

KOMPARASI ALGORITMA *GREEDY BEST FIRST SEARCH* DAN *ANT COLONY OPTIMIZATION* UNTUK PEMILIHAN JALUR EVAKUASI BENCANA ALAM

Ade Sutedi^{*1}, Mohammad Dimas Maulana Nugraha², Leni Fitriani³, Fitri Nuraeni⁴

^{1,2,3,4}Institut Teknologi Garut, Kabupaten Garut

Email: *adesutedi@itg.ac.id, 1906105@itg.ac.id, lenifitriani@itg.ac.id, fitri.nuraeni@itg.ac.id

*Penulis Korespondensi

(Naskah masuk: 25 Juli 2026, diterima untuk diterbitkan: 03 Maret 2026)

Abstrak

Ketidaktepatan dalam pemilihan jalur evakuasi pada situasi bencana alam berpotensi menyebabkan keterlambatan proses penyelamatan dan meningkatkan risiko bagi korban. Permasalahan tersebut menjadi dasar dilakukannya penelitian ini untuk mengkaji strategi penentuan rute evakuasi yang lebih efisien. Penelitian ini bertujuan mengevaluasi dan membandingkan performa algoritma *Greedy Best First Search* (GBFS) dan *Ant Colony Optimization* (ACO) dalam menentukan jalur evakuasi dengan jarak terpendek dan waktu tempuh tercepat dari lokasi bencana ke titik evakuasi. Proses pengujian dilakukan melalui pengembangan prototipe berbasis Python yang memanfaatkan graf sebagai representasi jaringan jalur dan posisi geografis lokasi bencana, dengan pendekatan metode pengembangan prototipe. Analisis kinerja kedua algoritma difokuskan pada pengukuran waktu pemrosesan dan panjang rute yang dihasilkan. Hasil evaluasi menunjukkan adanya perbedaan kinerja kedua algoritma secara kuantitatif berdasarkan parameter yang diuji. Secara statistik, GBFS menunjukkan efisiensi komputasi yang sangat signifikan dengan waktu rata-rata >8.000 kali lebih cepat dibandingkan ACO serta tingkat keberhasilan 100%. Meskipun kualitas bobot rute setara pada sebagian besar skenario, ACO menunjukkan variabilitas waktu yang tinggi dan kegagalan solusi pada satu titik awal, namun unggul dalam menghasilkan rute alternatif. Temuan ini diharapkan dapat memberikan kontribusi sebagai bahan rujukan dalam pengembangan sistem evakuasi yang lebih efektif untuk mendukung upaya mitigasi dan penanggulangan bencana alam.

Kata kunci: ACO, bencana, evakuasi, GBFS, komparasi

COMPARATION OF GREEDY BEST FIRST SEARCH ALGORITHM AND ANT COLONY OPTIMIZATION FOR NATURAL DISASTER EVACUATION ROUTE SELECTION

Abstract

Inaccurate evacuation route selection during natural disasters can potentially delay rescue efforts and increase the risk to victims. This issue underpins this research, which explores strategies for determining more efficient evacuation routes. This study aims to evaluate and compare the performance of the Greedy Best First Search (GBFS) and Ant Colony Optimization (ACO) algorithms in determining evacuation routes with the shortest distance and fastest travel time from the disaster site to the evacuation point. The testing process was conducted through the development of a Python-based prototype that utilizes graphs to represent the path network and the geographic location of the disaster site, using a prototype development method approach. Performance analysis of both algorithms focused on measuring processing time and the length of the resulting route. The evaluation results showed quantitative differences in the performance of the two algorithms based on the tested parameters. Statistically, GBFS demonstrated highly significant computational efficiency, with an average time >8,000 times faster than ACO and a 100% success rate. Although the quality of the route weights was equivalent in most scenarios, ACO exhibited high time variability and solution failure at one starting point, but excelled in generating alternative routes. These findings are expected to contribute as reference material in developing a more effective evacuation system to support natural disaster mitigation and response efforts.

Keywords: ACO, disaster, evacuation, GBFS, comparison

1. PENDAHULUAN

Bencana merupakan peristiwa kompleks yang berdampak signifikan terhadap kehidupan manusia,

baik dari aspek sosial, ekonomi, maupun psikologis (Yunus 2025). Bencana timbul karena faktor alam, faktor non-alam, ataupun juga dapat disebabkan oleh

manusia (Bayu and Pamungkas 2023). Di Indonesia, kerentanan bencana dipengaruhi oleh kondisi geografis yang beragam dan tingkat kesiapsiagaan infrastruktur yang belum merata. Salah satu wilayah dengan tingkat kerawanan tinggi adalah Kabupaten Garut, yang memiliki karakteristik wilayah pegunungan, aliran sungai, dan kawasan pesisir menjadikannya rentan terhadap berbagai jenis bencana, terutama banjir. Berdasarkan data historis dari BPBD Garut, tercatat sebanyak 915 kejadian bencana terjadi selama periode 2020-2024, di mana banjir menjadi salah satu ancaman utama yang secara langsung berdampak pada keselamatan publik dan infrastruktur (BPBD 2025).

Tantangan utama dalam manajemen bencana meliputi berbagai aspek yang kompleks. Salah satunya adalah terletak pada proses evakuasi yang efisien dan penting untuk mengurangi korban jiwa (Dong, K. et al. 2024). Temuan ini sejalan dengan penelitian di berbagai daerah di Indonesia yang menunjukkan bahwa banjir merupakan bencana dengan frekuensi kejadian tinggi dan memerlukan penanganan yang kompleks, terutama pada wilayah (Sutedi et al. 2024) dengan keterbatasan infrastruktur pendukung evakuasi (Rivaldy et al. 2025). Karakteristik bencana ini yang bersifat dinamis seringkali menyebabkan terputusnya akses jalan secara tiba-tiba, yang merupakan urat nadi transportasi (Yulita, Kurniabudi, and Nurhadi 2023). Kegagalan dalam menemukan rute alternatif secara cepat dapat mengakibatkan isolasi wilayah dan keterlambatan fatal dalam proses penyelamatan. Kondisi ini juga dilaporkan pada studi mitigasi banjir di beberapa kota di Indonesia, dimana keterbatasan akses jalan dan perubahan kondisi lahan secara cepat menjadi faktor utama keterlambatan proses evakuasi (Mase, Fauzi, and Edriani, n.d.)

Perkembangan teknologi informasi, khususnya dalam bidang kecerdasan buatan, menawarkan kerangka kerja yang kuat untuk menghadapi tantangan tersebut. Penelitian ini membangun sebuah model graf sebagai studi kasus. Struktur dan topologi dari model graf ini didasarkan pada data jaringan jalan yang ada di Kecamatan Cikajang, Kabupaten Garut, yang digunakan sebagai dasar perancangan. Wilayah ini dipilih sebagai basis pemodelan karena secara representatif memiliki karakteristik daerah yang rentan terhadap bencana banjir. Dengan demikian, penelitian ini tidak bertujuan untuk memvalidasi data lapangan secara real-time, melainkan menggunakan topologi tersebut untuk menciptakan sebuah skenario simulasi yang realistis.

Penelitian ini secara khusus mengkaji dua algoritma yang berasal dari paradigma komputasi yang berbeda: *Greedy Best-First Search* (GBFS) dan *Ant Colony Optimization* (ACO). GBFS, yang merupakan algoritma pencarian heuristik untuk merepresentasikan strategi respons cepat. Kecepatan ini sangat relevan dalam skenario darurat, meskipun

berkonsekuensi pada kualitas solusi yang tidak dijamin optimal.

Sebagai pembandingnya, dipilih ACO, yang mewakili paradigma metaheuristik. Keputusan untuk membandingkan dua algoritma dari keluarga yang berbeda ini diambil secara sadar untuk mendapatkan wawasan yang lebih mendasar. Sementara komparasi antar algoritma sejenis akan terbatas pada analisis efisiensi dalam satu metode, perbandingan antara GBFS dan ACO menguji perbandingan antara pendekatan deterministik melawan probabilistik. Dalam konteks bencana banjir, kemampuan ACO untuk menemukan rute alternatif melalui eksplorasi kolektif yang adaptif menjadi sebuah hipotesis sentral yang membedakannya dari algoritma heuristik standar.

Meskipun berbagai penelitian telah mengkaji penerapan algoritma pencarian jalur berbasis heuristik maupun metaheuristik pada permasalahan optimasi rute, sebagian besar studi masih berfokus pada penggunaan satu pendekatan algoritma secara terpisah. Penelitian berbasis heuristik umumnya menitikberatkan pada efisiensi waktu komputasi dan respons cepat dalam penentuan jalur, seperti pada penerapan metode Greedy dalam optimasi distribusi bantuan (Harahap and Triase 2024). Sementara itu, penelitian berbasis metaheuristik banyak menyoroti kemampuan eksplorasi solusi dan optimalitas rute melalui pendekatan probabilistik, seperti penggunaan *Ant Colony Optimization* (ACO) untuk pencarian jalur terpendek pada konteks layanan darurat dan transportasi (Alfa Husna et al. 2023).

Namun demikian, kajian komparatif yang secara langsung membandingkan algoritma heuristik deterministik seperti *Greedy Best-First Search* (GBFS) dengan algoritma metaheuristik probabilistik seperti ACO dalam satu skenario simulasi evakuasi bencana yang sama masih relatif terbatas, khususnya pada konteks simulasi berbasis model graf di wilayah Indonesia. Padahal, karakteristik bencana banjir yang dinamis menuntut keseimbangan antara kecepatan pengambilan keputusan dan kemampuan adaptasi terhadap perubahan kondisi jaringan jalan. Oleh karena itu, penelitian ini berupaya mengisi kesenjangan tersebut melalui analisis komparatif kuantitatif antara GBFS dan ACO dalam skenario simulasi evakuasi berbasis graf.

Oleh karena itu, penelitian ini bertujuan untuk mengisi celah tersebut. Menggunakan metode pengembangan sistem *prototyping*, sebuah aplikasi simulasi akan dibangun sebagai alat uji utama. Untuk menjawab pertanyaan penelitian, kinerja kedua algoritma akan diukur dan dibandingkan secara sistematis berdasarkan tiga parameter utama: (1) Kualitas Solusi, yang dinilai dari total jarak rute yang dihasilkan (semakin pendek semakin baik); (2) Efisiensi Waktu, yang dinilai dari waktu komputasi yang dibutuhkan untuk menemukan solusi (semakin cepat semakin baik); dan (3) Ketangguhan, yang dinilai dari kemampuan algoritma menemukan rute

alternatif yang efisien saat dihadapkan pada skenario jalur terputus. Hasil dari penelitian ini diharapkan dapat memberikan rekomendasi berbasis bukti mengenai strategi algoritmik yang paling sesuai untuk sistem pendukung keputusan evakuasi bencana.

2. METODE PENELITIAN

2.1. Greedy Best First Search (GBFS)

Algoritma *greedy* adalah grup prosedur pemecahan masalah yang selalu mengambil penyelesaian sementara yang terbaik pada setiap langkahnya untuk menyelesaikan suatu masalah (Purnamasari, et. al. 2019). Pilihan terbaik diambil disetiap langkah tanpa perlu memikirkan bagaimana pengaruhnya terhadap penyelesaian secara keseluruhan (Goenawan, Faqih, and Pulungan 2023).

GBFS memiliki sebuah fungsi evaluasi $f(n)$. Nilai fungsi evaluasi GBFS bergantung pada nilai fungsi herusitik $h(n)$ itu sendiri. $h(n)$ akan memperoleh estimasi arah yang benar, sehingga pencarian jalur terpendek dapat sangat singkat. Secara sistematis fungsi evaluasi pada GBFS dapat ditulis sebagai berikut :

$$f(n) = h'(n) \quad (1)$$

Dengan :

$f(n)$: fungsi evaluasi

$h'(n)$: estimasi biaya untuk sampai pada suatu tujuan mulai dari n .

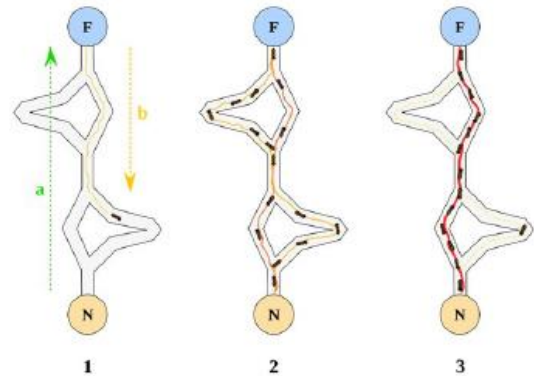
Pendekatan heuristic seperti *Greedy Best First Search* (GBFS) telah banyak digunakan dalam studi *path planning* dan optimasi rute karena kemampuannya dalam menghasilkan solusi lebih cepat dibandingkan algoritma pencarian optimal sehingga menjadikannya relevan pada kondisi yang menuntut respon waktu yang nyata (Raghavan 2022).

2.2. Ant Colony Optimization (ACO)

Ant Colony Optimization (ACO) adalah salah satu metode optimisasi yang mengadopsi sifat-sifat koloni semut dalam menemukan sumber makanan. ACO juga termasuk kedalam metode pencariin jalur terpendek yang tergolong dalam algoritma meraheuristik, yang artinya ACO menggunakan mekanisme pencarian yang terdiri dari berberapa tahapan dan bukan menggunakan algoritma pencarian yang pasti derministik (Ihsan, Adlie, and Harliansyah 2024).

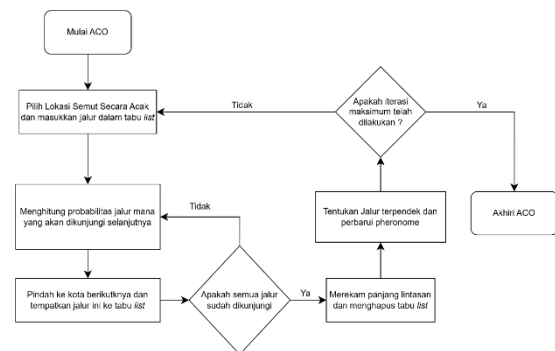
Gambar 1. memperlihatkan ilustrasi dari algoritma ACO. dari ilustrasi ini terlihat bagaimana semut meninggalkan jejak feromon saat menentukan jalur terpendek menuju sumber makanan. Peristiwa inilah yang diadaptasi menjadi model komputasi untuk optimalisasi rute evakuasi sehingga dapat membuka jalan bagi implementasi optimasi jalur pencarian dalam sistem pendukung keputusan

operasional untuk pencarian dan penyelamatan (Morin, et. al. 2022).



Gambar 1. ACO (Ihsan, Adlie, and Harliansyah 2024).

Selain ilustrasi konsep dasar, algoritma ACO juga memiliki alur yang sistematis. Proses ini melibatkan sejumlah langkah mulai dari inialisasi parameter hingga pembaruan nilai feronom. Gambar 2. menunjukkan diagram alur ACO secara menyeluruh.



Gambar 2. Diagram alur ACO

Pada Gambar 2. algoritma ACO dimulai dengan inialisasi sistem, di mana sejumlah semut ditempatkan secara acak pada berbagai lokasi. Setiap semut dilengkapi dengan tabu *list*, yaitu daftar yang mencatat jalur yang telah dilalui untuk mencegah pengulangan kunjungan ke lokasi yang sama setelah inialisasi, semut mulai bergerak dengan memilih jalur berikutnya berdasarkan probabilitas yang dihitung dari dua factor utama. Ketika semut tiba di titik berikutnya, jalur yang dilalui segera ditambahkan ke tabu *list* (Purwokartiko 2022). Proses ini berlanjut hingga semua titik tujuan berhasil dikunjungi. Setelah itu, panjang total lintasan setiap semut dihitung untuk menentukan rute terpendek pada iterasi tersebut. Selanjutnya adalah perbaruan feronom. Siklus ini diulang hingga mencapai iterasi maksimum atau solusi yang dihasilkan sudah stabil.

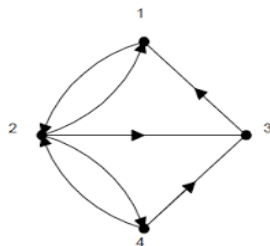
Tahapan eksplorasi rute waktu para semut menentukan tujuan titik berikutnya. Dimana semut tersebut yang berada di lokasi akan menuju lokasi dengan persamaan rumus sebagai berikut :

$$p_{ij}^k = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum [\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta} \text{ jika } \epsilon \text{ diperoleh } k \quad (2)$$

Dalam rumus 2, $\tau(i,j)$ merepresentasikan jumlah feromon pada jalur i ke j pada waktu t , yang mencerminkan pengalaman koloni semut dalam mengeksplorasi jalur tersebut. Sementara itu, $\eta(i,j)$ adalah nilai visibilitas heuristik yang biasanya dihitung sebagai kebalikan dari jarak antara node i dan j , memberikan informasi tentang kualitas jalur berdasarkan pengetahuan awal. Parameter α dan β mengontrol pengaruh relatif antara feromon (α) dan visibilitas heuristik (β) dalam pengambilan keputusan. Penyebut dalam rumus ini menjumlahkan produk feromon dan visibilitas untuk semua jalur yang mungkin, sehingga probabilitas yang dihasilkan ternormalisasi antara 0 dan 1. Dengan demikian, rumus ini menggabungkan eksploitasi pengalaman koloni (feromon) dan eksplorasi pengetahuan lokal (heuristik) untuk memandu semut dalam menemukan solusi optimal (Ihsan, Adlie, and Harliansyah 2024).

2.3. Graf

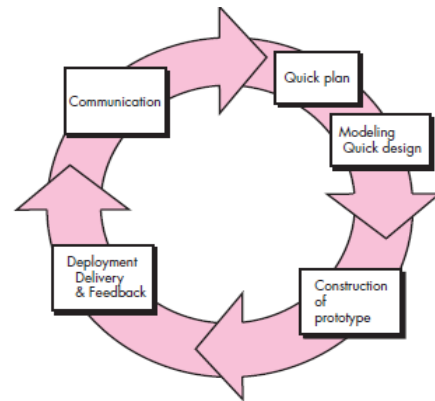
Graf adalah struktur diskrip yang terdiri dari adanya simpul (*vertex*) dan adanya sebuah sisi (*edge*), graf adalah pasangan himpunan (V,E) dimana V merupakan sebuah himpunan yang tidak kosong dari sebuah *vertex* dan E adalah himpunan sisi yang menghubungkan sepasang simpul dalam graf tersebut (Pratama, Sahay, and Nugrahaningsih 2025). Secara sederhana graf dapat diartikan sebagai sekumpulan objek terstruktur dimana beberapa pasang objek mempunyai keterkaitan (Buhaerah, Busrah, and Sanjaya 2022).



Gambar 3. Model Graf (Buhaerah, Busrah & Sanjaya, 2022)

2.4. Prototyping

Prototyping adalah suatu pendekatan dimana sebuah prototipe (perkiraan awal dari sistem atau produk akhir) dibangun, diujin dan kemudian diperbaiki secara berulang hingga diperoleh prototipe yang dapat diterima (Pressman 2021). Metode *prototyping* juga merupakan sebuah proses membangun model yang menunjukkan fitur-fitur dari sistem yang diusulkan pada pengguna (Kendall 2021). Metode *Prototyping* memiliki tahapan yang tampak pada Gambar 4.



Gambar 4. Tahapan *Prototyping* (Pressman, 2021)

Communication: pada proses ini dilakukan komunikasi dengan pihak-pihak yang berkepentingan. Tujuan utamanya adalah untuk menggali informasi terkait kebutuhan simulasi, kondisi geografis, titik rawan bencana, dan titik akhir jalur evakuasi. Informasi yang diperoleh menjadi dasar untuk menyusun fungsi utama dari prototipe.

Quick Plan: dilakukan perencanaan cepat terkait fitur yang akan dikembangkan dalam simulasi. Perencanaan ini mencakup identifikasi algoritma pencarian jalur evakuasi yang akan digunakan, struktur graf sebagai representasi wilayah, serta data titik awal dan akhir evakuasi bencana alam. Fokus utamanya adalah menyusun gambaran awal pengembangan simulasi dalam ruang lingkup terbatas namun fungsional.

Modeling Quick Design: dilakukan desain awal dari simulasi, termasuk representasi graf berbobot yang merepresentasikan desa-desa dalam satu wilayah, peta jalur evakuasi, serta rancangan antarmuka visual. Tahapan ini juga mencakup bagaimana algoritma pencarian jalur evakuasi akan diterapkan dan divisualisasikan dalam bentuk simulasi interaktif.

Construction of Prototype: dibuat berdasarkan tahap modeling quick design yang telah dibuat. Pengembangan mulai mengimplementasikan algoritma yang dipilih dalam bahasa pemrograman Python, serta membangun tampilan graf dan antarmuka pengguna menggunakan *library* pendukung seperti *Streamlit*.

Deployment, Delivery & Feedback: tahap ini adalah tahap terakhir yang mana dilakukan evaluasi dari simulasi jalur evakuasi bencana alam. Umpan balik dari pengguna digunakan untuk menyempurnakan tampilan graf, keakuratan hasil pencarian jalur evakuasi bencana alam, serta perbandingan performa algoritma.

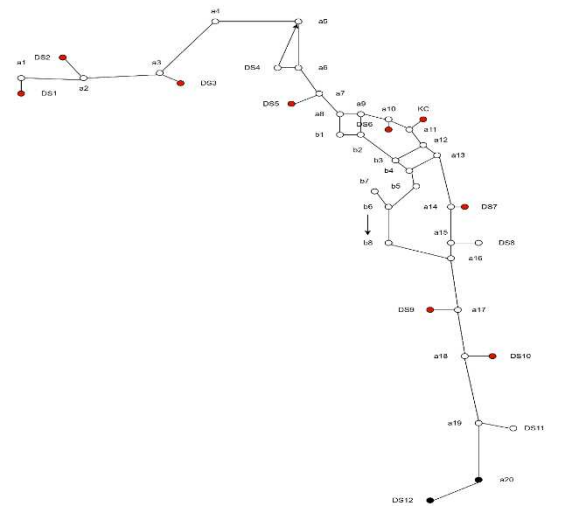
3. PERANCANGAN SISTEM

Dalam fase perancangan sistem, dikembangkan tiga komponen fundamental sebagai berikut : pertama model graf visual yang merepresentasikan jaringan evakuasi; kedua, struktur data untuk menyimpan

informasi graf; dan ketiga, rancangan antarmuka pengguna sebagai media interaksi. Ketiga komponen ini dirancang secara integrasi untuk memenuhi kebutuhan simulasi.

3.1. Model Graf Visual

Perancangan Model graf visual ini dibuat berdasarkan peta kecamatan cikajang, sebuah model graf jaringan dirancang secara visual menggunakan perangkat lunak draw.io, seperti yang ditunjukkan pada Gambar 5.



Gambar 5. Model graf wilayah kecamatan Cikajang

Proses pemodelan pada Gambar 5 melibatkan penempatan setiap titik persimpangan, titik awal evakuasi, dan titik tujuan akhir sebagai *node* pada kanvas digital. Selanjutnya. Setiap ruas jalan yang menghubungkan antar *node* digambarkan sebagai *edge*. Model visual ini menjadi sumber utama untuk mengekstraksi data spasial.

3.2. Struktur Data

Model graf yang telah dirancang tidak dapat langsung digunakan oleh program. Oleh karena itu, data tersebut perlu di transformasikan ke dalam format terstruktur yang dapat dibaca oleh sistem. Dalam penelitian ini, format CSV dipilih karena kesederhanaannya. Sampel dari struktur data ini disajikan pada Tabel 1. dan Tabel 2.

Tabel 1. Nodes.CSV

id	x	y	type
DS1	50	260	awal
a1	50	230	simpang
a2	140	230	simpang
DS2	110	190	awal
a3	250	220	simpang
DS3	280	240	awal

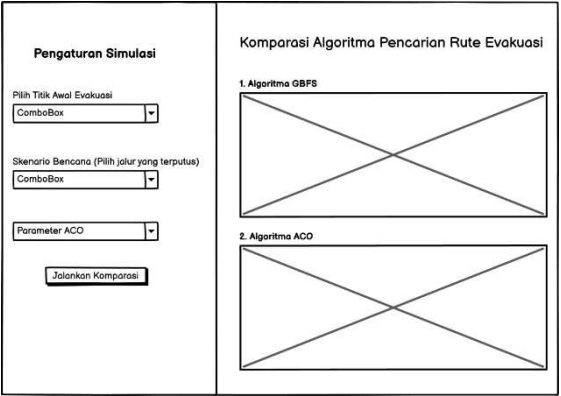
Tabel 2. Edges.CSV

sumber	target
DS1	a1
a1	a2
a2	DS2
a2	a3
a3	DS3
a3	a4

Tabel 2. secara rinci mendefinisikan setiap titik (*node*) yang ada dalam model graf. Kolom id berfungsi sebagai pengenalan unik untuk setiap titik, di mana awalan DS merepresentasikan titik awal evakuasi (Desa), awalan a dan b merepresentasikan titik persimpangan jalan, dan KC adalah titik tujuan akhir. Kolom x dan y menyimpan data koordinat posisi yang diekstraksi dari perangkat lunak draw.io untuk memastikan akurasi tata letak visual dalam simulasi. Terakhir, kolom tipe mengkategorikan fungsi dari setiap *node*, yaitu sebagai titik awal, tujuan, atau simpang, yang penting untuk logika simulasi. Tabel 2.2 melengkapi ini dengan mendefinisikan konektivitas jaringan, di mana setiap barisan mencatat satu jalur yang menghubungkan *node* sumber dan target.

3.3. Rancangan Antarmuka Pengguna (UI)

Untuk memandu pengembangan aplikasi, sebuah *wireframe* antarmuka pengguna dirancang. *Wireframe* ini dapat dilihat pada Gambar 6, berfungsi sebagai cetak biru yang memetakan tata letak komponen interaktif dan area visualisasi.



Gambar 6. Wireframe antarmuka aplikasi

Desain ini secara strategis membagi antarmuka menjadi dua panel utama sebuah sidebar di sisi kiri untuk menempatkan semua kontrol simulasi (pemilihan titik awal, definisi jalur terputus, dan pengaturan parameter ACO), dan sebuah area konten utama di sisi kanan untuk menampilkan hasil secara berdampingan. Pendekatan ini bertujuan untuk menciptakan alur kerja yang efisien bagi peneliti, memungkinkan skenario dan perbandingan hasil dapat dilakukan dengan cepat.

4. HASIL DAN PEMBAHASAN

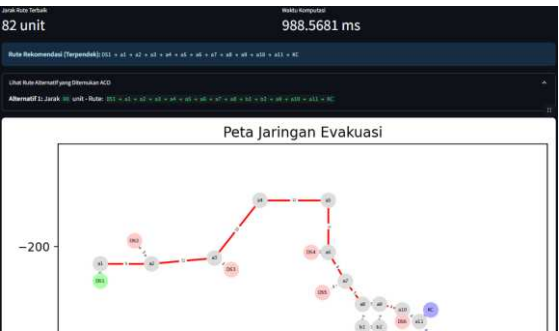
4.1. Hasil Simulasi

Pada pengujian ini, simulasi dijalankan pada titik awal DS1. Algoritma *Greedy Best First Search* berhasil menemukan satu jalur dengan total jarak 82 unit dalam waktu 0.0531 ms. Jalur yang direkomendasikan adalah DS1 → a1 → a2 → a3 → a4 → a5 → a6 → a7 → a8 → a9 → a10 → a11 → KC. Visualisasi rute ditunjukkan pada Gambar 7.



Gambar 7. Simulasi 1 GBFS

Gambar 7. menampilkan hasil visual dari algoritma GBFS. Rute yang direkomendasikan, ditandai dengan garis merah, menunjukkan jalur yang ditemukan dari titik awal DS1 (ditandai hijau) menuju titik tujuan yaitu KC. Terlihat bahwa algoritma secara efisien memilih jalur utama tanpa melakukan deviasi yang tidak perlu. Strategi GBFS menunjukkan kinerja yang konsisten dari waktu ke waktu, yang ditunjukkan sebagai titik-titik di sekitar garis konstan. Hal ini tidak mengejutkan karena dengan strategi ini, agen selalu memilih untuk mencapai korban terdekat terlebih dahulu, yang tidak bergantung pada faktor lain dalam lingkungan penyelamatan .



Gambar 8. Simulasi 1 ACO

Gambar 8. menampilkan rute terbaik yang ditemukan oleh algoritma ACO. seperti halnya GBFS, rute optimal yang ditemukan juga memiliki jalur utama yang sama. Namun, perlu dicatat bahwa solusi ini dicapai melalui proses eksplorasi oleh banyak semut, yang juga berhasil mengidentifikasi beberapa rute alternatif yang valid. Sebuah kapabilitas yang menjadi pembeda utama dengan pendekatan deterministik GBFS.

4.2. Rangkuman Hasil Pengujian

Tabel 3. Hasil Simulasi

Titik Awal	Algoritma	Bobot	Waktu Komputasi	Rute alternatif
DS1	GBFS	82	0.0531 ms	0
	ACO	82	988.57 ms	1
DS4	GBFS	28	0.0591 ms	0
	ACO	28	427.1196 ms	3
DS8	GBFS	31	0.0996 ms	0
	ACO	31	519.9029 ms	3
DS12	GBFS	81	0.1047 ms	0
	ACO	Tidak ditemukan	-	-

Data yang disajikan pada Tabel 3. merangkum temuan dari serangkaian pengujian yang telah dilakukan. Dari tabel tersebut, dapat ditarik beberapa pengamatan awal yang signifikan. Pertama, dari segi kualitas solusi, kedua algoritma secara konsisten menemukan jalur dengan jarak yang identik pada 3 percobaan. Kedua, terdapat perbedaan performa yang sangat ekstrem pada parameter waktu komputasi, dimana *Greedy Best First Search* (GBFS) menunjukkan kecepatan yang jauh lebih superior. Ketiga, hanya algoritma ACO yang mampu menghasilkan jalur-jalur alternatif. Terakhir, yang paling menonjol adalah kegagalan ACO untuk menemukan jalur dari titik awal DS12, sementara GBFS berhasil.

Secara statistik, GBFS menunjukkan dominasi yang signifikan pada aspek waktu komputasi. Rata-rata waktu komputasi GBFS pada seluruh titik awal adalah 0,079 ms, sedangkan ACO mencapai 645,53 ms, yang berarti ACO membutuhkan waktu sekitar 8.170 kali lebih lama dibandingkan GBFS. Nilai rentang (range) waktu komputasi GBFS sangat kecil (0,0531–0,1047 ms), menunjukkan stabilitas kinerja yang tinggi, sementara ACO memiliki rentang waktu yang jauh lebih besar (427,12–988,57 ms), yang mengindikasikan variabilitas komputasi yang tinggi.

Dari sisi bobot rute, kedua algoritma menghasilkan nilai identik pada 75% skenario uji (DS1, DS4, dan DS8), menunjukkan bahwa tidak terdapat perbedaan signifikan dalam kualitas solusi ketika solusi berhasil ditemukan. Namun, pada DS12, ACO gagal menemukan solusi, sehingga secara statistik GBFS memiliki tingkat keberhasilan 100%, sedangkan ACO hanya 75%, yang memperkuat keunggulan GBFS dalam hal reliabilitas algoritmik.

Analisis rute alternatif menunjukkan bahwa ACO menghasilkan rata-rata 2,33 rute alternatif pada skenario yang berhasil, sedangkan GBFS selalu menghasilkan nol rute alternatif. Hal ini menegaskan perbedaan karakteristik algoritma: GBFS bersifat deterministik dengan satu solusi optimal, sementara ACO bersifat probabilistik dan eksploratif.

4.3. Analisis Kualitas Solusi

Merujuk pada Tabel 3., dapat diamati sebuah temuan yang signifikan. Untuk titik awal DS1, DS4, dan DS8, algoritma mampu menghasilkan rute dengan kualitas yang setara dengan rute optimal yang ditemukan oleh ACO. Hal ini mengindikasikan bahwa fungsi heuristic yang diterapkan pada GBFS terbukti sangat efektif untuk memandu pencarian pada struktur graf ini, sehingga mampu mencapai tujuan secara efisien.

4.4. Analisa Efisiensi Waktu

Pada Tabel 3., terlihat perbedaan performa yang sangat signifikan. *Greedy Best First Search* (GBFS) secara konsisten menyelesaikan pencarian dalam waktu dibawah 1 milidetik, yang menjadikannya hampir instan. Sebaliknya, ACO membutuhkan waktu komputasi yang lebih lama, berkisaran antara 427 ms hingga 988 ms. Perbedaan masif ini menegaskan karakteristik fundamental kedua algoritma yaitu *Greedy Best First Search* (GBFS) sebagai algoritma *single-pass* yang terarah memberikan solusi secara langsung, sementara *Ant Colony Optimization* sebagai algoritma iteratif membutuhkan proses komputasi yang sangat intensif untuk memungkinkan para semut menjelajahi graf.

4.5. Analisis Kemampuan Eksplorasi

Pada percobaan simulasi, hasil dari titik awal DS12 menyoroti perbedaan mendasar dalam hal keandalan dan eksplorasi. *Greedy Best First Search* (GBFS) terbukti sangat andal dengan selalu berhasil menemukan satu jalur tunggal tanpa menemukan jalur alternatif. Sebaliknya, *Ant Colony Optimization* (ACO) menunjukkan masalah keandalan pada kasus DS12 dimana ia gagal menemukan jalur. Kegagalan ini terjadi karena pada graf yang kompleks, semut yang memulai penjelajahan secara acak memiliki probabilitas untuk tersesat dan tidak pernah mencapai tujuan dalam iterasi yang ditemukan. Namun, ketika berhasil, keunggulan untuk *Ant Colony Optimization* (ACO) adalah kemampuannya mengidentifikasi beberapa jalur alternatif yang mana hal ini tidak dimiliki oleh *Greedy Best first Search* (GBFS).

4.6. Implikasi dan Trade-off

Hasil analisis secara komprehensif menyajikan sebuah *trade-off* fundamental antara dua algoritma yang berbeda. Disatu sisi, *Greedy Best First Search* (GBFS) menunjukkan dirinya sebagai algoritma yang unggul dalam efisiensi dan keandalan taktis. Dengan waktu komputasi yang hampir instan dan kemampuan untuk secara konsisten menghasilkan jalur berkualitas tinggi pada studi kasus ini, GBFS merupakan kandidat ideal untuk respon darurat dimana kecepatan pengambilan keputusan menjadi prioritas tertinggi. Kemampuannya untuk memberikan

satu rekomendasi jalur yang andal secara cepat sangat berharga dalam situasi krisis yang membutuhkan kepastian. Di sisi lain, *Ant Colony Optimization* (ACO), meskipun secara signifikan lebih lambat dan menunjukkan tantangan keandalan pada parameter standar, menawarkan keunggulan dalam domain yang berbeda yaitu fleksibilitas dan perencanaan yang strategis. ACO mampu mengeksplorasi berbagai solusi potensial memungkinkan identifikasi beberapa jalur alternatif yang valid. Kapabilitas ini tidak dapat direplikasi oleh GBFS yang bersifat deterministik. Dalam konteks pra-bencana, ketersediaan beberapa opsi jalur ini sangat berharga bagi pihak berwenang untuk merancang rencana jalur evaluasi. Dengan demikian, pilihan antara kedua algoritma ini bukanlah tentang superioritas absolut, melainkan tentang kesesuaian strategis terhadap kebutuhan spesifik dalam siklus manajemen bencana.

5. KESIMPULAN

Berdasarkan hasil pengujian pada beberapa titik awal, penelitian ini menunjukkan bahwa tidak terdapat satu algoritma yang unggul secara mutlak dalam seluruh aspek pemilihan rute evakuasi. *Greedy Best-First Search* (GBFS) terbukti unggul dalam efisiensi waktu komputasi dan keandalan dalam menemukan solusi, bahkan pada kasus tertentu seperti DS12 ketika *Ant Colony Optimization* (ACO) gagal menemukan rute. GBFS secara konsisten menghasilkan bobot rute minimum tanpa variasi rute alternatif, sehingga sangat sesuai untuk sistem yang membutuhkan respons cepat, deterministik, dan andal, seperti aplikasi respon darurat real-time. Sebaliknya, ACO memiliki kemampuan eksploratif yang lebih baik, yang ditunjukkan dengan ditemukannya beberapa rute alternatif pada kasus DS4 dan DS8. Namun, keunggulan tersebut diperoleh dengan konsekuensi waktu komputasi yang jauh lebih besar serta potensi kegagalan solusi ketika menggunakan parameter standar. Oleh karena itu, pemilihan algoritma sebaiknya disesuaikan dengan kebutuhan dan konteks aplikasi, di mana GBFS lebih tepat untuk skenario tanggap darurat, sedangkan ACO lebih sesuai untuk perencanaan strategis pra-bencana yang membutuhkan variasi solusi dan toleransi terhadap biaya komputasi yang lebih tinggi.

DAFTAR PUSTAKA

- ALFA HUSNA, NUR, DESVITA HENDRI, HILMI ZALNEL HAQ, & AKHAS RAHMADEYAN. 2023. "Implementasi Algoritma *Ant Colony Optimization* Untuk Penentuan Jalur Terpendek Klinik Dari Lokasi Rawan Kecelakaan Di Kota Pekanbaru." *SENTIMAS: Seminar Nasional Penelitian Dan Pengabdian Masyarakat*, 112–19.
- BAYU, SEPTIAN, & AJI PAMUNGKAS. 2023. "Seminar Nasional Teknologi Informasi Dan Komunikasi-2023 "Exploring the Intersection

- of Big Data, Cyber Security, Rancang Bangun Sistem Manajemen Bencana Alam Berbasis Website Pada BPBD Kabupaten Madiun,” 213–22.
- BPBD. 2025. “Data Kebencanaan Kabupaten Garut.” BPBD Garut. 2025. <https://sikat.bpbd.garutkab.go.id/>.
- BUHAERAH, ZULFIQAR BUSRAH, & HERLAN SANJAYA. 2022. *Teori Graf Dan Aplikasinya. Living Spiritual Quotient*.
- DONG, K., YANG, D., SHENG, J., ZHANG, W. & JING, P., 2024. Dynamic planning method of evacuation route in dam-break flood scenario based on the ACO-GA hybrid algorithm. *International Journal of Disaster Risk Reduction*, 100, p.104219. <https://doi.org/10.1016/j.ijdr.2023.104219>
- GOENAWAN, AAQILA DHIYAANISAF, ABDURRAHMAN FAQIH, & MUTIARA PERSADA PULUNGAN. 2023. “Penerapan Algoritme Greedy Best First Search (Gbf) Terhadap Sistem Kasir.” *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)* 2 (1): 10–17. <http://senafiti.budiluhur.ac.id/index.php/senafiti/article/view/572>.
- HARAHAP, SUKMA, & TRIASE. 2024. “Algoritma Greedy Untuk Mendukung Keputusan Pemilihan Rute Distribusi Bantuan Tercepat Pasca Banjir” 13:1689–1704.
- IHSAN, AHMAD, TAUFAN ARIF ADLIE, & Septia Harliansyah. 2024. “Optimalisasi Pencarian Jalur Terpendek Mobile Robot Dengan Menggunakan Metode *Ant Colony Optimization* (ACO).” *Techné : Jurnal Ilmiah Elektroteknika* 23 (1): 39–54. <https://doi.org/10.31358/techn.v23i1.389>.
- KENDALL, KENNETH E. 2021. *Systems Analysis And Design Eight Edition. Sustainability (Switzerland)*. Eighth Edi. Vol. 11. New York: Prentice Hall.
- MASE, LINDUNG ZALBUIN, YULIAN FAUZI, & ANISSA FITRIA EDRIANI. N.D. “Evaluation of The Road Vulnerability Network During the Evacuation Process (A Case Study in A Coastal Area of Bengkulu City , Indonesia)” 27 (10): 81–91. <https://doi.org/10.4186/ej.2023.27.10.81>.
- MORIN, M., ABI-ZEID, I. AND QUIMPER, C.-G., 2023. Ant colony optimization for path planning in search and rescue operations. *European Journal of Operational Research*, 305(1), pp.53–63. <https://doi.org/10.1016/j.ejor.2022.06.019>
- PRATAMA, FERDHA ALIF, ABERTUN SAGIT SAHAY, & NAHUMI NUGRAHANINGSIH. 2025. “Perbandingan Algoritma Dijkstra Dan A-Star Dalam Pencarian Rute Terpendek Sekolah Menengah Atas (SMA) Di Kota Palangka Raya Berbasis Website” 5:1–3.
- PRESSMAN, ROGER S. 2021. *Software Engineering A Practitioner's Approach*. <https://doi.org/10.1145/336512.336521>.
- PURWOKARTIKO, BUDI SANTOSO. 2022. “No Title.” Youtube. 2022. <https://www.youtube.com/watch?v=KMsftukh5jg>.
- RAGHAVAN, SUBRAMANIAN. 2022. “Investigation of Heuristic Search Techniques for Path Planning” 11 (05): 353–56.
- RIVALDY, MOHAMMAD, SOBAR SUTISNA, ROBERTUS ANUGERAH, PURWOKO PUTRO, & ADI SUBIYANTO. 2025. “Engineering , Environment , and Technology Vol 10 No 3 2025 Planning of Evacuation Places and Routes for Flood Disaster in Kesambi District , Cirebon , Indonesia” 10 (3): 369–75. <https://doi.org/10.25299/jgeet.2025.10.3.22858>.
- SUTEDI, A., JULIANTO, I.T. & FITRIANI, L., 2024. Segmentasi Wilayah Terdampak Bencana Berdasarkan Fitur Geo-Posisi. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 11(4), pp.797–804. DOI: 10.25126/jtiik.1148557.
- YANG, Z., NGUYEN, L., ZHU, J., PAN, Z., LI, J. & JIN, F., 2020. Coordinating disaster emergency response with heuristic reinforcement learning. In: *Proceedings of the 2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. The Hague, Netherlands, 7–10 December 2020. pp.565–572. <https://doi.org/10.1109/ASONAM49781.2020.9381416>
- YULITA, FITRI, KURNIABUDI, & NURHADI. 2023. “Analisis Dan Perancangan Sistem Informasi Geografis Pemetaan Daerah Rawan Bencana Banjir Pada Dinas Sosial Kota Jambi.” *Jurnal Informatika Dan Rekayasa Komputer (JAKAKOM)* 1 (2): 2808–5469. <https://doi.org/10.33998/jakakom.v4i2>.
- YUNUS. 2025. “Peran Badan Penanggulangan Bencana Daerah (BPBD) Dalam Penanggulangan Pasca Bencana Banjir Di Desa Dampal Kecamatan Sirenja Kabupaten Donggala The Role of the Regional Disaster Management Agency (BPBD) in Post-Flood Disaster Management in Dampal Villa.” *Peran Badan Penanggulangan Bencana Daerah (BPBD) Dalam Penanggulangan Pasca Bencana Banjir Di Desa Dampal Kecamatan Sirenja Kabupaten Donggala* 8 (5): 2319–27. <https://doi.org/10.56338/jks.v8i5.7524>.