

## IMPLEMENTASI PERHITUNGAN JARAK TERPENDEK DENGAN MENGGUNAKAN ALGORITMA ANT COLONY OPTIMIZATION (ACO) BERBASIS ANDROID

Juanda Hakim Lubis

juandahakim@gmail.com  
Universitas Medan Area

### ABSTRAK

Penggunaan jalur terpendek pada suatu perjalanan merupakan hal yang penting, tidak hanya dalam dunia pengiriman barang tetapi juga untuk pencarian suatu lokasi. Salah satunya adalah pencarian lokasi wisata di provinsi Jawa Barat. Secara umum, pencarian jalur terpendek dapat dibagi menjadi dua metode yaitu metode konvensional dan heuristik. Metode heuristik terdiri dari beberapa macam algoritma yang biasa digunakan, salah satunya adalah algoritma Ant colony optimization (ACO). Algoritma Ant colony optimization (ACO) cukup efektif dalam penentuan jalur terpendek, karena hasil perhitungan yang didapatkan cukup akurat berdasarkan perhitungan matematis. Selain jumlah kota, nilai parameter juga cukup mempengaruhi hasil dari perhitungan. Hasil yang dapat diperoleh adalah memberikan rute terpendek lokasi-lokasi wisata di provinsi Jawa Barat dan informasi jalur padat menggunakan algoritma Ant colony optimization (ACO).

Kata kunci : Pencarian jalur terpendek, Heuristik, Ant colony optimization (ACO)

### 1. PENDAHULUAN

Perkembangan moda transportasi darat saat ini semakin pesat, baik itu transportasi umum maupun transportasi pribadi. Hampir semua orang melakukan perjalanan menggunakan moda transportasi darat dari satu tempat ke tempat lain setiap hari. Tentu saja perjalanan yang dilakukan tidak tanpa pertimbangan terlebih dahulu. Pertimbangan yang dilakukan tentu berdasarkan beberapa faktor seperti : biaya, waktu, dan efisiensi. Oleh karena itu sangat perlu sekali untuk memilih jalur terpendek dari suatu tempat ke tempat lain untuk dilalui.

Jawa Barat merupakan salah satu provinsi yang banyak dikunjungi wisatawan. Jawa Barat juga memiliki banyak lokasi wisata yang menarik minat wisatawan untuk berkunjung. Misalnya Tangkuban Perahu dan Dusun Bambu di Bandung, Kebun Raya di Bogor, pantai Pangandaran, dan masih banyak lagi yang lainnya. Wisatawan yang berkunjung tidak hanya masyarakat Jawa Barat, tetapi banyak juga yang berasal dari provinsi lain bahkan dari luar pulau

Jawa maupun dari luar negeri. Hal yang menjadi permasalahan adalah banyak wisatawan yang belum mengenal secara keseluruhan lokasi wisata di provinsi Jawa Barat atau baru pertama kali kesana, sehingga mereka belum mengetahui rute-rute untuk menuju ke tempat wisata tersebut. Semakin banyak wisatawan yang berkunjung ke lokasi wisata-wisata yang ada di Jawa Barat, maka jalanan juga menjadi semakin padat dan macet terutama untuk hari libur dan akhir pekan. Sehingga Wisatawan tentunya menginginkan rute terpendek yang dapat ditempuh dan informasi jalur-jalur yang biasanya padat agar dapat mempersingkat waktu perjalanan dan menghindari kemacetan jalan.

Pencarian jalur terpendek dapat menggunakan metode heuristik. Algoritma *Ant colony optimization* (ACO) atau algoritma semut merupakan salah satu dari algoritma metode heuristik. Algoritma semut diambil dari perilaku koloni semut dalam pencarian jalur terpendek antara sarang dan sumber makanan (Hardjasutanto, 2010)

Algoritma semut adalah algoritma yang diadopsi dari perilaku koloni semut. Secara alamiah koloni semut mampu menemukan rute terpendek dalam perjalanan dari sarang ke tempat-tempat sumber makanan. Koloni semut dapat menemukan rute terpendek antara sarang dan sumber makanan berdasarkan jejak kaki pada lintasan yang telah dilewati. Semakin banyak semut yang melewati suatu lintasan, maka akan semakin jelas bekas jejak kakinya. Algoritma Semut sangat tepat digunakan untuk diterapkan dalam penyelesaian masalah optimasi, salah satunya adalah untuk menentukan jalur terpendek, dengan menganalogikan titik awal sebagai sarang semut dan titik tujuan sebagai sumber makanan semut. Algoritma semut cukup efektif dalam penentuan jalur terpendek, karena hasil perhitungan yang didapatkan cukup akurat. Selain jumlah kota, nilai parameter juga cukup mempengaruhi akurasi hasil perhitungan (Yuwono, 2009).

## 2. LANDASAN TEORI

### a. Algoritma Ant Colony Optimization (ACO)

Pada tahun 1996, dunia ilmu pengetahuan pun ikut belajar dari semut dengan diperkenalkannya algoritma semut, atau *Ant colony optimization*(ACO), sebagai sebuah simulasi multi agen yang menggunakan metafora alami semut untuk menyelesaikan *problem* ruang fisik. Algoritma semut diperkenalkan oleh **Moyson** dan **Manderick** dan secara meluas dikembangkan oleh **Marco Dorigo**, merupakan teknik probabilistik untuk menyelesaikan masalah komputasi dengan menemukan jalur terbaik. Algoritma ini diambil dengan analogi oleh perilaku semut dalam menemukan jalur dari koloninya menuju makanan (Mindaputra, 2009)

Semut mampu mengindra lingkungannya yang kompleks untuk mencari makanan dan kemudian kembali ke sarangnya dengan meninggalkan zat *pheromone* pada jalur-jalur yang mereka lalui. *Pheromone* adalah zat kimia yang berasal dari kelenjar endokrin dan digunakan oleh makhluk hidup untuk mengenali sesama jenis, individu lain, kelompok, dan untuk membantu proses reproduksi. Berbeda dengan hormon, *pheromone* menyebar ke luar tubuh dapat mempengaruhi dan dikenali oleh individu lain yang sejenis (satu spesies). Proses

peninggalan *pheromone* ini dikenal sebagai *stigmergy*, sebuah proses memodifikasi lingkungan yang tidak hanya bertujuan untuk mengingat jalan pulang ke sarang, tetapi juga memungkinkan para semut berkomunikasi dengan koloninya. Seiring waktu, bagaimanapun juga jejak *pheromone* akan menguap dan akan mengurangi kekuatan daya tariknya. Lebih lama seekor semut pulang pergi melalui jalur tersebut, lebih lama jugalah *pheromone* menguap. Agar semut mendapatkan jalur optimal, diperlukan beberapa proses (Mindaputra, 2009):

1. Pada awalnya, semut berkeliling secara acak, hingga menemukan makanan (lihat gambar 2.1).
2. Ketika menemukan makanan mereka kembali ke koloninya sambil memberikan tanda dengan jejak *pheromone*.
3. Jika semut-semut lain menemukan jalur tersebut, mereka tidak akan bepergian dengan acak lagi, melainkan akan mengikuti jejak tersebut.
4. Kembali dan menguatkannya jika pada akhirnya mereka pun menemukan makanan.
5. Seekor semut yang secara tidak sengaja menemukan jalur optimal akan menempuh jalur ini lebih cepat dari rekan-rekannya, melakukan *round-trip* lebih sering, dan dengan sendirinya meninggalkan *pheromone* lebih banyak dari jalur-jalur yang lebih lambat ditempuh.
6. *Pheromone* yang berkonsentrasi tinggi pada akhirnya akan menarik semut – semut lain untuk berpindah jalur, menuju jalur paling optimal, sedangkan jalur lainnya akan ditinggalkan.
7. Pada akhirnya semua semut yang tadinya menempuh jalur yang berbeda - beda akan beralih ke sebuah jalur tunggal yang ternyata paling optimal dari sarang menuju ke tempat makanan.



Gambar 2.1

**Perjalanan semut menuju sumber makanan[2]**

Ket :

A : Tempat awal koloni semut (sarang)

B : Tujuan koloni semut (makanan)

**b. Karakteristik *Ant colony optimization* (ACO)**

Terdapat tiga karakteristik utama dari ACO, yaitu Aturan transisi status, Aturan pembaruan *pheromone* lokal, Aturan pembaruan *pheromone* global (Mindaputra, 2009).

1. Aturan transisi status

Seekor semut yang ditempatkan pada titik  $t$  memilih untuk menuju ke titik  $v$ , kemudian diberikan bilangan pecahan acak  $q$  dimana  $0 \leq q \leq 1$ ,  $q_0$  adalah sebuah parameter yaitu Probabilitas semut melakukan eksplorasi pada setiap tahapan, dimana ( $0 \leq q_0 \leq 1$ ) dan  $p_k(t,v)$  adalah probabilitas dimana semut  $k$  memilih untuk bergerak dari titik  $t$  ke titik  $v$ .

Jika  $q \leq q_0$  maka pemilihan titik yang akan dituju menerapkan aturan yang ditunjukkan oleh persamaan (1) :

$$\text{temporary}(t,u) = [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta],$$

$$i = 1, 2, 3, \dots, n$$

$$v = \max \{ [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta] \} \quad (1)$$

dengan  $v$  = titik yang akan dituju

sedangkan jika  $q > q_0$  digunakan

persamaan (2)

$$v = p_k(t,v) = \frac{[\tau(t,v)] \cdot [\eta(t,v)^\beta]}{\sum_{i=1}^n [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta]} \quad (2)$$

$$\text{dengan } \eta(t,u) = \frac{1}{\text{jarak}(t,u_i)}$$

dimana  $\tau(t,v)$  adalah nilai dari jejak *pheromone* pada titik  $(t,u)$ ,  $\eta(t,u)$  adalah fungsi heuristik dimana dipilih sebagai invers jarak antara titik  $t$  dan  $u$ ,  $\beta$  merupakan sebuah parameter yang mempertimbangkan kepentingan relatif dari informasi heuristik yaitu besarnya bobot yang diberikan terhadap parameter informasi heuristik sehingga solusi yang dihasilkan cenderung berdasarkan nilai fungsi matematis. Nilai untuk

parameter  $\beta$  adalah  $> 0$ . *Pheromone* adalah zat kimia yang berasal dari kelenjar endokrin dan digunakan oleh makhluk hidup untuk mengenali sesama jenis, individu lain, kelompok dan untuk membantu proses reproduksi. Berbeda dengan hormon, *pheromone* menyebar keluar tubuh dapat mempengaruhi dan dikenali oleh individu lain yang sejenis (satu spesies). Seiring waktu, bagaimanapun juga jejak *pheromone* akan menguap dan akan mengurangi kekuatan daya tariknya, sehingga jejak *pheromone* harus diperbaharui. Pada ACO pembaharuan *pheromone* dibagi menjadi 2, yaitu : aturan pembaharuan *pheromone* lokal dan aturan pembaharuan *pheromone* global (Mindaputra, 2009).

2. Aturan pembaharuan *pheromone* lokal

Selagi melakukan tur untuk mencari solusi, semut mengunjungi ruas-ruas dan mengubah tingkat *pheromone* pada ruas-ruas tersebut dengan menerapkan aturan pembaruan *pheromone* lokal yang ditunjukkan oleh persamaan (3).

$$\tau(t,v) \rightarrow (1-\rho) \cdot \tau(t,v) + \rho \cdot \Delta\tau(t,v) \quad (3)$$

$$\Delta\tau(t,v) = \frac{1}{L_{nn} \cdot C}$$

Dimana :

$L_{nn}$  = panjang tur yang diperoleh

$C$  = jumlah lokasi

$\rho$  = parameter dengan nilai 0 sampai 1

$\Delta\tau$  = perubahan *pheromone*

$\rho$  adalah sebuah parameter (koefisien evaporasi), yaitu besarnya koefisien penguapan *pheromone*. Adanya penguapan *pheromone* menyebabkan tidak semua semut mengikuti jalur yang sama dengan semut sebelumnya. Hal ini memungkinkan dihasilkan solusi alternatif yang lebih banyak. Peranan dari aturan pembaruan *pheromone* lokal ini adalah untuk mengacak arah lintasan yang sedang dibangun, sehingga titik-titik yang telah dilewati sebelumnya oleh tur seekor semut mungkin akan dilewati kemudian oleh tur semut yang lain. Dengan kata lain, pengaruh dari pembaruan lokal ini adalah untuk membuat tingkat ketertarikan ruas-ruas yang ada berubah secara dinamis. Setiap kali seekor semut menggunakan sebuah ruas makaruas ini dengan segera akan berkurang tingkat ketertarikannya

(karena ruastersebut kehilangan sejumlah *pheromone*-nya), secara tidak langsung semut yang lain akan memilih ruas-ruas lain yang belum dikunjungi. Konsekuensinya, semut tidak akan memiliki kecenderungan untuk berkumpul pada jalur yang sama.

Fakta ini, yang telah diamati dengan melakukan percobaan [Dorigo dan Gambardella, 1997]. Merupakan sifat yang diharapkan bahwa jika semut membuat tur-tur yang berbeda maka akan terdapat kemungkinan yang lebih tinggi dimana salah satu dari mereka akan menemukan solusi yang lebih baik daripada mereka semua berkumpul dalam tur yang sama. Dengan cara ini, semut akan membuat penggunaan informasi *pheromone* menjadi lebih baik tanpa pembaruan lokal, semua semut akan mencari pada lingkungan yang sempit dari tur terbaik yang telah ditemukan sebelumnya (Dorigo, 999).

3. Aturan pembaruan *pheromone* global

Pembaruan *pheromone* secara global hanya dilakukan oleh semut yang membuat tur terpendek sejak permulaan percobaan. Pada akhir sebuah iterasi, setelah semua semut menyelesaikan tur mereka, sejumlah *pheromone* ditaruh pada ruas-ruas yang dilewati oleh seekor semut yang telah menemukan tur terbaik (ruas-ruas yang lain tidak diubah). Tingkat *pheromone* itu diperbarui dengan menerapkan aturan pembaruan *pheromone* global yang ditunjukkan oleh persamaan (4).

$$\tau(t,v) \rightarrow (1-\alpha) \cdot \tau(t,v) + \alpha \cdot \Delta\tau(t,v) \quad (4)$$

$$\Delta\tau(t,v) \begin{cases} L_{gb}^{-1} & \text{jika } (t,v) \in \text{tur\_terbaik} \\ 0 & \end{cases}$$

Dimana :

$\tau(t,v)$  = nilai *pheromone* akhir setelah mengalami pembaruan lokal

$L_{gb}$  = panjang jalur terpendek pada akhir siklus

$\alpha$  = parameter dengan nilai 0 sampai 1

$\Delta\tau$  = perubahan *pheromone*

$\Delta\tau(t,v)$  bernilai  $\frac{1}{L_{gb}}$  jika ruas  $(t,v)$  merupakan bagian dari rute terpendek, namun jika sebaliknya

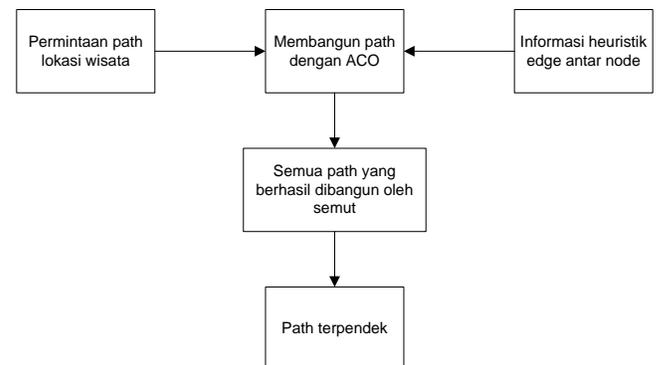
nilai  $\Delta\tau(t,v) = 0$ .  $\alpha$  adalah tingkat kepentingan relatif dari *pheromone* atau besarnya bobot yang diberikan terhadap *pheromone*, sehingga solusi yang dihasilkan cenderung mengikuti sejarah masa lalu dari semut dari perjalanan sebelumnya, dimana nilai parameter  $\alpha$  adalah  $\geq 0$ , dan  $L_{gb}$  adalah panjang dari tur terbaik secara global sejak permulaan percobaan. Pembaruan *pheromone* global dimaksudkan untuk memberikan *pheromone* yang lebih banyak pada tur-tur yang lebih pendek. Persamaan (4) menjelaskan bahwa hanya ruas-ruas yang merupakan bagian dari tur terbaik secara global yang akan menerima penambahan *pheromone*.

3. ANALISIS SISTEM

a. Analisis Metode ACO

Sistem yang dibuat adalah untuk mencari rute (*path*) terpendek berdasarkan kriteria yaitu jarak dan kepadatan. Cara kerja dari sistem ini secara umum adalah :

1. *User* meminta path lokasi wisata dari *node* awal sampai *node* tujuan
2. Sistem akan membangun solusi dengan ACO melalui semut buatan (*ant*) berdasarkan informasi heuristik (*visibilitas*) antar *node*
3. Semua path yang berhasil dibangun oleh semut akan dicari rute terpendek



Gambar 3.1 Sistem pencarian rute terpendek[2]

Dalam algoritma semut, diperlukan beberapa variabel dan langkah-langkah untuk menentukan jalur terpendek, yaitu:

1. Menentukan *pheromone* awal masing-masing semut. Tapi sebelum itu tentukan terlebih dahulu

banyaknya semut dalam proses tersebut, setelah itu tentukan titik awal masing-masing semut.

- Setelah itu tentukan titik selanjutnya yang akan dituju, ulangi proses sampai semua titik terlewati. Dengan menggunakan persamaan 1 atau 2 dapat ditentukan titik mana yang akan dituju, yaitu dengan :

Jika  $q \leq q_0$  maka pemilihan titik yang akan dituju menerapkan aturan yang ditunjukkan oleh persamaan (1)

$$\begin{aligned} \text{temporary}(t,u) &= [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta], \\ i &= 1,2,3,\dots,n \\ v &= \max \{[\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta]\} \end{aligned} \quad (1)$$

dengan  $v$  = titik yang akan dituju

sedangkan jika  $q > q_0$  digunakan

persamaan (2)

$$v = p_k(t,v) = \frac{[\tau(t,v)] \cdot [\eta(t,v)^\beta]}{\sum_{i=1}^n [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta]} \quad (2)$$

$$\text{dengan } \eta(t,u) = \frac{1}{\text{jarak}(t,u)}$$

Pada probabilitas  $Q_0$  semut akan memilih sisi jalan dengan nilai *pheromone* dan visibilitas yang baik (mengeksplotasi informasi yang sudah dianggap baik). Sedangkan pada probabilitas  $(1-Q_0)$  semut akan mengeksplorasi rute yang baru. Sehingga dengan mengatur parameter  $Q_0$  diharapkan dapat menyeimbangkan kecenderungan semut untuk mengambil rute yang lama (dianggap baik) dengan keinginan untuk mengeksplorasi rute alternative (Dorigo, 1999).

- Apabila telah mendapatkan titik yang dituju, *pheromone* masing-masing pada titik tersebut diubah dengan menggunakan persamaan 3, yaitu :

$$\tau(t,v) \rightarrow (1-\rho) \cdot \tau(t,v) + \rho \cdot \Delta\tau(t,v) \quad (3)$$

$$\Delta\tau(t,v) = \frac{1}{L_{nn} \cdot C}$$

dimana :

$L_{nn}$  = panjang tur yang diperoleh

$C$  = jumlah lokasi

$\rho$  = parameter dengan nilai 0 sampai 1

$\Delta\tau$  = perubahan *pheromone*

Perubahan *pheromon* tersebut dinamakan perubahan *pheromon* lokal.

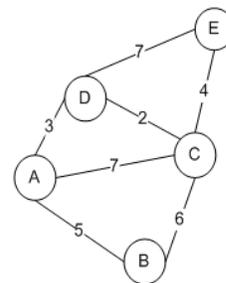
- Setelah proses diatas selesai, hitung panjang lintasan masing-masing semut.
- Kemudian akan didapatkan panjang lintasan yang minimal.
- Ubah *pheromone* pada titik-titik yang termuat dalam lintasan tersebut.
- Setelah semua proses telah dilalui, maka akan didapatkan lintasan dengan panjang lintasan yang minimal.

### b. Contoh Perhitungan ACO

Dalam pengaplikasian *software* pencarian rute terpendek yang dibuat berdasarkan algoritma *Ant colony optimization* (ACO) maka diperlukan suatu data jarak antar lokasi yang terdapat pada lokasi asal sebagai acuan utama dalam pencarian rute terpendek tersebut. Sebagai contoh perhitungan dari penerapan aplikasi yang dibuat menuju lokasi tujuan E, lokasi asal A sebagai acuan utama dalam pencarian rute terpendek, data jarak antar lokasi dapat dilihat dari tabel 3.1, serta denah lokasi dari tiap-tiap lokasi dapat dilihat pada gambar 3.2.

**Tabel 3.1 Jarak antar lokasi dalam satuan meter**

	A	B	C	D	E
A	0	5	7	3	0
B	5	0	6	0	0
C	7	6	0	2	4
D	3	0	2	0	7
E	0	0	4	7	0



**Gambar 3.2 Denah lokasi berdasarkan data jarak antar lokasi**

Terdapat tiga tahapan dalam menghitung jarak ruteterpendek dengan menggunakan algoritma *Ant colony optimization*, yaitu:

**1. Tahap pemilihan titik yang akan dituju**

Pada tahap ini seorang wisatawan yang ditempatkan pada titik A memilih untuk menuju ke titik E dengan menerapkan aturan yang ditunjukkan oleh persamaan (1) dan persamaan (2).

$$\text{temporary}(t,u) = [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta],$$

$$i = 1,2,3,\dots,n$$

$$v = \max \{[\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta]\} \quad (1)$$

sedangkan jika  $q > q_0$  digunakan persamaan (2)

$$v = p_k(t,v) = \frac{[\tau(t,v)] \cdot [\eta(t,v)^\beta]}{\sum_{i=1}^n [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta]} \quad (2)$$

$$\text{dengan } \eta(t,u) = \frac{1}{\text{jarak}(t,u_i)}$$

dimana :

- v = titik yang akan dituju
- (t,u) = titik awal dan titik tujuan
- $\tau$  = nilai *pheromone*
- $\eta$  = nilai invers
- $\beta$  = parameter dengan nilai  $> 0$

Contoh perhitungan :

Pada contoh perhitungan ini, titik awal lokasi wisatawan untuk menjalani turnya berawal dari lokasi A.

- a. Sebelum memasuki perhitungan pada tahap satu dalam perhitungan algoritma ACO maka terlebih dahulu dilakukan perhitungan awal untuk menghitung *invers* jarak  $\eta(t,u)$  antar tiap titik berdasarkan tabel 3.1 sebagai berikut :

$$\eta(t,u) = \frac{1}{\text{jarak}(t,u)}$$

Contoh perhitungan  $\eta(t,u)$  pada titik  $\eta(A,D)$  :

$$\eta(t,u) = \frac{1}{\text{jarak}(A,D)} = \frac{1}{3} = 0,333333$$

Hasil keseluruhan dari *invers* jarak  $\eta(t,u)$  dapat dilihat dari tabel 3.2 dibawah ini.

**Tabel 3.2 Invers Jarak ( $\eta(t,u)$ )**

	A	B	C	D	E
A	0	0,2	0,1429	0,3333	0
B	0,2	0	0,16667	0	0
C	0,1429	0,16667	0	0,5	0,25
D	0,3333	0	0,5	0	0,1429
E	0	0	0,25	0,1429	0

Nilai dari semua *pheromone* ( $\tau$ ) pada awal perhitungan ditetapkan dengan angka awal yang sangat kecil. Pada contoh perhitungan penelitian ini nilai *pheromone* awal menggunakan nilai awal sebesar 0,0001. Penetapan nilai *pheromone* awal dimaksudkan agar tiap-tiap ruas memiliki nilai

ketertarikan untuk dikunjungi oleh tiap-tiap semut. Nilai *pheromone* untuk semua titik dapat dilihat pada tabel 3.3 dibawah ini.

**Tabel 3.3 Pheromone( $\tau$ ) awal**

	A	B	C	D	E
A	0,0001	0,0001	0,0001	0,0001	0,0001
B	0,0001	0,0001	0,0001	0,0001	0,0001
C	0,0001	0,0001	0,0001	0,0001	0,0001
D	0,0001	0,0001	0,0001	0,0001	0,0001
E	0,0001	0,0001	0,0001	0,0001	0,0001

- b. Tahap pemilihan titik yang akan dituju  
 Dalam pemilihan titik selanjutnya yang dituju, pertama-tama dilakukan penetapan dari nilai  $\beta \geq 0$  adalah parameter perhitungan untuk mendapatkan nilai yang optimal dalam ACO, untuk mempermudah perhitungan diambil nilai  $\beta = 2$ . Selanjutnya dilakukan perhitungan untuk mendapatkan nilai *temporary* ( $t,u$ ) berdasarkan persamaan (1) serta nilai probabilitas berdasarkan persamaan (2) dari titik awal A (t) ke titik selanjutnya yang belum dilalui (u). Nilai *temporary* digunakan untuk menentukan titik-titik yang akan dituju selanjutnya. Contoh perhitungan serta hasil perhitungan nilai *temporary* dan nilai probabilitas dapat dilihat sebagai berikut :

$$\text{temporary}(t,u) = [\tau(t,u_i)] \cdot [\eta(t,u_i)^\beta],$$

$$i = 1,2,3,\dots,n$$

$$\begin{aligned} \text{temporary (A,D)} &= [\tau(A,D)].[\eta(A,D)]^\beta \\ &= [0,0001].[0,333333]^2 \\ &= 1,111E-05 \\ \text{Probabilitas (r,u)} &= \frac{[\tau(t,v)].[\eta(t,v)]^\beta}{\sum_{i=1}^n [\tau(t,u_i)].[\eta(t,u_i)]^\beta} \\ \text{Probabilitas(r,u)} &= \frac{1,1111E-05}{1,715E-05} \\ &= 0,647805394 \end{aligned}$$

Untuk memilih persamaan yang tepat sebagai acuan dalam pemilihan lokasi selanjutnya maka perlu dibangkitkan suatu bilangan *random* (q) antara 0 sampai 1 serta menetapkan suatu bilangan pembatas (q<sub>0</sub>) antara 0 sampai 1. Pada perhitungan ini ditetapkan nilai q<sub>0</sub> sebesar 0,1 serta bilangan *random* yang dibangkitkan memiliki nilai q sebesar 0,9 yang artinya semut melakukan proses eksploitasi dengan probabilitas 10% dan proses eksplorasi 90% (Bauer,n.d). Karena q > q<sub>0</sub>, maka penentuan lokasi yang akan dituju berdasarkan persamaan (2), yaitu dengan melihat hasil Probabilitas yang paling besar. Sehingga lokasi yang terpilih adalah lokasi **D**.

**2. Tahap pembaharuan pheromone(τ) lokal**

Setelah wisatawan berpindah menuju lokasi selanjutnya maka tahap selanjutnya adalah melakukan pembaharuan *pheromone* (τ) secara lokal dengan menggunakan persamaan (3). Persamaan dari pembaharuan *pheromone* (τ) lokal, contoh perhitungan serta hasil perhitungan dapat dilihat sebagai berikut :

$$\tau(t,v) \rightarrow (1-\rho) \cdot \tau(t,v) + \rho \cdot \Delta\tau(t,v) \dots\dots\dots (3)$$

$$\Delta\tau(t,v) = \frac{1}{L_{nn} \cdot c}$$

dimana :

- L<sub>nn</sub> = panjang tur yang diperoleh
- c = jumlah lokasi
- ρ = parameter dengan nilai 0 sampai 1
- Δτ = perubahan *pheromone*
- τ = nilai *pheromone*

Contoh perhitungan :

Dalam memperbaharui *pheromone* secara lokal dibutuhkan suatu parameter (ρ) yang memiliki nilai

antara 0 sampai 1. Pada perhitungan ini nilai ditetapkan dengan nilai sebesar 0,1. Contoh perhitungan serta hasil perhitungan dapat dilihat sebagai berikut :

$$\begin{aligned} \Delta\tau (A,D) &= \frac{1}{3.5} = 0,066667 \\ \tau (A,D) &= (1-\rho) \cdot \tau (A,D) + \rho \cdot \Delta\tau (A,D) \\ &= (1-0,1) \cdot 0,0001 + 0,1 \cdot 0,066667 \\ &= 0,006757 \end{aligned}$$

Hasil pembaharuan *pheromone* (τ) lokal untuk τ(A,D) dapat dilihat pada tabel 3.5 dibawah ini dengan tulisan yang dicetak miring. Dengan proses yang sama hasil keseluruhan dari pembaharuan *pheromone* lokal dari semua wisatawan dapat dilihat pada tabel 3.4 dibawah ini.

**Tabel 3.4 Nilai pheromone(τ) setelah mengalami pembaharuan pheromone lokal**

	A	B	C	D	E
A	0.0001	0.0001	0.0001	<b>0.006757</b>	0.0001
B	0.0001	0.0001	0.0001	0.0001	0.0001
C	0.0001	0.0001	0.0001	<b>0.01009</b>	<b>0.00509</b>
D	<b>0.006757</b>	0.0001	<b>0.01009</b>	0.0001	0.0001
E	0.0001	0.0001	<b>0.00509</b>	0.0001	0.0001

**3. Tahap pembaharuan pheromone(τ) global**

Setelah tahap 1 dan 2 telah selesai untuk mendapatkan satu rute dan setiap lokasi yang dikunjungi telah mengalami pembaharuan *pheromone* (τ) secara lokal, maka tahap selanjutnya adalah untuk membaharui *pheromone*(τ) secara global berdasarkan persamaan (4) namun hanya lokasi yang menghasilkan rute dengan jarak terpendek. Persamaan dari pembaharuan *pheromone* (τ) global, contoh perhitungan serta hasil perhitungan dapat dilihat sebagai berikut :

$$\tau(t,v) \rightarrow (1-\alpha) \cdot \tau(t,v) + \alpha \cdot \Delta\tau(t,v) (4)$$

$$\Delta\tau(t,v) \begin{cases} L_{gb}^{-1} & \text{jika } (t,v) \in \text{tur\_terbaik} \\ 0 & \end{cases}$$

Dimana :

- τ(t,v) = nilai *pheromone* akhir setelah mengalami pembaharuan lokal
- L<sub>gb</sub> = panjang jalur terpendek pada akhir siklus
- α = parameter dengan nilai 0 sampai 1
- Δτ = perubahan *pheromone*

Contoh perhitungan:

Setelah *wisatawan* pada iterasi 1 telah melewati tahap I dan tahap II, maka rute yang dihasilkan adalah **A-D-C-E**. Dari rute tersebut didapat panjang jalur sebesar 9 m dan merupakan panjang jalur terpendek pada iterasi pertama. Maka pembaharuan *pheromone*-nya adalah sebagai berikut.

$$\alpha = 0,1$$

$$L_{gb} = 9$$

Nilai *pheromone* akhir :

Untuk (t,v) bagian dari rute terpendek

$$\Delta\tau(t,v) = L_{gb}^{-1} = (9)^{-1} = 0,1111$$

Contoh perhitungan *pheromone* global (A,D) :

$$\tau(A,D) = (1-\alpha) * \tau(A,D) + \alpha * \Delta\tau$$

$$= (1-0,1) * 0,006757 + 0,1 * 0,1111$$

$$= 0,017192$$

Untuk (t,v) bukan bagian dari rute terpendek

$$\Delta\tau(t,v) = 0$$

Contoh perhitungan *pheromone* global (A,B) :

$$\tau(A,B) = (1-\alpha) * \tau(A,B) + \alpha * \Delta\tau$$

$$= (1-0,1) * (0,0001) + (0,1 * 0) = 0,0009$$

Dengan proses yang sama hasil keseluruhan dari pembaharuan *pheromone* global dari semua wisatawan dapat dilihat pada tabel 3.5

**Tabel 3.5** Pembaharuan *pheromone*( $\tau$ ) global

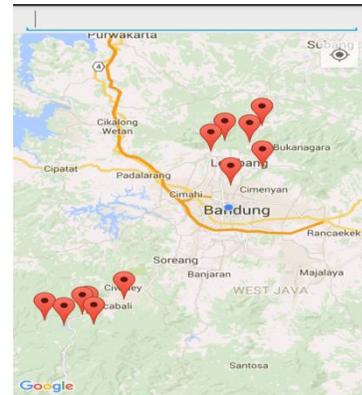
	A	B	C	D	E
A	0.0009	0.0009	0.0009	0.0172	0.0009
B	0.0009	0.0009	0.0009	0.0009	0.0009
C	0.0009	0.0009	0.0009	0.0157	0.0157
D	0.0142	0.0001	0.0202	0.0009	0.0009
E	0.0009	0.0009	0.0009	0.0009	0.0009

#### 4. IMPLEMENTASI SISTEM

##### a. Implementasi Aplikasi End User

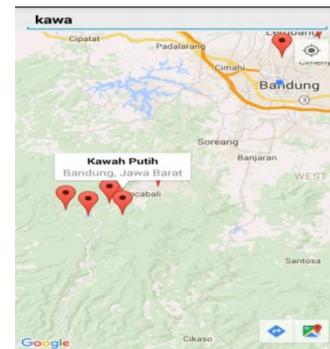
Gambar 4.1 di bawah adalah tampilan halaman awal dari sebuah aplikasi yang dibuat. Aplikasi tersebut terlebih dahulu sudah diinstall pada handphone dengan OS Android. Setelah aplikasi telah terinstall, pastikan handphone yang dipakai layanan GPS-nya telah aktif dan terhubung dengan internet. Pada aplikasi ini, user dapat mengetahui lokasi dirinya dan lokasi wisata terdekat oleh user.

Simbol titik biru merupakan lokasi user saat ini, sedangkan warna merah menunjukkan lokasi wisata terdekat dari user



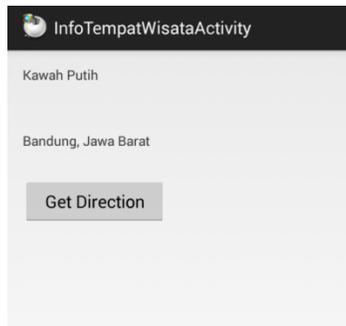
Gambar 4.1 Tampilan Awal Aplikasi

Untuk proses berikutnya, user dapat mengetikkan nama tempat wisata yang ingin dicari pada *textbox* search atau mengklik titik-titik lokasi wisata yang sudah ada. Ketika user sudah mengetikkan sebuah kata pada *textbox* search, sistem akan mencari ke database lokasi wisata yang sesuai dengan inputan user. Setelah lokasi ditemukan, sistem akan menampilkan secara detail lokasi wisata yang dicari oleh user dengan mengklik tooltip yang ditampilkan (Gambar 4.2).



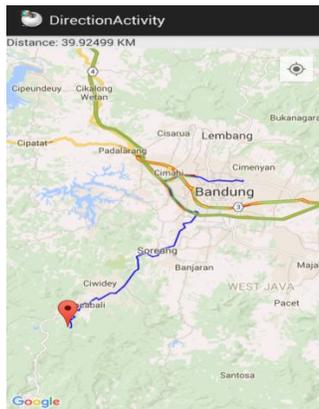
Gambar 4.2 Tampilan Search Lokasi Wisata

Apabila user sudah mengklik tooltip lokasi wisata yang diinginkan, sistem akan menampilkan detail lokasi wisata pada halaman yang lain seperti gambar 4.3. Untuk mencari jalur terpendek menuju lokasi wisata, user dapat mengklik tombol Get Direction. Setelah itu sistem akan memproses pencarian jalur terpendek.



Gambar 4.3 Tampilan Detail Lokasi Wisata

Sistem menampilkan jalur terpendek dan total jarak yang akan ditempuh menuju lokasi wisata yang dicari. Pada tampilan terlihat jalur yang berwarna biru (Gambar 4.4). Jalur berwarna biru tersebut merupakan jalur terpendek menuju lokasi wisata yang diinginkan oleh user. Selain itu sistem juga dapat menampilkan jalur-jalur padat yang diambil dari data google map. Jalur padat digambarkan dengan garis berwarna merah dan jalur lancar dengan garis berwarna hijau.

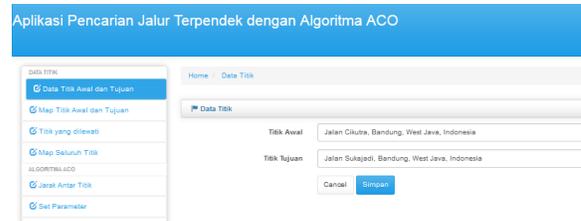


Gambar 4.4 Tampilan Jalur Terpendek Aplikasi

**b. Implementasi Aplikasi Back End**

Dalam pengaplikasian *software* pencarian rute terpendek yang dibuat berdasarkan algoritma *Ant colony optimization* (ACO) diperlukan suatu data jarak antar lokasi, dimana lokasi awal sebagai acuan utama dalam pencarian rute terpendek. Sebagai contoh perhitungan ACO dari penerapan aplikasi yang dibuat, ditentukan titik awal Jalan Cikutra Bandung dan titik tujuan Jalan Sukajadi Bandung seperti terlihat pada gambar 4.5 di bawah ini. Penentuan titik awal diperlukan untuk penempatan lokasi semut di awal. Semut akan mencari jalur

terpendek dari lokasi awal menuju ke lokasi tujuan dengan membandingkan jarak setiap titik yang dilalui.



Gambar 4.5 Tampilan penentuan titik awal dan titik tujuan

Setelah menentukan titik awal dan titik tujuan, kemudian tentukan jumlah titik yang akan dilalui dari titik awal menuju ke titik tujuan. Sebagai contoh diambil 3 titik (lihat gambar 4.6) yang dapat dilewati dari titik awal jalan Cikutra menuju ke titik tujuan jalan Sukajadi.



Gambar 4.6 Tampilan penentuan jumlah titik yang dilewati

Pengisian data 3 titik yang dapat dilalui, sebagai contoh titik pertama yang dapat dilalui yaitu jalan Pasteur Bandung, titik kedua yaitu jalan Asia Afrika Bandung dan titik ketiga yaitu jalan Cihampelas Bandung seperti ditampilkan pada gambar 4.7.



Gambar 4.7 Tampilan data titik yang dilewati

Perhitungan jarak menggunakan koordinat lokasi asal dan lokasi tujuan dengan memperoleh data dari *google maps api*, sehingga mendapatkan bobot seperti pada gambar 4.8. Bobot jarak akan disimpan

ke dalam tabel jarak. Data jarak yang diperoleh dari *google maps api* menyatakan bahwa jarak titik 1 ke titik 2 tidak sama dengan jarak titik 2 ke titik 1.

No	Titik Awal	Titik Tujuan	Jarak
1	Jalan Cikuta, Bandung, West Java, Indonesia	Jalan Cikuta, Bandung, West Java, Indonesia	0
2	Jalan Cikuta, Bandung, West Java, Indonesia	Jalan Pasteur, Bandung, West Java, Indonesia	4993
3	Jalan Cikuta, Bandung, West Java, Indonesia	Jalan Asia Afrika, Bandung, West Java, Indonesia	5940
4	Jalan Cikuta, Bandung, West Java, Indonesia	Jalan Champelas, Bandung, West Java, Indonesia	4993
5	Jalan Cikuta, Bandung, West Java, Indonesia	Jalan Sukajadi, Bandung, West Java, Indonesia	8935
6	Jalan Pasteur, Bandung, West Java, Indonesia	Jalan Cikuta, Bandung, West Java, Indonesia	4791
7	Jalan Pasteur, Bandung, West Java, Indonesia	Jalan Pasteur, Bandung, West Java, Indonesia	0
8	Jalan Pasteur, Bandung, West Java, Indonesia	Jalan Asia Afrika, Bandung, West Java, Indonesia	3991
9	Jalan Pasteur, Bandung, West Java, Indonesia	Jalan Champelas, Bandung, West Java, Indonesia	1793
10	Jalan Pasteur, Bandung, West Java, Indonesia	Jalan Sukajadi, Bandung, West Java, Indonesia	2998
11	Jalan Asia Afrika, Bandung, West Java, Indonesia	Jalan Cikuta, Bandung, West Java, Indonesia	6193
12	Jalan Asia Afrika, Bandung, West Java, Indonesia	Jalan Pasteur, Bandung, West Java, Indonesia	4949
13	Jalan Asia Afrika, Bandung, West Java, Indonesia	Jalan Asia Afrika, Bandung, West Java, Indonesia	0

Gambar 4.8 Tampilan jarak setiap titik

Dari seluruh proses perhitungan menggunakan algoritma *Ant colony optimization* yang telah dilakukan diperoleh jalur terpendek yaitu titik **1-4-2-5**. Hal ini berarti bahwa jalur terpendek dari lokasi awal jalan **Cikutra** menuju lokasi tujuan Jalan **Sukajadi** adalah melalui jalan **Cihampelas** kemudian ke jalan **Pasteur** dengan total jarak **8323** m. Jalan Asia Afrika tidak dilalui karena pada iterasi ke 3 yaitu dari titik 2 diperoleh hasil jalur selanjutnya adalah titik 5 sehingga proses iterasi berakhir karena sudah mencapai titik tujuan yaitu titik 5.

Hasil Algoritma ACO

Tabel Pembaharuan Pheromon Lokal

	Titik 1	Titik 2	Titik 3	Titik 4	Titik 5
Titik 1	0.001	0.001	0.001	0.00094087	0.001
Titik 2	0.001	0.001	0.001	0.001	0.00090952
Titik 3	0.001	0.001	0.001	0.001	0.001
Titik 4	0.001	0.00091832	0.001	0.001	0.001
Titik 5	0.001	0.001	0.001	0.001	0.001

Tabel Pembaharuan Pheromon Global

	Titik 1	Titik 2	Titik 3	Titik 4	Titik 5
Titik 1	0.0009	0.0009	0.0009	0.000819793	0.0009
Titik 2	0.0009	0.0009	0.0009	0.0009	0.00082999847411
Titik 3	0.0009	0.0009	0.0009	0.0009	0.0009
Titik 4	0.0009	0.0008294599	0.0009	0.0009	0.0009
Titik 5	0.0009	0.0009	0.0009	0.0009	0.0009

Maka Jalur yang terpilih adalah 1-4-2-5  
Dengan Total Jarak 8323

Gambar 4.9 Tampilan hasil perhitungan ACO dan jalur yang terpendek

## 5. KESIMPULAN

Berdasarkan dari keseluruhan pembahasan dalam implementasi ini, maka dapat ditarik kesimpulan sebagai berikut.

- Penerapan algoritma *Ant colony optimization* (ACO) untuk proses pencarian jalur terpendek memiliki beberapa tahapan yaitu inialisasi parameter, titik awal dan titik tujuan, menghitung jarak setiap titik, menghitung visibilitas dan temporary untuk menentukan titik selanjutnya, menghitung perubahan *pheromone* lokal, menghitung perubahan *pheromone* global sehingga diperoleh jalur terpendek.
- Sistem yang dibangun dapat menerapkan algoritma *Ant colony optimization* (ACO) untuk proses pencarian jalur terpendek lokasi wisata di provinsi Jawa Barat. Proses pencarian jalur terpendek diperoleh dari perhitungan matematis berdasarkan tahapan-tahapan yang ada pada algoritma *Ant colony optimization* (ACO). Jalur terpendek yang diperoleh digambarkan dalam map dengan garis berwarna biru atau abu dan ditampilkan jarak dari titik awal sampai titik tujuan. Nilai  $\beta$  berpengaruh pada pembaharuan nilai *pheromone*. Dengan nilai  $\beta > 0$  maka akan dihasilkan titik selanjutnya yang akan dituju. Tanpa adanya perubahan *pheromone*, maka semut akan cenderung mengeksplorasi rute yang sebelumnya telah ditemukan. Sehingga tidak ada hasil jalur terpendek yang diperoleh. Sistem yang dibangun juga dapat menampilkan jalur-jalur padat menuju lokasi wisata yang dicari. Data kepadatan jalur diperoleh dari google map. Jalur-jalur yang padat digambarkan dengan jalur berwarna merah, sedangkan jalur yang lancar digambarkan dengan jalur berwarna hijau.

## DAFTAR PUSTAKA

- Hardjasutanto Oktavino, Fabrian. 2010. *Penerapan Algoritma Semut untuk Pencarian Jalur Terpendek*. Bandung : Institut Teknologi Bandung.
- Mindaputra, Eka. 2009. *Penggunaan Algoritma Ant Colony System Dalam Traveling Salesman Problem (TSP) Pada PT. Eka Jaya Motor*. Semarang : Universitas

Diponegoro.

de Bruxelles, Belgium.

Yuwono, Bambang, Aribowo Sasmito, Agus., dan Wardoyo Budi, Siswanto. 2009. *Implementasi Algoritma Koloni Semut Pada Proses Pencarian Jalur Terpendek Jalan Protokol di Lokasi Yogyakarta*. Yogyakarta : UPN "Veteran" Yogyakarta.

Wardy, I. S. 2007. *Penggunaan graph dalam algoritma semut untuk melakukan optimisasi*. Program studi Teknik Informatika. Bandung : ITB.

Dorigo,Marco; DiCaro Giani; Gambardella, Luca. 1999. *Ant Algorithm for Discrete Optimization*. IRIDIA, Universite Libre

Google. (2015). Google Maps Api. [Online]. Tersedia: <https://developers.google.com/maps/documentation> [05 September 2015].

Pressman, Roger S. Ph.D.1997.*Software Engineering: A Practitioner's Approach*, McGraw-Hill

Common Laboratory.2010. *Modul Praktikum*

*Rekayasa Perangkat Lunak*. Bandung: IT Telkom

Mufti, Yusuf. 2010. *Panduan Mudah Pengembangan Google Map Android*. Bandung : Andi Publisher