

BAB II

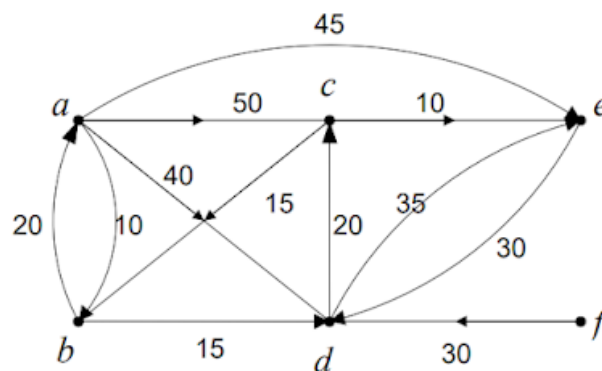
TINJUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Teori Graf

Teori graf pertama kali diperkenalkan oleh *Leonhard Euler* dengan memodelkan masalah Jembatan Königsberg pada tahun 1735 (Marcus, 2020). Konsep-konsep seperti graf Eulerian dan graf Hamiltonian dikembangkan seiring waktu oleh berbagai ilmuwan. Teori graf membahas struktur matematika yang dikenal sebagai graf, didefinisikan sebagai pasangan $G = (V, E)$, di mana V merupakan himpunan berhingga yang berisi titik-titik atau simpul (*vertices*) yang tidak kosong, dan E adalah himpunan sisi-sisi (*edges*) yang menghubungkan titik-titik tersebut (Majeed and Rauf, 2020). Setiap sisi $e \in E$ dalam graf G adalah pasangan tak berurutan dari titik-titik di V , dengan kata lain, sebuah sisi menghubungkan sepasang titik $\{u, v\}$, dimana $u, v \in V$.

Jumlah titik dalam suatu graf disebut *order* graf, dinyatakan dengan $|V|$, yang menunjukkan seberapa banyak simpul yang ada dalam graf tersebut. Sementara itu, jumlah sisi dalam graf disebut *size* graf, yang dilambangkan dengan $|E|$, yaitu total banyaknya koneksi antara titik-titik. Agar lebih jelas dapat dilihat pada Gambar 2.1 dibawah ini.



Gambar 2. 1 Representasi Graf

Sumber: Gunawan & Andriani, 2021

Pada Gambar 2.1 Graf G merupakan contoh graf yang memuat himpunan titik $|V|(G) = 6$ dan himpunan sisi $|E|(G) = 13$ sebagai berikut.

$$V(G) = (a, b, c, d, e, f)$$

$$E(G) = ((a, b), (a, c), (a, e), (a, d), (b, a), (b, d), (c, b), (c, e), (d, c), (d, e), (e, d), (f, d)).$$

Pada Graf G sisi $((a, b), (b, a))$ dinamakan sisi-ganda atau ruas sejajar (*parallel edges*) karena kedua sisi menghubungkan dua buah titik yang sama yaitu $V(a)$ dan $V(b)$. Setelah memahami definisi graf, perlu diketahui istilah penting lainnya yang sering digunakan dalam teori graf. Berikut adalah beberapa terminologi dasar yang perlu dipahami dalam penelitian ini :

a. Bertetangga

Dua simpul dikatakan bertetangga jika keduanya dihubungkan oleh sebuah sisi langsung. Pada suatu graf dengan jumlah simpul sebanyak n , matriks kedekatannya akan berbentuk matriks persegi. Setiap elemen dalam matriks ini menunjukkan hubungan antar simpul dalam graf. Jika dua simpul saling terhubung, maka elemen pada matriks tersebut bernilai 1. Sebaliknya, jika tidak ada hubungan antara kedua simpul tersebut, elemen pada matriks bernilai 0.

Pada graf berbobot, ketetanggaan antara dua simpul mencakup bobot sisi yang menunjukkan besaran seperti jarak atau biaya hubungan tersebut. Berikut dalam Tabel 2.1 ini adalah contoh tabel matriks kedekatan untuk graf Gambar 2.1.

Tabel 2.1 Matriks Ketetanggaan dari Contoh Graf

	a	b	c	d	e	f
a	0	20	50	40	45	0
b	20	0	10	15	0	0
c	50	10	0	15	10	0
d	40	15	15	0	35	30
e	45	0	10	35	0	30
f	0	0	0	30	30	0

Sumber: diolah oleh peneliti

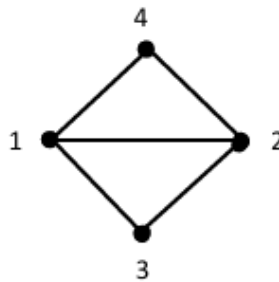
b. Jalur (*path*)

Jalur dalam graf adalah urutan yang menghubungkan titik awal V_0 ke titik tujuan V_n melalui simpul dan sisi yang berselang-seling. Jalur ini ditulis sebagai $P = v_0, e_1, v_1, e_2, v_2, \dots, v_n$, di mana setiap sisi e_i menghubungkan simpul-simpul berturutan. Kita dapat bergerak dari satu simpul ke simpul lainnya untuk mencapai tujuan yang diinginkan dengan menggunakan alur ini.

Terdapat berbagai jenis graf dalam Teori graf yang memiliki karakteristik tertentu. Menurut Marcus (2020), berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, graf dapat digolong menjadi dua jenis :

a. Graf Sederhana (*Simple Graph*).

Graf ini tidak memiliki gelang atau sisi ganda. Sisi pada graf ini dianggap sebagai pasangan tak-terurut, yang berarti bahwa sisi (u, v) sama dengan (v, u) . Agar lebih jelas, graf sederhana digambarkan oleh Gambar 2.2 dibawah ini.



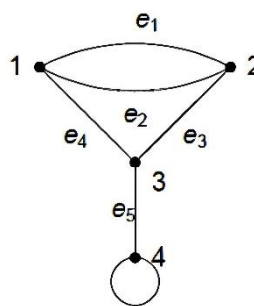
Gambar 2. 2 Contoh Graf Sederhana

Sumber : diolah oleh peneliti

b. Graf Tidak Sederhana (*Unsimple Graph*).

Graf yang memiliki sisi-ganda atau ruas sejajar (*parallel edges*).

Untuk lebih jelasnya seperti Gambar 2.3 berikut.



Gambar 2. 3 Contoh Graf Tidak Sederhana

Sumber : diolah oleh peneliti

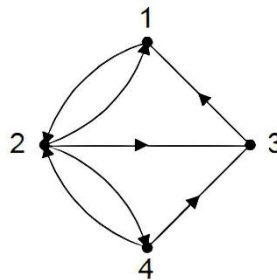
Graf yang memiliki sisi-ganda dibagi lagi menjadi dua kategori, yaitu graf ganda (*multigraph*) dan graf semu (*pseudograph*). Graf G_2 merupakan graf ganda adalah graf yang memiliki sisi ganda, yaitu sisi yang menghubungkan dua simpul yang jumlahnya dapat mencapai dua buah. Sementara itu Graf semu dicontohkan oleh graf G_3 yaitu graf yang mengandung gelang (*loop*). Selain itu, berdasarkan orientasi arah pada sisi, graf terbagi menjadi dua jenis :

a. Graf tak berarah (*Undirect Graph*).

Graf yang semua sisinya tidak memiliki arah tertentu, sehingga pasangan titik yang terhubung tidak berurutan.

b. Graf berarah (*Directed Graph*).

Setiap sisi memiliki arah, yang ditulis sebagai pasangan berurutan (u, v) , menunjukkan adanya arah dari u ke v . Untuk lebih jelasnya, contoh graf berarah ada di Gambar 2.4 berikut.



Gambar 2. 4 Contoh Graf Berarah

Sumber : diolah oleh peneliti

Teori graf diterapkan dalam kehidupan sehari-hari untuk memodelkan hubungan dan optimasi jalur, seperti dalam jaringan komputer, transportasi, manajemen proyek, rekomendasi produk, hingga penentuan rute evakuasi dan navigasi GPS. Teori graf menjadi landasan dalam masalah rute terpendek yaitu mengembangkan algoritma yang dapat menemukan jalur dengan total bobot minimum antara dua simpul. Berbagai algoritma telah dikembangkan untuk menyelesaikan masalah ini, seperti algoritma Dijkstra, algoritma Bellman-Ford, serta algoritma berbasis heuristik seperti ABC yang digunakan dalam penelitian ini untuk menentukan jalur evakuasi.

2.1.2 Rute Evakuasi Terpendek

Rute Terpendek atau *Shortest Path* (SP) dalam matematika berfokus pada masalah optimisasi jaringan yang bertujuan menemukan jalur paling efisien antara dua simpul dengan meminimalkan bobot yang digunakan, seperti biaya, waktu, atau jarak. SP bertujuan untuk mencari rute terpendek evakuasi yang mengarahkan individu dari titik rawan menuju titik aman yang telah ditentukan, sehingga mengurangi waktu tempuh dan meningkatkan keselamatan pengungsi. Masalah SP sering dimodelkan dengan graf, di mana simpul merepresentasikan lokasi dan sisi menunjukkan rute dengan bobot tertentu (Jovanović et al., 2024).

2.1.3 Algoritma Bee Colony

Algoritma Bee Colony (ABC) merupakan bagian dari metode metaheuristik yang terinspirasi oleh perilaku koloni lebah dalam mencari sumber makanan dan telah terbukti efektif dalam mencari solusi optimal dalam berbagai masalah optimisasi. ABC dibagi menjadi tiga fase utama yaitu, *employee bee*, *onlooker bee* dan *scout bee* (Niyomubyeyi et al., 2019).

Fase pertama, para lebah memulai dengan inialisasi populasi secara acak, yaitu mencari dan menyeleksi rute untuk menemukan solusi terbaik. Setelah itu setiap rute akan dievaluasi berdasarkan nilai *fitness* (atau kualitas sumber makanan) dari seluruh populasi. Informasi posisi sumber makanan ini kemudian diberikan kepada para *employee bee*. Fase *employee bee* seluruh lebah menyimpan informasi mengenai setiap sumber makanan, seperti jarak dan arah dari sarang, tingkat profitabilitas dan parameter kecocokan lainnya. *Employee bee* juga menyebar dan mengulangi proses iterasi untuk menemukan sumber makanan baru sehingga jumlah *employee bee* harus sesuai dengan jumlah sumber makanan yang ada. Mereka kemudian membandingkan nilai *fitness* dari sumber makanan baru tersebut dengan nilai sebelumnya dan meneruskan informasi kepada *onlooker bee* dengan tarian (*waggle dance*). Setelah menerima informasi dari *waggle dance*, *onlooker bee* menghitung probabilitas dari nilai *fitness* dari lama waktu *waggle dance* dan mencari sumber makanan baru, lalu membandingkan kembali hasilnya, inilah fase kedua dari ABC (Ebrahimnejada et al., 2021).

Fase terakhir yaitu *scout bee* mencari sumber makanan ke segala arah untuk menghindari lebah terjebak di solusi lokal. Nilai *fitness* tertinggi akan dipilih sedangkan sumber makanan yang lain ditinggalkan. Iterasi ini akan diulang jika belum menyentuh batas iterasi hingga menemukan solusi terbaik. Pada konteks penelitian optimasi rute, lokasi sumber makanan direpresentasikan sebagai kandidat solusi dari masalah optimasi, sedangkan jumlah nektar merepresentasikan nilai *fitness* yang

menunjukkan bobot solusi tersebut (Ebrahimnejada et al., 2021). Langkah-langkah utama proses ABC dapat diuraikan sebagai berikut (Ebrahimnejada et al., 2016) :

a. Tahap Inisialisasi

Pada tahap ini, dilakukan *input* parameter dan dijalankan pencarian solusi awal dan hitung nilai *fitness* dari solusi awal yang terbentuk dengan persamaan berikut.

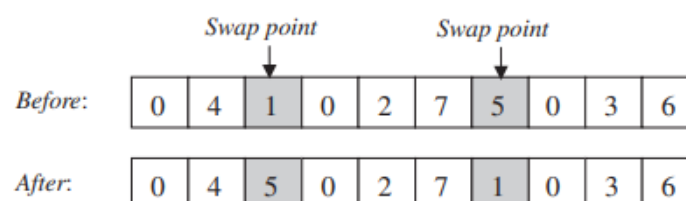
$$fit_i = \begin{cases} \frac{1}{1+fit_i} & ; f \geq 0 \\ 1 + |f| & ; f < 0 \end{cases} \quad (2.1)$$

dimana:

- fit_i = Nilai *fitness* dari solusi i yang sebanding dengan nektar jumlah sumber makanan diposisi i
- f_i = Jumlah bobot tiap sumber makanan

b. Tahap *Employee Bee*

Setiap *employed bee* melakukan pencarian lokal di sekitar solusi dengan menggunakan operator mutasi. Jika jalur baru yang ditemukan lebih baik, solusi diperbarui; jika tidak, solusi lama dipertahankan. Untuk peningkatan solusi, digunakan *Neighborhood Operator* berupa *Swap Operator* dan *Swap Sequence*, yang memilih posisi (dalam matriks) misalkan n dan m , dengan $n \neq m$ dan menukar dua posisi tersebut untuk eksplorasi lebih lanjut (Kaveh et al., 2020). Untuk lebih jelasnya dapat dilihat di Gambar 2.5 berikut. Misalkan $n = 2$ dan $m = 6$.



Gambar 2. 5 Proses Swap Operator

Sumber: Szeto et al., 2011.

c. Tahap *Onlooker Bee*

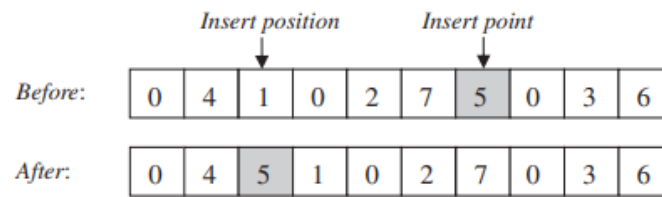
Pada tahap ini *onlooker bee* melakukan perhitungan probabilitas terkait dengan nilai *fitness* yang didapat dari tahapan sebelumnya untuk memilih solusi terbaik. Probabilitas dari setiap pilihan (P_i) diberikan sebagai berikut (Karaboga,2011).

$$P_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \quad (2.2)$$

dimana:

- P_i = nilai probabilitas
- fit_i = nilai *fitness* Solusi ke-i
- $\sum_{n=1}^{SN} fit_n$ = jumlah dari nilai *fitness* ke-i sampai SN (jumlah jalur populasi yang sama dengan employed bee)

Setelah itu dilakukan optimasi solusi kembali dengan melakukan eksplorasi dengan *Insert Operator* berdasarkan seleksi *roulette wheel*. *Roulette wheel* adalah metode seleksi acak yang memilih individu terbaik dari populasi saat ini dan diposisikan pada populasi yang baru. Semakin baik kualitas individu maka akan semakin besar peluang individu tersebut akan terpilih untuk menjadi anggota populasi yang baru. Nilai acak pada seleksi *roulette wheel* bernilai antara $[0,1]$. Jika nilai random lebih besar sama dengan nilai probabilitas dan lebih kecil dari nilai probabilitas maka tukar nilai populasi. Proses *Insert Operator* dalam ABC melibatkan pemilihan dua indeks secara acak; indeks kedua dipindahkan ke posisi indeks pertama, sementara sisa nilai indeks diurutkan. Proses ini dilakukan sebanyak jumlah populasi, dan rangkaian *Insert Operator* disebut *Insert Sequence* (Kaveh et al., 2020). Agar lebih jelas perhatikan Gambar 2.6 berikut.



Gambar 2. 6 Proses *Insert Operation*

Sumber: Szeto et al., 2011.

d. Tahap *Scout Bee*

Apabila solusi tidak mengalami perbaikan hingga *limit* iterasi yang telah ditentukan, maka solusi tersebut dihilangkan dan *Scout Bee* akan mencari sumber makanan baru secara random. Pada fase scout bee nilai kumulatif (q_i) probabilitas dibandingkan dengan bilangan acak antara 0 sampai dengan 1. Sebelum dilakukan perbandingan terlebih dahulu dihitung nilai kumulatif probabilitas dengan persamaan berikut.

$$q_i = \sum_{k=1}^i p_k$$

(2.3)

dimana:

- p_k = probabilitas untuk memilih solusi tertentu
- q_i = nilai kumulatif dari probabilitas tersebut

e. Evaluasi

Setelah *Scout Bee* telah melakukan tugasnya maka akan dilakukan kembali dijalankan iterasi hingga memenuhi batas kriteria berhenti yaitu iterasi maksimum (*limit*). Sumber makanan yang terbaik akan dipilih dan disimpan sebagai solusi terakhir (Zong et al., 2022).

2.2 Hasil Penelitian Terdahulu

Penelitian-penelitian sebelumnya yang membahas penerapan ABC dan penentuan rute terpendek yang menjadi acuan dalam skripsi ini disajikan dalam tabel berikut.

Tabel 2. 2 Hasil Penelitian Terdahulu

No	Penulis (Tahun)	Judul Penelitian	Metode	Hasil
1	Mhd. Furqan, Yusuf Ramadhan Nasution, dan Khairunnisa (2022)	<i>Application of Artificial Bee Colony Algorithm to Optimize The Shortest Route to Distribute Clean Water Pipes</i>	Model optimasi rute distribusi pipa air bersih menggunakan ABC.	Penelitian ini menunjukkan bahwa ABC mampu menentukan rute distribusi pipa air bersih secara efisien, menghasilkan rute optimal $V1 \rightarrow V7 \rightarrow V4 \rightarrow V9 \rightarrow V8 \rightarrow V2 \rightarrow V6 \rightarrow V3 \rightarrow V5$, yang meningkatkan efisiensi dalam pelayanan distribusi air oleh PDAM Tirtanadi.
2	Hilda Chaerunnissa, Respitawulan (2021)	Penyelesaian TSP dengan Menggunakan ABC	ABC untuk menyelesaikan masalah TSP	Keterbaruan dari penelitian ini terletak pada penerapan ABC untuk menyelesaikan masalah TSP dalam konteks pengiriman paket. Penelitian ini menghasilkan rute optimal untuk pengiriman paket yang menghubungkan 18 lokasi dengan total jarak 95,9 km menggunakan ABC.
4	Fico D, Deden	<i>Path Planning Using the ABC</i>	Menggunakan ABC untuk merencanakan	Penelitian ini mampu memberikan aplikasi inovatif dalam penerapan ABC untuk pemantauan pertanian, ditemukan

	W,Dedy S, Ahmad L, Hanif F, M.Izman H (2023)	<i>to Find Routes for a Coffee Plantation Using Drones</i>	jalur terbaik bagi drone dalam memantau kematangan biji kopi.	jalur tercepat untuk dua area perkebunan kopi. Hasil penelitian menunjukkan bahwa jalur tercepat untuk area 1 dapat dilalui dalam waktu 18,4 menit, dan untuk area 2 dalam 13,4 menit, dengan kecepatan drone 30 km/jam.
5	Ni Wayan Windayani, Cokorda Rai Adi Pramatha (2022)	Implementasi Algoritma <i>Artificial Bee Colony</i> dalam Penentuan Rute Terpendek Perusahaan Ekspedisi	Model rute terpendek menggunakan algoritma ABC untuk menyelesaikan masalah TSP dalam konteks pengiriman barang.	Penelitian ini menunjukkan bahwa meskipun pengujian parameter seperti jumlah lebah, limit, dan maksimum iterasi tidak menghasilkan perubahan signifikan pada jarak terpendek, ABC berhasil menemukan rute terpendek dengan jarak 38,42 km, yang dapat meningkatkan efisiensi dalam pengiriman barang oleh perusahaan ekspedisi.

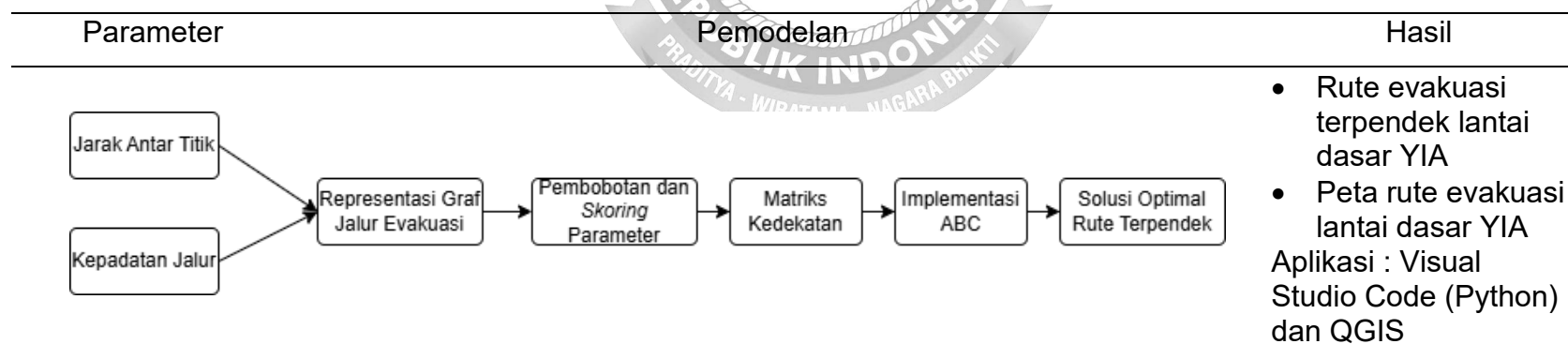
Sumber: diolah oleh peneliti

Meskipun seluruh penelitian terdahulu pada Tabel 2.2 tersebut menunjukkan efektivitas ABC dalam berbagai konteks optimasi jalur, terdapat sejumlah keterbatasan yang menjadi celah penelitian ini. Mayoritas penelitian terdahulu cenderung hanya mempertimbangkan satu titik awal dan satu titik tujuan akhir dalam proses perhitungan, tanpa mengakomodasi kompleksitas situasi multi-titik yang justru lazim dijumpai dalam skenario evakuasi darurat. Selain itu, dasar pembobotan graf yang digunakan umumnya hanya mengandalkan parameter jarak, tanpa memasukkan variabel penting lainnya seperti kepadatan jalur. Kepadatan sangat memengaruhi efektivitas dan kecepatan perpindahan manusia pada konteks evakuasi massal, sehingga pengabaian terhadap faktor ini dapat mengurangi relevansi model terhadap kondisi nyata.

Menanggapi kekosongan tersebut, penelitian ini mengembangkan pendekatan baru melalui pemodelan graf berbobot yang mengintegrasikan dua parameter utama, yaitu jarak dan kepadatan. Pemodelan ini diterapkan dalam skenario multi-titik evakuasi pada kawasan strategis YIA, yang memiliki karakteristik spasial kompleks dan dinamika kepadatan manusia yang tinggi. Untuk meningkatkan kualitas solusi yang dihasilkan oleh algoritma, studi ini juga menerapkan modifikasi pada fase pencarian lokal ABC melalui penyisipan mekanisme mutasi lokal berupa operasi *swap* dan *insert*, yang bertujuan untuk memperluas ruang pencarian solusi dan mencegah terjebaknya algoritma pada titik lokal optimum.

2.3 Kerangka Berpikir

Tabel 2. 3 Kerangka Berpikir Penelitian



Sumber: diolah oleh peneliti

Tabel 2.3 merupakan hal-hal yang dilakukan dalam pelaksanaan penelitian. Penelitian ini diawali dengan pengumpulan data dan observasi di YIA, yaitu peta *layout* lantai dasar dalam format SHP, dari peta tersebut diolah untuk mendapat data jarak antar titik, dan kepadatan jalur di asumsikan untuk membangun model graf yang merepresentasikan

jalur evakuasi. Selanjutnya dilakukan pembobotan dengan memberikan skor dan bobot pada parameter jarak dan kepadatan. ABC kemudian diterapkan untuk mengevaluasi jalur, melalui tiga fase utama—*employed bee*, *onlooker bee*, dan *scout bee*—hingga ditemukan jalur terpendek. Akhirnya, jalur evakuasi optimal dianalisis untuk menilai efektivitasnya dalam mendukung mitigasi risiko bencana dan perencanaan evakuasi yang lebih baik di YIA.

