

# Implementasi Algoritma Greedy dan Algoritma A\* Untuk Penentuan Cost Pada Routing Jaringan

Ristasari Dwi Septiana<sup>1,\*</sup>, Dimas Abisono Puskastyo<sup>2</sup>, Nurhasan Nugroho<sup>3</sup>

<sup>1</sup>Fakultas Teknologi, Program Studi Teknik Informatika, Institut Teknologi dan Bisnis Swadharma, Jakarta, Indonesia

<sup>2</sup>Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Banten, Indonesia

<sup>3</sup>Fakultas Ilmu Komputer, Program Studi Ilmu Komputer, Universitas Bina Bangsa, Banten, Indonesia

Email: <sup>1,\*</sup>[ristasari@swadharma.ac.id](mailto:ristasari@swadharma.ac.id), <sup>2</sup>[dosen00675@unpam.ac.id](mailto:dosen00675@unpam.ac.id), <sup>3</sup>[nurhasan.nugroho@binabangsa.ac.id](mailto:nurhasan.nugroho@binabangsa.ac.id)

Email Penulis Korespondensi: [ristasari@swadharma.ac.id](mailto:ristasari@swadharma.ac.id)

**Abstrak**—Peningkatan perkembangan internet saat ini menimbulkan masalah baru dalam hal optimasi jalur pada internet, khususnya pada penentuan cost dalam routing protocol. Hal ini membuat optimalisasi jalur jaringan menjadi permasalahan utama dalam memilih rute terpendek. Tujuan penelitian ini adalah untuk mengimplementasikan dan menganalisis proses pencarian rute terpendek dengan menggunakan dua algoritma, yaitu Greedy dan A\*. Algoritma A\* memiliki keunggulan dalam mengatasi beban kerja jaringan dibandingkan dengan algoritma Greedy. Dalam implementasi, kedua algoritma memiliki hasil yang sama dalam menentukan jalur pengiriman. Namun, algoritma A\* lebih efektif digunakan pada jaringan yang luas dan rumit karena memiliki perhitungan yang lebih pasti dan akurat. Dari hasil pengujian didapatkan bahwa algoritma A\* memiliki kinerja yang lebih baik dibandingkan dengan algoritma greedy pada pengujiannya. Dimana nilai biaya akhir dari algoritma greedy adalah 49, sedangkan untuk algoritma A\* adalah 48.

**Kata Kunci:** Algoritma Greedy; Algoritma A\*; Jaringan; Routing Protocol

**Abstract**— The current increase in internet development raises new problems in terms of path optimization on the internet. This makes network path optimization a major problem in choosing the shortest route. The purpose of this research is to understand and compare the process of finding the shortest route using two algorithms, namely Greedy and A\*. The A\* algorithm has an advantage in overcoming network workloads compared to the Greedy algorithm. In implementation, both algorithms have the same results in determining the delivery path. However, the A\* algorithm is more effective for use on large and complex networks because it has more certain and accurate calculations. From the test results, it was found that the A\* algorithm has better performance than the greedy algorithm in the test. Where the final cost value of the greedy algorithm is 49, while for the A\* algorithm is 48.

**Keywords:** Greedy Algorithm; A\* Algorithm; Network; Routing Protocol

## 1. PENDAHULUAN

Jaringan komputer adalah suatu sistem yang terdiri dari komputer dan perangkat jaringan yang bekerja sama untuk mencapai tujuan bersama. Jaringan komputer digunakan untuk berbagi sumber daya seperti printer, CPU, RAM, dan hard disk serta memfasilitasi metode komunikasi seperti email, chatting, dan instant messaging. Jaringan juga memudahkan akses informasi melalui browsing online. Setiap bagian dari jaringan komputer saling berinteraksi melalui permintaan dan layanan. Dalam desain ini, server menawarkan layanan sementara klien menerima layanan. Arsitektur ini biasanya disebut sebagai sistem *client-server* dan menjadi arsitektur jaringan komputer yang umum. Jaringan komputer dapat dibangun dengan beberapa metode, seperti menggunakan perangkat seperti hub, bridge, switch, dan router [1].

Router merupakan sebuah perangkat jaringan komputer yang menggunakan prosedur routing untuk mentransfer paket data ke jaringan tujuan. Fungsi utama routing, yang merupakan lapisan kedua atau ketiga dari model jaringan, adalah untuk mendistribusikan informasi antar jaringan sehingga mereka bisa terhubung. Ketika terjadi pembaharuan status jaringan, tugas utama router adalah memberitahu tetangganya tentang jalur mereka. Dalam memutuskan jalur, protokol routing harus bisa menjelaskan detail penting, termasuk bagaimana memberikan pembaharuan, bagaimana memperbarui pengetahuan, kapan saat pengiriman pembaharuan, dan bagaimana menyebarkan informasi ke router berikutnya [1]. Untuk mengirim pesan dari komputer asal ke komputer tujuan, diperlukan manajemen jaringan yang tepat dan mekanisme routing yang efektif. Semakin besar jaringan, semakin sulit dan rumit manajemennya. Routing adalah proses memilih jalur. Protokol routing bertujuan untuk mengatur bagaimana router berpartisipasi dalam routing melalui desain statis atau dinamis. Proses ini harus dilakukan secara cepat dan tepat untuk mencapai jalur pengiriman pesan [2], [3].

Untuk merancang jaringan yang menggunakan teknik routing memerlukan pendekatan pencarian untuk mendapatkan jarak dan waktu yang optimal. Oleh karena itu, kesulitan dalam mengakses jarak dan waktu harus dipertimbangkan untuk memperoleh jalur yang benar dan akurat dibutuhkan algoritma pencarian. Algoritma pencarian merupakan sebuah algoritma yang menyelesaikan permasalahan untuk mendapatkan solusi terbaik yang diperoleh berdasarkan mengevaluasi segala kemungkinan yang ada [4] Terdapat beberapa algoritma pencarian yang dapat digunakan diantaranya adalah algoritma Greedy dan algoritma A\*. Algoritma Greedy bekerja dengan menemukan titik terkecil dengan menghitung rute yang dilalui [5]. Algoritma Greedy menghitung titik tergantung pada langkah-langkah titik yang telah dilalui dan titik pada tahap itu sendiri [6]. Sedangkan Algoritma A\* atau A-Star adalah gabungan dari dua algoritma, yaitu Algoritma *Greedy Best First* dan Algoritma *Uniform Cost* [7]. Algoritma *Greedy Best First* digunakan untuk memberikan perkiraan biaya dengan heuristik, sementara algoritma *Uniform Cost* digunakan untuk menemukan biaya terendah dari awal sampai akhir. Algoritma A\* adalah algoritma yang memanfaatkan estimasi jarak melalui pencarian jalur yang paling dekat menuju tujuannya dan bersifat heuristik yang dipergunakan untuk mempertimbangkan dalam penentuan keputusan dari beberapa jalur guna mencapai apa yang dituju secara efektif [7].

Terdapat beberapa penelitian yang menggunakan algoritma Greedy dan A\* yang digunakan untuk pencarian, dari beberapa penelitian menunjukkan algoritma tersebut dapat melakukan pencarian dengan baik. Penelitian tentang algoritma Greedy antara lain penelitian mengenai penerapan algoritma Greedy untuk mencari jalur paling pendek pada instansi penting di Kota Argamakmur [8] dan penelitian tentang pencarian masjid di Samarinda melalui jalur yang terdekat dengan algoritma Greedy [9]. Pada penelitian tersebut algoritma Greedy dapat melakukan pencarian titik bobot yang paling kecil melalui perhitungan rute yang telah terlewati dan bobotnya didapatkan dari bobot tahapan yang terlewati serta berdasarkan bobot pada tahapan tersebut. dan bobot pada tahap itu sendiri. Sedangkan, untuk penelitian penerapan algoritma A\* diantaranya penelitian implementasi A\* dalam pencarian rute terpendek pada *Robot Hexapod* [10] dan penelitian tentang pencarian rute terpendek untuk angkutan umum di Kota Sukabumi [11]. Pada penelitian tersebut A\* mampu menemukan rute terpendek melalui efektifitas estimasi jarak pencarian jalur yang paling dekat menuju tujuan dan bersifat heuristik untuk mempertimbangkan jalur guna mencapai apa yang dituju.

Perbedaan penelitian ini dengan penelitian sebelumnya yaitu pada penelitian sebelumnya algoritma Greedy dan A\* digunakan pada pengembangan aplikasi pencarian rute terdekat sedangkan pada penelitian ini algoritma Greedy dan A\* digunakan untuk mencari rute *cost* pada proses *routing* jaringan sehingga didapatkan jalur *routing* yang optimal. Sehingga tujuan dari penelitian ini adalah melakukan proses *routing* dengan implementasi algoritma Greedy dan A untuk menentukan jalur *routing* terbaik. Ini dilakukan untuk membandingkan proses penentuan jalur dan menentukan jalur yang lebih cepat untuk pengiriman data ke tujuan.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Sebelum dilaksanakan penelitian perlu dilakukan penyusunan tahapan penelitian yang tersistematis, terstruktur dan direncanakan sehingga dapat mencapai apa yang menjadi tujuan dalam penelitian [12], [13]. Untuk melakukan penelitian pada pengembangan topologi membangun model yang bertujuan untuk mensimulasikan algoritma greedy dan A\* terdapat beberapa tahapan yang akan dilakukan. Dalam analisis ini, dilakukan perbandingan kerja dari masing-masing algoritma dengan skema routing yang akan digunakan. Tahapan penelitian yang dilakukan disajikan pada Gambar 2.



Gambar 1. Tahapan penelitian

- Berdasarkan tahapan penelitian pada Gambar 1, berikut ini adalah penjelasan untuk masing-masing tahapan:
- Studi Literatur**  
Pada tahap ini dilakukan kajian terhadap berbagai literatur yang mendukung yang bertujuan untuk mendapatkan informasi yang cukup dalam melakukan penelitian.
  - Perancangan dan Simulasi**  
Pada tahap ini dilakukan perancangan terhadap model routing yang akan kami ukur dengan menggunakan algoritma greedy dan A\*. Setelah model berhasil dibangun maka langkah selanjutnya adalah melakukan simulasi dan pengujian.
  - Analisis dan Evaluasi**  
Pada tahap ini kami kami melakukan analisis dan evaluasi terhadap hasil yang telah didapatkan pada proses evaluasi. Evaluasi berguna untuk mengukur kinerja dari model yang dikembangkan [14]. Sehingga pada tahap ini bertujuan untuk mendapatkan informasi yang lebih komprehensif terkait dengan penerapan dari algoritma Greedy dan A\*.
  - Hasil dan Kesimpulan**  
Pada tahap ini kami melakukan pemaparan terkait dengan hasil yang telah dilakukan. Tujuannya adalah agar dari hasil penelitian yang dilakukan dapat dimanfaatkan sebagai informasi untuk penelitian-penelitian berikutnya.

### 2.2 Routing

Routing adalah proses tertentu yang menggunakan kunci untuk mengirim data melalui jaringan komputer. Routing dilakukan pada baris kedua atau ketiga dari model jaringan dan bertujuan untuk mentransmisikan informasi antar jaringan

untuk memungkinkan komunikasi. Tugas utama router adalah memberitahu tetangga tentang jalur mereka sendiri saat status jaringan diterima. Protokol routing harus mampu mendeteksi kesalahan dengan cermat dengan memperhatikan detail-detail kritikal seperti bagaimana mengirim dan menerima data, bagaimana menangani informasi yang salah, kapan harus mengirim data, dan bagaimana mentransmisikan informasi ke router berikutnya. Routing menunjukkan kapan datagram akan dikirim untuk mencapai tujuan yang diinginkan [15], [16].

### 2.3 Algoritma A\*

A\* atau A-Star adalah algoritma digunakan untuk mendeteksi jalur dengan biaya untuk mentransfer medan dari simpul awal ke simpul target. Algoritma saat ini adalah algoritma generasi pertama yang terbaik dengan pengurangan biaya solusi yang sederhana dan penciptaan heuristik keputusan dan batas sebagai kriteria utama [17]. Algoritma A\* pertama kali diperkenalkan pada tahun 1968 oleh Peter Hart, Nils Nilsson, dan Bertram Raphael. Dalam ilmu komputer, A\* merupakan algoritma grafik terbaik yang dapat mendeteksi perilaku abnormal dengan biaya pengeluaran yang relatif rendah dari titik awal yang diberikan ke titik akhir yang diinginkan (dari satu atau lebih tujuan). Algoritma A\* (Admissible Heuristic) merupakan strategi best-first-search yang menggunakan estimasi biaya rendah untuk menyelesaikan masalah dengan toleransi yang dekat dan memiliki taman heuristic yang digunakan sebagai panduan untuk merumuskan kesimpulan. Heuristik adalah seperangkat kriteria, metode, atau seperangkat prinsip panduan yang dapat diterapkan pada operasi untuk mencapai tujuan secara efektif. Nilai heuristic digunakan untuk memberihk rapat ruangan. Algoritma A\* menemukan solusi terbaik mulai dari lokasi awal dan bergerak maju menggunakan grafik lokasi tujuan. Metode ini didasarkan pada persamaan berikut:

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

Keterangan :

$h(n)$  = estimasi cost dari node n ke tujuan

$g(n)$  = path cost atau perjalanan

$f(n)$  = solusi estimasi cost termurah dari node n untuk mencapai tujuan [2].

$h(n)$  dapat ditentukan melalui interaksi dengan rumus Heuristic Euclidean Distance (HED). Berikut ini adalah sumber untuk menemukan HED, dengan a dan b berfungsi sebagai koordinat untuk dua tick yang akan disebutkan:

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

Keterangan :

$d(a,b)$  = jarak antara titik a dan b

$x$  = nilai absis suatu titik

$y$  = nilai ordinat suatu titik

Semakin akurat heuristik yang digunakan, semakin akurat algoritma A-Star dapat mencapai perulangan, sehingga perhitungan bisa selesai lebih cepat. Jika heuristik yang digunakan akurat (yaitu, menetapkan jarak minimum rute antara dua node secara konsisten), algoritma A\* akan memberikan hasil yang lebih baik, yang dapat menghasilkan algoritma menjadi  $O(p)$ , di mana p adalah jumlah besar jalur. Menentukan apakah keakuratan heuristik dipengaruhi oleh nilai heuristik setelah diimplementasikan terlalu tinggi atau terlalu rendah.

### 2.4 Algoritma Greedy

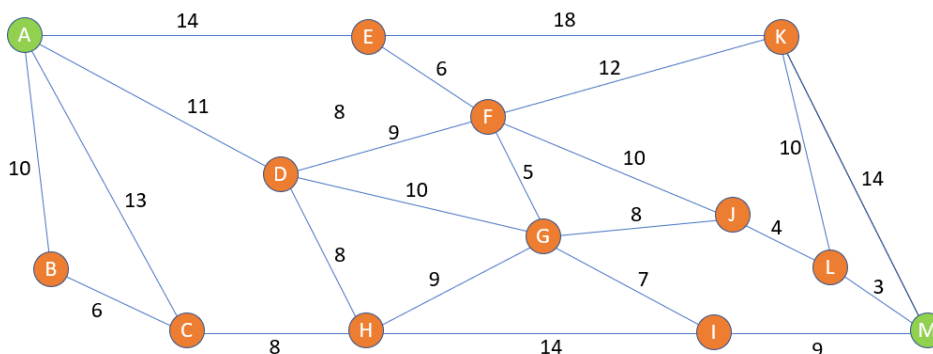
Algoritma Greedy adalah metode yang paling populer untuk mencapai kinerja yang optimal. Greedy sendiri diterjemahkan dari bahasa Inggris sebagai rakus, tamak, atau serakah. Berdasarkan kata tersebutlah yang digunakan untuk memeberikan nama algoritma Greedy dimana algoritma ini mengambil apa yang bisa di dapatkan sekarang yang menjadi prinsip dari algoritma ini [18], [19]. Algoritma greedy memilih jalur terbaik pada setiap langkah untuk mencapai solusi terbaik secara keseluruhan. Algoritma ini memilih opsi terbaik pada setiap tahap tanpa mempertimbangkan dampaknya pada tahap berikutnya. Algoritma ini menghasilkan solusi yang lebih baik dari solusi saat ini, namun tidak menjamin solusi terbaik secara keseluruhan. Algoritma Greedy memilih jalur terbaik untuk diambil pada setiap titik sepanjang perjalanan hingga mencapai jalur terpendek secara keseluruhan [20].

## 3. HASIL DAN PEMBAHASAN

Pada bagian ini kami akan melakukan perbandingan terhadap kedua algoritma greedy dan A\* untuk melihat kenerja terhadap masing-masing dari algoritma tersebut. Kinerja akan dinilai melalui penentuan rute yang akan dilalui berdasarkan biaya yang paling kecil.

### 3.1 Penerapan Algoritma Greedy

Pada penelitian ini kami telah merancang topologi yang akan digunakan dalam mengukur kinerja antara kedua algoritma yaitu greedy dan A\* dengan nilai yang sudah ditentukan terlebih dahulu. Sebagaimana diperlihatkan pada Gambar 2, topologi ini memiliki titik awal node A dengan tujuan node M dimana jarak antar node terlebih dahulu sudah tentukan.



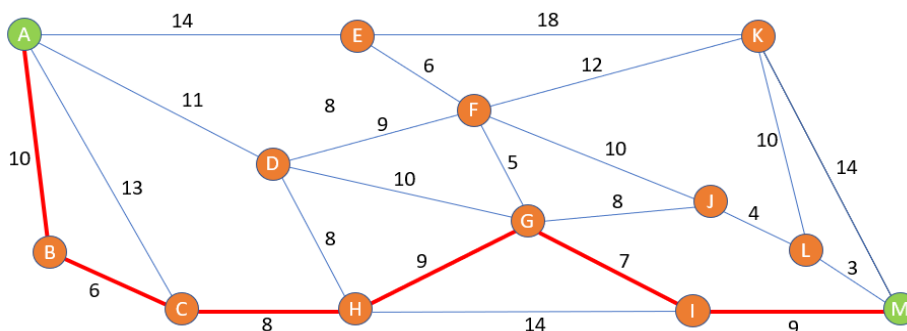
**Gambar 2.** Bentuk Topologi Untuk Algoritma Greedy

Pada Gambar 2 terlihat bahwa operasi routing dari node A, kemudian akan berakhir sampai pada node M. Node A memiliki potensi untuk memilih 4 rute dari titik awal, yaitu node B, C, D, dan E. Oleh karena itu, algoritma greedy akan memperhatikan jarak yang lebih pendek dari titik awal dan akan meletakkan jalur di node B karena jaraknya 10, sementara node lain seperti C, D, dan E masing-masing nilainya adalah 13, 11, dan 14. Prosedur saat ini akan berlanjut sampai mencapai tujuan node M berikutnya. Tabel 1 menjelaskan proses perhitungan yang terjadi ketika algoritma greedy digunakan.

**Tabel 1.** Proses Perhitungan Algoritma Greedy

Proses Routing	Node		Jarak	Hasil	Urutan
	Asal	Tujuan			
1	A	B	10	10	1
		C	13	13	
		D	11	11	
		E	14	14	
2	B	C	6	16	2
3	C	H	8	24	3
4	H	G	9	33	4
		I	14	38	
5	G	J	8	41	5
		I	7	40	
6	I	M	9	49	6

Berdasarkan tugas yang telah dilakukan, algoritma greedy akan memilih pekerjaan dengan nilai terendah pada saat dieksekusi. Total jarak dari node A ke node M diperoleh berdasarkan hasil perhitungan adalah 49. Nodes A, B, C, H, G, dan I adalah node yang dibutuhkan dari node A untuk mencapai node M. Pembentukan jalur ini didasarkan pada proses berikut: node A ke node B, node C ke node C, node H ke node H, node G ke node G, node I ke node I, dan node M ke node M. Node-node tersebut dipilih berdasarkan probabilitas jarak yang sedikit belum pernah digunakan sebelumnya. Dalam proses pertama, ditemukan bahwa node B adalah node yang lebih kecil jika dipisahkan dari node A, yang berarti node B akan digunakan untuk melakukan proses berikutnya. Kemudian sinyal akan diperpanjang dari node B dan jalur dari node B ke node C, yaitu node dengan sinyal terkecil, akan ditunjukkan dalam diagram. Proses dimulai ketika Greedy menemukan node yang akan menjadi tujuannya. Gambar 3 mewakili hasil dari algoritma pencarian jalur greedy.

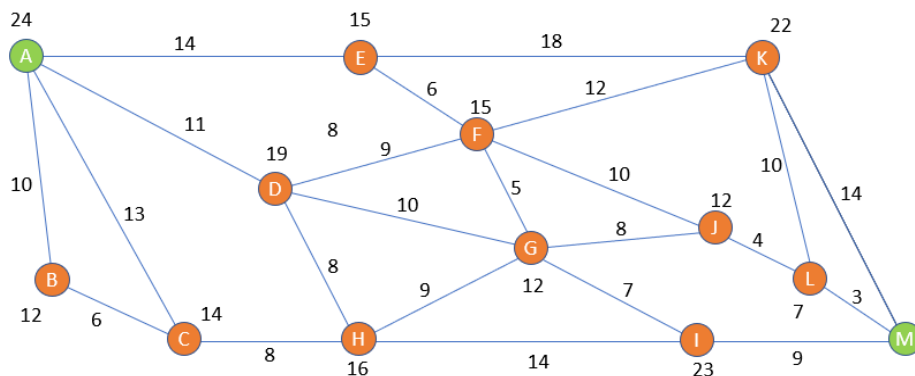


**Gambar 3.** Hasil Pemilihan Rute Menggunakan Greedy

Pada Gambar 3 terlihat hasil rute terbaik yang dihasilkan oleh algoritma Greedy dimana dihasilkan rute dari A menuju M yang terbaik adalah A-B-C-H-I-M.

### 3.2 Penerapan Algoritma A\*

Model ini, digambarkan bahwa node A ke M dengan jarak antar node dan nilai heuristic yang sudah ditentukan terlebih dahulu, sebagaimana ditunjukkan pada Gambar 4.



**Gambar 4.** Rancangan Topologi Untuk Algoritma A\*

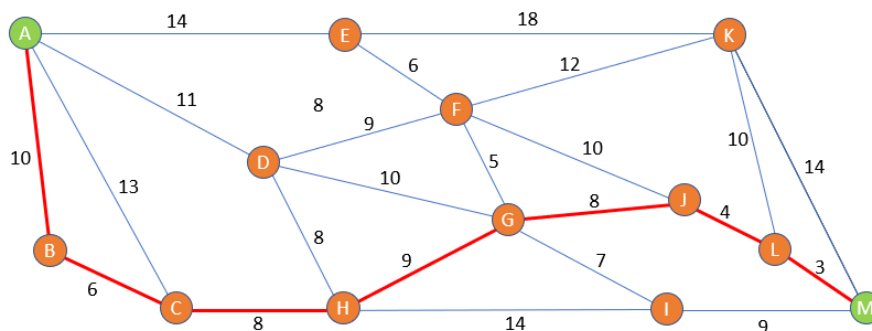
Pada Gambar 4 terlihat bahwa akan dilakukan perhitungan terhadap node A yang akan menuju node M. Dari awal node A memiliki kemungkinan 4 rute yang dapat dipilih, yaitu node B, C, D dan node E. Berbeda dengan algoritma greedy sebelumnya yang akan memilih jarak terkecil dari setiap rute yang ada, pada algoritma A\* akan mempertimbangkan nilai heuristic yang diberikan pada pada setiap simpul node. Untuk mempermudah serta memberikan gambaran yang jelas, maka dalam penelitian ini kami melakukan pengelompokan proses kedalam dua penamaan, yaitu open list dan close list sebagaimana diperlihatkan pada Tabel 2. Open list bertujuan untuk memetakan kemungkinan dari setiap rute yang akan dipilih, seperti yang diperlihatkan pada Gambar 4, maka terdapat 4 rute yang akan di proses pada step 1, masing-masing node tujuan akan dinilai harganya berdasarkan node yang akan dilaluinya berdasarkan dari node sebelumnya. Dari sini algoritma A\* akan menilai dan menandai jarak yang paling kecil. Setelah dilakukan perbandingan nilai dari setiap kemungkinan jarak yang ada, maka selanjutnya nilai terkecil dari rute akan di pindahkan dan disimpan ke close list. Jadi close list dapat dikatakan sebagai tempat untuk mengunci hasil pengukuran nilai terkecil yang diperoleh dari open list. Proses yang sama ini akan terus dilanjutkan samapi menemukan tujuan akhir yaitu node M.

**Tabel 2.** Proses Komputasi Algoritma A\*

Step	Open List			Close List		
	Tujuan	Harga	Melalui	Tujuan	Harga	Via
1	B	22	A	A	24	A
	C	27	A			
	D	30	A			
	E	29	A			
2	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A			
3	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A			
	C	20	B			
4	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
5	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	H	24	C			
6	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
7	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	G	21	H	H	24	C
I	27	H				

8	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	I	27	H	H	24	C
9	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	I	27	H	H	24	C
	J	20	G	G	21	H
10	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	I	27	H	H	24	C
	I	30	G	G	21	H
11	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	I	27	H	H	24	C
	I	30	G	G	21	H
12	C	27	A	A	10	A
	D	30	A	B	22	A
	E	29	A	C	20	B
	I	27	H	H	24	C
	I	30	G	G	21	H
				J	20	G
				L	11	J

Berdasarkan performa Algoritma A\*, mungkin untuk mendapatkan total lompatan dari node A ke node M sekitar 48, seperti yang ditunjukkan pada Tabel 2. Sejumlah node yang terlihat berurutan merupakan node A, B, C, H, G, J, L, dan M. Hasil perhitungan menggunakan algoritma A\* ditunjukkan pada Gambar 5.



**Gambar 5.** Jalur Terpendek Dengan Algoritma A\*

Pada Gambar 5, terlihat bahwa algoritma A\* dapat memindahkan sebuah node ke posisi yang berbeda atau tidak sama sekali terkait dengan node lain, selain dari posisi yang paling dekat dengan node yang bersangkutan. Dari informasi di bawah, mungkin dapat ditarik kesimpulan bahwa algoritma A\* sangat sesuai pada topologi jaringan yang fokus pada pengiriman paket tunggal, atau dapat ditarik kesimpulan bahwa itu cocok untuk jaringan yang fokus pada tugas tunggal saja. Oleh karena itu, untuk mengarahkan paket yang berbeda tujuan sesuai dengan algoritma A\*, proses router harus diulangi.

### 3.3 Perbandingan Hasil Algoritma Greedy dan A\*

Setelah melakukan analisa routing menggunakan algoritma Greedy dan A\* hasil dari rute terdekat digitung nilai *cost* yang dihasilkan kedua algoritma tersebut. Hasil pencarian ruter terdekat berdasarkan *cost* pada routing jaringan dari kedua algoritma tersebut disajikan pada Tabel 3.

**Tabel 3.** Hasil Peruteran Kedua Algoritma

Algoritma	Nilai Routing
Greedy	49
A*	48

Berdasarkan hasil perbandingan antara algoritma Greedy dan A\*, maka terdapat selisih nilai dari keduanya dalam menentukan rute yang akan dipilih. Seperti yang diperlihatkan pada Tabel 3 maka algoritma A\* memiliki kinerja yang lebih baik dibandingkan dengan algoritma greedy pada pengujian kali ini. Dimana nilai *cost* akhir dari algoritma Greedy adalah 49, sedangkan untuk algoritma A\* adalah 48.

#### 4. KESIMPULAN

Dari pengujian yang telah dilakukan dengan menggunakan masalah yang sama, maka telah ditemukan bahwa algoritma A\* lebih efisien dibandingkan Algoritma Greedy. Ini terlihat pada nilai akhir yang diperoleh dari kedua algoritma tersebut. Dimana nilai *cost* akhir dari algoritma greedy adalah 49, sedangkan untuk algoritma A\* adalah 48. Sehingga algoritma A\* lebih baik dalam memilih jalur yang akan dilalui dengan mempertimbangkan nilai heuristik yang terdapat pada masing-masing node. Namun, karena tujuannya hanya untuk mencapai satu node, algoritma Greedy akan memperoleh jarak yang lebih kecil. Karena jarak ini lebih besar dari jarak lain, algoritma Greedy akan memperoleh jalur tambahan meskipun akan menghasilkan lebih banyak hasil dan jalur yang diduga buruk.

#### REFERENCES

- [1] M. P. M. Routing, F. S. Mukti, A. Basuki, and O. Setyawati, "Pengendalian Kemacetan Jaringan," *J. EECCIS*, vol. 12, no. 1, pp. 15–19, 2018.
- [2] L. S. Moonlight and S. Suhardi, "Pengaruh Model Jaringan Terhadap Optimasi Routing Open Shortest Path First (Ospf)," *Teknologi*, vol. 1, no. 2, pp. 68–80, 2012, doi: 10.26594/teknologi.v1i2.56.
- [3] S. D. Riskiono and D. Darwis, "Peran Load Balancing Dalam Meningkatkan Kinerja Web Server Di Lingkungan Cloud," *Kreatif-TIF*, vol. 8, no. 2, p. 1, 2020, doi: 10.32832/kreatif.v8i2.3503.
- [4] R. D. Gunawan, R. Napianto, R. I. Borman, and I. Hanifah, "Penerapan Pengembangan Sistem Extreme Programming Pada Aplikasi Pencarian Dokter Spesialis di Bandarlampung Berbasis Android," *Format J. Ilm. Tek. Inform.*, vol. 8, no. 2, p. 148, 2020, doi: 10.22441/format.2019.v8.i2.008.
- [5] 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.
- [6] M. Z. Usman and T. Oktiarso, "Implementasi Algoritma Greedy untuk Menyelesaikan Travelling Salesman Problem di Distributor PT. Z," *J. Integr. Syst.*, vol. 1, no. 2, pp. 216–229, 2018.
- [7] 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.
- [8] 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.
- [9] F. Nova Arviantino, W. Gata, L. Kurniawati, Y. A. Setiawan, and D. Priansyah, "Penerapan Algoritma Greedy Dalam Pencarian Jalur Terpendek Pada Masjid–Masjid Di Kota Samarinda," *Metik J.*, vol. 5, no. 1, pp. 8–11, 2021, doi: 10.47002/metik.v5i1.188.
- [10] D. Hermanto and S. Dermawan, "Penerapan Algoritma A-Star Sebagai Pencari Rute Terpendek pada Robot Hexapod," *J. Nas. Tek. Elektro*, vol. 7, no. 2, p. 122, 2018, doi: 10.25077/jnte.v7n2.545.2018.
- [11] 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.
- [12] R. I. Borman and B. Priyopradono, "Implementasi Penerjemah Bahasa Isyarat Pada Bahasa Isyarat Indonesia (BISINDO) Dengan Metode Principal Component Analysis (PCA)," *J. Inform. J. Pengemb. IT*, vol. 03, no. 1, pp. 103–108, 2018.
- [13] R. I. Borman and A. Ansori, "Implementasi Augmented Reality pada Aplikasi Android Pegenalan Gedung Pemerintahan Kota Bandar Lampung," *J. Teknoinfo*, vol. 11, no. 1, pp. 1–5, Jun. 2017, doi: 10.33365/jti.v11i1.2.
- [14] I. Ahmad, A. T. Prastowo, E. Suwarni, and R. I. Borman, "Pengembangan Aplikasi Online Delivery Sebagai Upaya Untuk Membantu Peningkatan Pendapatan," *JMM (Jurnal Masy. Mandiri)*, vol. 5, no. 6, pp. 4–12, 2021.
- [15] I. Attamimi, W. Yahya, and M. H. Hanafi, "Analisis Perbandingan Algoritma Floyd-Warshall dan Dijkstra untuk Menentukan Jalur Terpendek Pada Jaringan Openflow," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 12, pp. 1842–1849, 2017.
- [16] L. Lukman, R. Hidayat, and M. F. R. Pribadi, "Perbandingan Algoritma A\* Dengan Algoritma Greedy Pada Penentuan Routing Jaringan," *CCIT J.*, vol. 12, no. 2, pp. 158–169, 2019, doi: 10.33050/ccit.v12i2.686.
- [17] Y. Mutsaqov, Ativ, Muhammad Fernando and D. A. Megawaty, "Penerapan Algoritma A-Star Pada Aplikasi Pencarian Lokasi Foto Berbasis Android," *Inov. Pembang. J. ....*, vol. 8, no. 1, pp. 39–52, 2020.
- [18] D. Grace, M. S. Tanciga, and Nurdin, "Sistem Informasi Letak Geografis Penentuan Jalur Tercepat Rumah Sakit Di Kota Palu Menggunakan Algoritma Greedy Berbasis Web," *J. Elektron. Sist. Inf. dan Komput.*, vol. 4, no. 2, pp. 59–76, 2018.
- [19] 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 Tenggara Menggunakan Algoritma Greedy," *J. Ilm. Ilmu Komput.*, vol. 7, no. 1, pp. 9–15, 2021, doi: 10.35329/jiik.v7i1.179.
- [20] 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/csridd.12.3.2020.170-178.