penerapan Algoritma A-Star untuk Pencarian Tempat Kuliner di Kota Tangerang," *J. Sist. dan Inform.*, vol. 15, no. 2, pp. 104–114, 2021, doi: 10.30864/jsi.v15i2.335.
- [12] B. T. D. Irianto, S. Andryana, and A. Gunaryati, "Penerapan Algoritma A-Star Dalam Mencari Jalur Tercepat dan Pergerakan NonPlayer Character Pada Game Petualangan Labirin Tech-Edu," *J. Media Inform. Budidarma*, vol. 5, no. 3, p. 953, 2021, doi: 10.30865/mib.v5i3.3094.
- [13] E. G. Agung, D. Eridani, and A. Fauzi, "Implementasi Metode Pathfinding dengan Algoritma A* pada Game Rogue-like menggunakan Unity," *Indones. J. Comput.*, vol. 7, no. December, pp. 81–94, 2022, doi: 10.34818/indojc.2022.7.3.677.
- [14] Y. H. Nuryoso, P. Pradjoko, and L. Lelah, "Implementasi Algoritma A-Star Untuk Mencari Rute Terpendek Angkutan Umum Kota (Studi Kasus Pada Rute Angkutan Umum Kota di Kota Sukabumi)," *JSTIE (Jurnal Sarj. Tek. Inform.*, vol. 8, no. 1, p. 21, 2020, doi: 10.12928/jstie.v8i1.13922.
- [15] Y. Darnita and R. Toyib, "Penerapan Algoritma Greedy Dalam Pencarian Jalur Terpendek Pada Instansi-Instansi Penting Di Kota Argamakmur Kabupaten Bengkulu Utara," *J. Media Infotama*, vol. 15, no. 2, 2019, doi: 10.37676/jmi.v15i2.867.
- [16] M. Akbar, Q. Quraysh, and R. I. Borman, "Otomatisasi Pemupukan Sayuran Pada Bidang Hortikultura Berbasis Mikrokontroler Arduino," *J. Tek. dan Sist. Komput.*, vol. 2, no. 2, pp. 15–28, 2021.
- [17] R. Harry, S. Pamungkas, S. D. Riskiono, and Y. Arya, "Rancang Bangun Sistem Penyiraman Tanaman Sayur Berbasis Arduino Dengan Sensor Kelembaban Tanah," *J. Ilm. Mhs. Kendali dan List.*, vol. 1, no. 1, pp. 23–32, 2020.
- [18] R. D. Septiana, D. A. Punkastyo, and N. Nugroho, "Implementasi Algoritma Greedy dan Algoritma A* Untuk Penentuan Cost Pada Routing Jaringan," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 3, no. 2, pp. 181–187, 2022.
- [19] A. Amarrulloh and S. Sidik, "Perancangan dan Implementasi Topologi WAN Menggunakan Routing Dynamic BGP Antar Cabang di PT Bank Woori Saudara Tbk," *Evolusi J. Sains dan Manaj.*, vol. 10, no. 1, pp. 10–19, 2022.
- [20] M. A. Arsyad, D. Supriyadi, V. Anggie, L. N. Hidayah, and D. Putri, "Penerapan Algoritma A Star Untuk Pencarian Rute Terpendek Puskesmas Rawat Inap Di Banyumas," in *Conference on Electrical Engineering, Telematics, Industrial Technology, and Creative Media*, 2019, pp. 74–82.
- [21] H. Puja Kekal, W. Gata, S. Nurdiani, A. J. Setio Rini, and D. Sely Wita, "Analisa Pencarian Rute Tercepat Menuju Tempat Wisata Pulau Kumala Kota Tenggarong Menggunakan Algoritma Greedy," *J. Ilm. Ilmu Komput.*, vol. 7, no. 1, pp. 9–15, 2021, doi: 10.35329/jiik.v7i1.179.
- [22] M. Furqan, Y. R. Nasution, and T. S. Nurdianti, "Penerapan Algoritma Greedy Untuk Menentukan Rute Terpendek Antar Klinik Gigi," *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 12, no. 3, p. 170, 2021, doi: 10.22303/csrld.12.3.2020.170-178.