

# SISTEM INFORMASI GEOGRAFIS PENCARIAN RUTE TERPENDEK KAMPUS TEKNIK DI KOTA MALANG DENGAN MENGGUNAKAN ALGORITMA ANT COLONY

Broto Poernomo T.P<sup>1)</sup>, Nuansa Okta Frydow<sup>2)</sup>

STMIK ASIA Malang

email: [papung@gmail.com](mailto:papung@gmail.com)<sup>1)</sup>, [frydowlers@yahoo.co.id](mailto:frydowlers@yahoo.co.id)<sup>2)</sup>

**Abstract:** Kampus mempunyai peranan yang sangat penting dalam kemajuan suatu daerah khususnya di kota Malang pada segi ilmu pendidikan dan kemajuan teknologi. Algoritma Semut (Ant Colony Optimization) diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut. Koloni semut dapat menemukan rute terpendek antara sarang dan sumber makanan berdasarkan jejak kaki pada lintasan yang telah dilalui. Pembuatan aplikasi dalam menentukan rute terpendek kampus Teknik Informatika di Kota Malang dengan menggunakan metode ant colony ini dapat diambil kesimpulan bahwa rute terpendek didapatkan dari tabulist yang diperoleh masing-masing semut

**Kata Kunci :** Jalur, Pencarian, Sistem,

## 1. Pendahuluan

Kampus mempunyai peranan yang sangat penting dalam kemajuan suatu daerah khususnya di kota Malang pada segi ilmu pendidikan dan kemajuan teknologi. Kampus merupakan lembaga pencetak sarjana yang mengedepankan lulusan yang memiliki daya saing dan kemandirian untuk berkompetisi di tingkat nasional dan internasional.

Teknologi SIG mengintegrasikan operasi pengolahan data berbasis *database* yang biasa digunakan saat ini, seperti pengambilan data berdasarkan kebutuhan, analisis statistik dengan menggunakan visualisasi yang khas serta berbagai keuntungan yang mampu ditawarkan analisis geografis melalui gambar-gambar petanya. SIG juga dapat memberikan penjelasan tentang suatu peristiwa, membuat peramalan kejadian, dan perencanaan strategis lainnya serta dapat membantu menganalisis permasalahan umum seperti mencari lokasi terdekat kampus teknik informatika yang ada di kota Malang dan sekitarnya.

Dalam membantu para calon sarjana dalam memilih kampus teknik informatika yang ada di kota Malang maka dibuatlah suatu aplikasi pencarian rute terdekat dengan menggunakan algoritma *ant colony*

### 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas maka dapat diambil rumusan masalah untuk membangun suatu aplikasi

pencarian rute terpendek kampus teknik informatika di kota Malang dengan menggunakan metode *ant colony*.

### 1.3 Batasan Masalah

Sistem informasi Geografis berbasis web yang mencakup beberapa hal, sebagai berikut:

1. Sistem ini menggunakan metode *ant colony* untuk pencarian rute terpendek.
2. Visualisasi peta dilengkapi dengan fasilitas *zooming* (perbesaran gambar) dan *zoomout* (memperkecil gambar).
3. Dalam penulisan laporan ini hanya membahas satu contoh perhitungan yaitu mencari rute terdekat kampus teknik informatika di kota Malang.
4. Sistem menyediakan informasi jarak antar objek dan pencarian objek pemetaan melalui form pemilihan objek.
5. Peta geografis jalan yang digunakan adalah jalan-jalan utama, tidak termasuk jalan-jalan kecil atau gang.
6. Sistem ini belum memperhatikan kondisi riil jalan di kota Malang dianggap jalan dua arah dan tidak memperhatikan aturan jalan.
7. Data yang ditampilkan adalah berupa titik atau *node* persimpangan jalan, lokasi kampus teknik informatika di kota Malang.

8. Sistem ini hanya mencakup 10 lokasi kampus teknik informatika di kota Malang.

## 2.1 Pengertian Sistem Informasi Geografis

Sistem informasi geografi merupakan gabungan dari tiga unsur pokok yaitu sistem, informasi, dan geografi. Sistem merupakan sekumpulan objek, ide, dan hubungannya dalam mencapai tujuan bersama. Sistem informasi merupakan suatu sistem antara manusia dan mesin yang terpadu untuk menyajikan informasi guna mendukung fungsi operasi, manajemen, dan pengambilan keputusan dalam organisasi. Penggunaan istilah informasi geografi mengandung pengertian informasi mengenai tempat-tempat yang terletak di permukaan bumi. Pengetahuan mengenai posisi di mana suatu objek terletak di permukaan bumi dan informasi mengenai keterangan dan posisi yang terdapat di permukaan bumi.

Menurut beberapa ahli kecerdasan buatan didefinisikan sebagai berikut. Menurut (**Demers, 1997**), SIG adalah sistem komputer yang digunakan untuk mengumpulkan, memeriksa, mengintegrasikan, dan menganalisis informasi-informasi yang berhubungan dengan permukaan bumi.

Menurut (**Esri, 1990**), SIG adalah kumpulan yang terorganisir dari perangkat keras komputer, perangkat lunak, data geografi, dan personil yang dirancang secara efisien untuk memperoleh, menyimpan, meng-update, memanipulasi, menganalisis, dan menampilkan semua bentuk informasi yang bereferensi geografi.

Menurut (**Rice, 2000**), SIG adalah sistem komputer yang digunakan untuk memasukkan, menyimpan, memeriksa, mengintegrasikan, memanipulasi, menganalisis, dan menampilkan data-data yang berhubungan dengan posisi- posisi di permukaan bumi.

Menurut (**Christman, 1997**), SIG adalah sistem yang terdiri atas perangkat keras, perangkat lunak, data, manusia (*brain ware*), organisasi dan lembaga yang digunakan untuk mengumpulkan, menyimpan, menganalisis, dan menyebarkan informasi-informasi mengenai daerah-daerah di permukaan bumi.

Menurut (**Foote, 1995**), SIG adalah sistem informasi yang dirancang untuk bekerja dengan data yang tereferensi secara spasial atau koordinat-koordinat geografi.

Menurut (**F. Purwadhi, 1994**), SIG merupakan suatu sistem yang mengorganisir

perangkat keras, perangkat lunak, data, serta dapat mendayagunakan sistem penyimpanan, pengolahan, maupun analisis data secara simultan sehingga dapat diperoleh informasi yang berkaitan dengan aspek keruangan. SIG merupakan manajemen data spasial dan nonspasial yang berbasis komputer dengan tiga karakteristik dasar yaitu mempunyai fenomena aktual (variabel data nonlokasi) yang berhubungan dengan topik permasalahan di lokasi bersangkutan, merupakan suatu kejadian di lokasi dan mempunyai dimensi waktu.

### 2.1.1 Tujuan Dan Manfaat Sistem Informasi Geografis

Seiring dengan kemajuan teknologi, SIG makin banyak digunakan dalam berbagai bidang, antara lain karena berikut ini.

1. SIG dapat digunakan sebagai alat bantu utama yang interaktif dan menarik dalam rangka peningkatan wawasan dan pengetahuan. Namun, yang paling penting adalah peningkatan penibelajaran dan pendidikan bagi usia sekolah, khususnya tentang konsep lokasi, ruang, dan unsur geografis di permukaan bumi.
2. SIG menggunakan data spasial dan data atribut secara terintegrasi sehingga sistemnya memiliki kemampuan analisis spasial dan non-spasial.
3. SIG dapat memisahkan secara tegas antara bentuk tampilan dan data-datanya. Oleh karena itu, SIG memiliki kemampuan untuk mengubah tampilan dalam berbagai bentuk.
4. SIG secara mudah dapat menghasilkan berbagai peta tematik. Peta-peta tematik tersebut merupakan turunan dari peta-peta lain yang data-datanya telah dimanipulasi.

### 2.1.2 Lingkup Utama Sistem Informasi Geografis

1. Input data digunakan untuk menginputkan data spasial dan data non-spasial. Data spasial biasanya berupa peta analog. Untuk SIG harus menggunakan peta digital sehingga peta analog tersebut harus dikonversi ke dalam bentuk peta digital dengan menggunakan alat digitizer. Selain proses digitasi dapat juga dilakukan proses

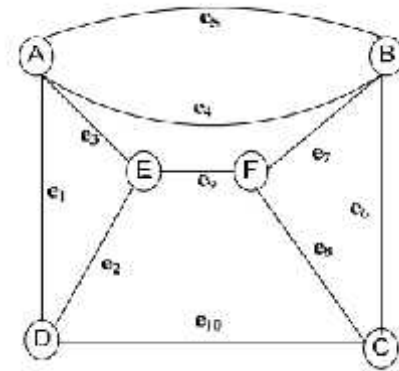
overlay dengan melakukan proses *scanning* pada peta analog.

2. Manipulasi data tipe data yang diperlukan oleh suatu bagian SIG mungkin perlu dimanipulasi agar sesuai dengan sistem yang dipergunakan. Oleh karena itu SIG mampu melakukan fungsi edit baik untuk data spasial maupun non-spasial.
3. Manajemen data setelah data spasial dimasukkan maka proses selanjutnya adalah pengolahan data non-spasial. Pengolahan data non-spasial meliputi penggunaan DBMS untuk menyimpan data yang memiliki ukuran besar.
4. *Query* dan *Analisis Query* adalah proses analisis yang dilakukan secara tabular. Secara fundamental SIG dapat melakukan dua jenis analisis, yaitu:

Analisis *Proximity* merupakan analisis geografi yang berbasis pada jarak antar layer. SIG menggunakan proses *buffering* (membangun lapisan pendukung di sekitar layer dalam jarak tertentu) untuk menentukan dekatnya hubungan antar sifat bagian yang ada. Analisis *Overlay* merupakan proses penyatuan data dari lapisan layer yang berbeda. Secara sederhana *overlay* disebut sebagai operasi visual yang membutuhkan lebih dari satu layer untuk digabungkan secara fisik.

## 2.2 Teori Graph

Suatu *graph* G terdiri atas himpunan yang tidak kosong dari elemen yang disebut titik (verteks), dan suatu daftar pasangan verteks yang tidak terurut disebut sisi (*edge*). Himpunan verteks dari suatu *graph* G dinotasikan dengan V, dan daftar himpunan *edge* dari *graph* tersebut dinotasikan dengan E. Untuk selanjutnya suatu *graph* G dapat dinotasikan dengan  $G = (V, E)$ .



**Gambar 1** Graph G

Gambar 2.1, menunjukkan graph G dengan  $V = \{A, B, C, D, E, F\}$  dan  $E = \{e_1, e_2, e_3, \dots, e_{10}\}$ .

## 2.3 Ant Colony Optimization (ACO)

*Ant Colony Optimization* (ACO) diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut (Dorigo, et.al, 1996). Semut mampu mengindera lingkungannya yang kompleks untuk mencari makanan dan kemudian kembali ke sarangnya dengan meninggalkan zat *Pheromone* pada rute-rute 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 dan hanya dapat mempengaruhi dan dikenali oleh individu lain yang sejenis (satu *spesies*).

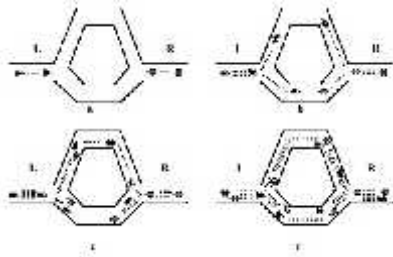
Proses peninggalan *Pheromone* ini dikenal sebagai *stigmergy*, yaitu sebuah proses memodifikasi lingkungan yang tidak hanya bertujuan untuk mengingat jalan pulang ke sarang, tetapi juga memungkinkan para semut berkomunikasi dengan koloninya.

Seiring waktu, bagaimanapun juga jejak *Pheromone* akan menguap dan akan mengurangi kekuatan daya tariknya. Lebih cepat setiap semut pulang pergi melalui rute tersebut, maka *Pheromone* yang menguap lebih sedikit. Begitu pula sebaliknya jika semut lebih lama pulang pergi melalui rute tersebut, maka *Pheromone* yang menguap lebih banyak.

### 2.4.1 Cara Kerja Semut Menemukan Rute Terpendek

Algoritma Semut (*Ant Colony Optimization*) diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut (Dorigo, 1996).

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 dilalui. Semakin banyak semut yang melalui suatu lintasan, maka akan semakin jelas bekas jejak kakinya. Hal ini akan menyebabkan lintasan yang dilalui semut dalam jumlah sedikit, semakin lama akan semakin berkurang kepadatan semut yang melewatinya, atau bahkan akan tidak dilewati sama sekali. Sebaliknya lintasan yang dilalui semut dalam jumlah banyak, semakin lama akan semakin bertambah kepadatan semut yang melewatinya, atau bahkan semua semut akan melalui lintasan tersebut. Gambar 2 menunjukkan perjalanan semut dalam menemukan jalur terpendek dari sarang ke sumber makanan.



**Gambar 2** Perjalanan semut menemukan sumber makanan

Gambar 2 di atas menunjukkan ada dua kelompok semut yang akan melakukan perjalanan. Satu kelompok bernama L yaitu kelompok yang berangkat dari arah kiri yang merupakan sarang semut dan kelompok lain yang bernama kelompok R yang berangkat dari kanan yang merupakan sumber makanan. Kedua kelompok semut dari titik berangkat sedang dalam posisi pengambilan keputusan jalan sebelah mana yang akan diambil. Kelompok semut L membagi dua kelompok lagi. Sebagian melalui jalan atas dan sebagian melalui jalan bawah.

Hal ini juga berlaku pada kelompok semut R. Gambar b dan gambar c menunjukkan bahwa kelompok Makalah IF2091 Struktur Diskrit – Sem. I Tahun 2011/2012 semut berjalan pada kecepatan yang sama dengan meninggalkan *feromon* atau jejak kaki di jalan yang telah dilalui. *Feromon* yang ditinggalkan oleh kumpulan semut yang melalui jalan atas telah mengalami banyak penguapan karena semut yang melalui jalan atas berjumlah lebih sedikit dari pada jalan yang di

bawah. Hal ini dikarenakan jarak yang ditempuh lebih panjang daripada jalan bawah. Sedangkan *feromon* yang berada di jalan bawah, penguapannya cenderung lebih lama. Karena semut yang melalui jalan bawah lebih banyak daripada semut yang melalui jalan atas. Gambar d atas menunjukkan bahwa semut-semut yang lain pada akhirnya memutuskan untuk melewati jalan bawah karena *feromon* yang ditinggalkan masih banyak. Sedangkan *feromon* pada jalan atas sudah banyak menguap sehingga semut-semut tidak memilih jalan atas tersebut. Semakin banyak semut yang melalui jalan bawah maka semakin banyak semut yang mengikutinya.

Demikian juga dengan jalan atas, semakin sedikit semut yang melalui jalan atas, maka *feromon* yang ditinggalkan semakin berkurang bahkan hilang. Dari sinilah kemudian terpilihlah jalur terpendek antara sarang dan sumber makanan.

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

Langkah 1 :

- a. Inisialisasi harga parameter-parameter algoritma. Parameter-parameter yang di inisialisasi adalah :
  1. Intensitas jejak semut antar kampus dan perubahannya (  $ij$  ).
  2. Banyak kampus (  $n$  ) termasuk koordinat (  $x,y$  ) atau jarak antar kampus (  $dij$  ).
  3. Koordinat awal dan koordinat akhir.
  4. Tetapan siklus semut (  $Q$  ).
  5. Tetapan pengendali intensitas jejak semut (  $\alpha$  ), nilai  $\alpha \geq 0$ .
  6. Tetapan pengendali visibilitas (  $\beta$  ), nilai  $\beta > 0$ .
  7. Visibilitas antar kampus =  $1/dij$  (  $ij$  ).
  8. Banyak semut (  $m$  ).
  9. Tetapan penguapan jejak semut (  $\rho$  ), nilai harus  $> 0$  dan  $< 1$  untuk mencegah jejak pheromone yang tak terhingga.
  10. Jumlah siklus maksimum (  $NC_{max}$  ) bersifat tetap selama algoritma dijalankan, sedangkan  $ij$  akan selalu diperbaharui harganya pada setiap siklus algoritma mulai dari siklus pertama (  $NC=1$  ) sampai tercapai jumlah siklus maksimum (  $NC=NC_{max}$  ) atau sampai terjadi konvergensi.

- b. Inisialisasi titik pertama setiap semut. Setelah inisialisasi  $ij$  dilakukan, kemudian  $m$  semut ditempatkan pada titik pertama tertentu secara acak.

Langkah 2:

Pengisian titik pertama ke dalam *tabulist*. Hasil inisialisasi titik pertama setiap semut dalam langkah 1 harus diisikan sebagai elemen pertama *tabulist*. Hasil dari langkah ini adalah terisinya elemen pertama *tabulist* setiap semut dengan indeks titik tertentu, yang berarti bahwa setiap *tabulist* 1 bisa berisi indeks 1 sampai  $n$  sebagaimana hasil inisialisasi pada langkah 1.

Langkah 3:

Penyusunan rute kunjungan setiap semut ke setiap kampus. Koloni semut yang sudah terdistribusi ke sejumlah atau setiap kampus, akan mulai melakukan perjalanan dari kampus yang pertama masing-masing sebagai titik asal dan salah satu kampus-kampus lainnya sebagai kampus tujuan. Kemudian dari kampus kedua masing-masing, koloni semut akan melanjutkan perjalanan dengan memilih salah satu kampus yang tidak terdapat pada tabuk sebagai kampus tujuan selanjutnya. Perjalanan koloni semut berlangsung terus menerus sampai semua kampus satu persatu dikunjungi atau telah menempati tabuk. Jika  $s$  menyatakan indeks urutan kunjungan, kampus asal dinyatakan sebagai  $tabuk(s)$  dan kampus-kampus lainnya dinyatakan sebagai  $\{N - tabuk\}$ , maka untuk menentukan kampus tujuan digunakan persamaan probabilitas kampus untuk dikunjungi sebagai berikut :

$$p_{ij}^k = \frac{[tij]^\alpha \cdot [nij]^\beta}{\sum_{k' \in \{N - tabuk_k\}} [tik]^\alpha \cdot [nik]^\beta} \text{ untuk } j \in \{N - tabuk_k\} \dots \dots \dots (1)$$

Dan

$$p_{ij}^k = 0, \text{ untuk } j \text{ lainnya} \dots \dots \dots (2)$$

dengan  $i$  sebagai indeks kampus asal dan  $j$  sebagai indeks kampus tujuan.

Langkah 4:

- a. Perhitungan panjang rute setiap semut. Perhitungan panjang rute tertutup (*length closed tour*) atau  $L_k$  setiap semut dilakukan setelah satu siklus diselesaikan oleh semua semut. Perhitungan ini dilakukan berdasarkan tabuk masing-masing dengan persamaan berikut :

$$L_k = d_{tabuk_k(n), tabuk_k(1)} + \sum_{s=1}^{n-1} d_{tabuk_k(s), tabuk_k(s+1)} \dots \dots \dots (3)$$

dengan  $d_{ij}$  adalah jarak antara kampus  $i$  ke kampus  $j$  yang dihitung berdasarkan persamaan :

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \dots \dots \dots (4)$$

- b. Pencarian rute terpendek. Setelah  $L_k$  setiap semut dihitung, akan didapat harga minimal panjang rute tertutup setiap siklus atau  $L_{minNC}$  dan harga minimal panjang rute tertutup secara keseluruhan adalah atau  $L_{min}$ .

- c. Perhitungan perubahan harga intensitas jejak kaki semut antar kampus. Koloni semut akan meninggalkan jejak-jejak kaki pada lintasan antar kampus yang dilaluinya. Adanya penguapan dan perbedaan jumlah semut yang lewat, menyebabkan kemungkinan terjadinya perubahan harga intensitas jejak kaki semut antar kampus. Persamaan perubahan ini adalah:

$$\Delta T_{ij} = \sum_{k=1}^m \Delta t_{ij}^k \dots \dots \dots (5)$$

Adalah perubahan harga intensitas jejak kaki semut antar kampus setiap semut yang dihitung berdasarkan persamaan :

$$\Delta t_{ij}^k = \frac{q}{L_k}, \text{ untuk } (i,j) \in \text{kampus asal dan kampus tujuan dalam } tabuk_k \dots \dots \dots (6)$$

$$\Delta t_{ij}^k = 0, \text{ untuk } (i,j) \text{ lainnya} \dots \dots \dots (7)$$

Langkah 5:

- a. Perhitungan harga intensitas jejak kaki semut antar kampus untuk siklus selanjutnya. Harga intensitas jejak kaki semut antar kampus pada semua lintasan antar kampus ada kemungkinan berubah karena adanya penguapan dan perbedaan jumlah semut yang melewati. Untuk siklus selanjutnya, semut yang akan melewati lintasan tersebut harga intensitasnya telah berubah. Harga intensitas jejak kaki semut antar kampus untuk siklus selanjutnya dihitung dengan persamaan :

$$\tau_{ij} = \rho \cdot \tau_{ij} + \Delta\tau_{ij} \dots \dots \dots (8)$$

- b. Atur ulang harga perubahan intensitas jejak kaki semut antar kampus. Untuk siklus selanjutnya perubahan harga intensitas jejak semut antar kampus perlu diatur kembali agar memiliki nilai sama dengan nol.

Langkah 6:

- a. Pengosongan *tabulist*, dan ulangi langkah 2 jika diperlukan. *Tabulist* perlu dikosongkan untuk diisi lagi dengan urutan kampus yang baru pada siklus selanjutnya, jika jumlah siklus maksimum belum tercapai atau belum terjadi konvergensi. Algoritma diulang lagi dari langkah 2 dengan harga parameter intensitas jejak kaki semut antar kampus yang sudah diperbaharui.

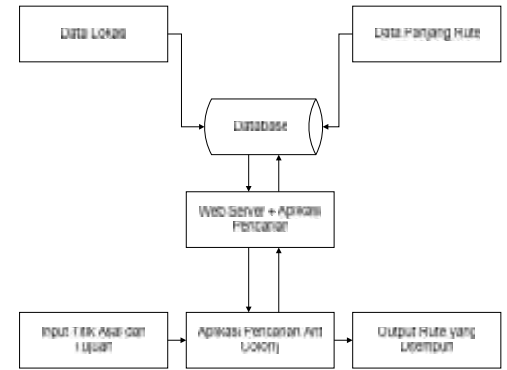
### 3.1 Pembahasan

Rute terpendek merupakan salah satu solusi ketika kita melakukan suatu pencarian kampus teknik informatika khususnya di kota Malang. Karena dengan rute terpendek, kita dapat menghemat biaya dan menghemat waktu. Namun terkadang kita tidak mengetahui rute terpendek untuk menuju salah rute kampus tertentu. Karena banyaknya rute yang banyak dilalui, maka dibuat sistem pencarian rute terpendek kampus teknik informatika di kota Malang dengan menggunakan *Ant Colony*.

### 3.2 Perancangan Sistem

Pada sistem pencarian rute terpendek dengan algoritma semut (*Ant Colony*) ini, sistem akan mencari jalur terdekat untuk mencapai lokasi tujuan. User akan menginputkan lokasi tujuan, setelah user menginputkan lokasi tujuan, sistem akan mengolah data dan mencari rute tujuan yang melewati jalur alternatif untuk mencapai lokasi tujuan. Setelah rute terpendek didapatkan, sistem akan memberikan hasil berupa rute tujuan terpendek beserta jarak yang akan ditempuh.

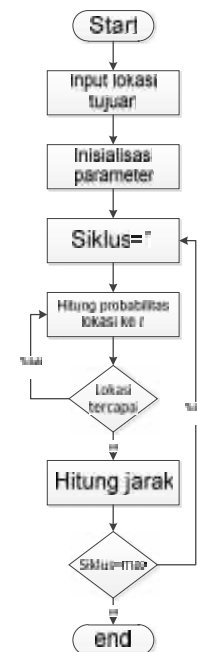
Pada sistem ini tidak membahas optimasi jarak, karena untuk optimasi jarak akan banyak variabel dan *factor* yang berpengaruh. Karena jalan yang berubah-ubah akan sangat berpengaruh jarak yang ditempuh pada tujuan selanjutnya



**Gambar 3** Struktur Sistem Pencarian Rute Terpendek

Sistem pencarian rute terpendek berbasis web ini dimulai dari mengumpulkan data lokasi yang akan digunakan untuk melakukan pencarian rute terpendek. Selain data lokasi, diperlukan juga data panjang rute yang akan ditempuh sesuai dengan keadaan sebenarnya. Kedua data tersebut akan disimpan di database. Aplikasi web yang berisi algoritma ant colony berjalan didalam web server yang juga telah diintegrasikan dengan GIS agar tampilan output yang dihasilkan mampu menampilkan peta jalur yang dicari sesuai dengan input yang diberikan yaitu titik asal dan titik tujuan.

Flowchart sistem akan menjelaskan rancangan alur pada sistem pencarian rute terpendek ini.



**Gambar 4** Flowchart Sistem Pencarian Rute Terpendek

Pada flowchart di atas dapat dijelaskan sebagai berikut:

1. User menginputkan node awal dan node tujuan.
2. Kemudian dilakukan inisialisasi parameter.
3. Dimulai siklus 1 perhitungan pencarian rute terpendek.
4. Kemudian dihitung probabilitas antar lokasi persimpangan.
5. Jika lokasi tujuan telah ditemukan (ya), maka dihitung jarak antar lokasi dengan rute terpendek.
6. Jika tidak, maka dihitung kembali probabilitas antar lokasi persimpangan kembali sampai ditemukan lokasi tujuan.
7. Setelah diketahui jarak terpendeknya maka dicapai siklus maksimum. Jika tidak, maka proses kembali ke siklus=1
8. Setelah semua proses dilalui maka akan didapatkan rute terpendeknya.

### 3.3 Dataset Node Kampus

Pada program pencarian rute terpendek ini digunakan dataset *node* awal, *node* akhir, dan jarak antar *node* (kilometer). Misalkan *node* awal ditentukan dari Kota Malang.

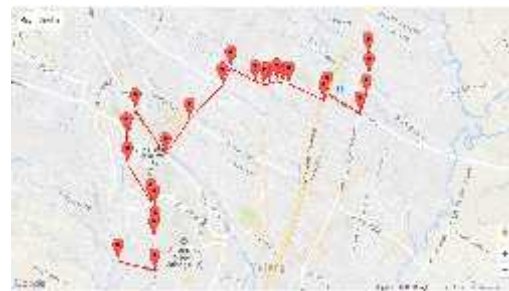
**Tabel 1** Data Node Menentukan Kampus di Malang

No	Node 1	Node 2	Jarak Antar Node ( Km )
1	Bundaran Monumen Pesawat Suhat	Rumah Sakit Permata Bunda	0.26 Km
2	Bundaran Monumen Pesawat Suhat	Pertigaan Bakso Kota Borobudur	0.55 Km
3	Rumah Sakit Permata Bunda	Perempatan Pizza Hut Suhat	1.5 Km
4	Perempatan Pizza Hut Suhat	Perempatan Universitas Brawijaya	0.75 Km
5	Perempatan Universitas Brawijaya	Pertigaan Dinoyo	0.95 Km
6	Pertigaan Dinoyo	Pertigaan Sardo Swalayan	0.45 Km
7	Pertigaan Dinoyo	Universitas Islam Malang	0.4 Km
8	Pertigaan Sardo Swalayan	Kampus UIN	0.45 Km
9	Kampus UIN	Perempatan ITN	0.8 Km
10	Perempatan ITN	Kampus UMM 2	0.14 Km
11	Perempatan ITN	SMK Negeri 2 Malang	0.4 Km
12	Perempatan ITN	Kampus ITN Malang	0.13 Km
13	Universitas Islam Malang	MAN 1 Malang	1.1 Km
14	MAN 1 Malang	Baiduri Ballroom Cafe And Resto	0.35 Km
15	Baiduri Ballroom Cafe And Resto	SMK PGRI 3 Malang	0.26 Km
16	SMK PGRI 3 Malang	RS UMM 3	0.35 Km
17	RS UMM 3	Kampus UMM 3 Malang	0.65 Km
18	Kampus UMM 2	Pertigaan Maestro	0.35 Km
19	Pertigaan Maestro	Olimpico Futsal Area	0.14 Km
20	Olimpico Futsal Area	Perempatan Galunggung	0.7 Km
21	Perempatan Galunggung	Kampus STIKI	0.7 Km
22	SMK Negeri 2 Malang	ATM BNI UM	0.5 Km
23	ATM BNI UM	Perempatan UM	0.35 Km
24	Pertigaan Bakso Kota Borobudur	Warung Makan Cak Pi'i	0.19 Km
25	Warung Makan Cak Pi'i	Perempatan Doremi	0.79 Km
26	Perempatan Doremi	ATM Bank Panin Borobudur	0.16 Km
27	ATM Bank Panin Borobudur	Indomart Borobudur	0.12 Km
28	Indomart Borobudur	Kantor Pos Blimbing	0.11 Km
29	Kantor Pos Blimbing	Pertigaan Masjid Sabililah Malang	2.7 Km
30	Pertigaan Masjid Sabililah Malang	Pertigaan Polek Blimbing	0.13 Km
31	Pertigaan Polek Blimbing	Perempatan Panji Suroso	0.7 Km
32	Perempatan Panji Suroso	Perempatan Lori	0.3 Km
33	Perempatan Lori	Pertigaan STIKMA	0.45 Km
34	Pertigaan STIKMA	Komplek Araya	0.2 Km

**Tabel 2** Daftar *Node* Kampus

No	Nama Node	Nama Kampus
1	Bundaran Monumen Pesawat Suhat	STMIK ASIA Malang
2	Perempatan Universitas Brawijaya	Universitas Brawijaya
3	Kampus UIN	Universitas Islam Negeri Malang (UIN)
4	Universitas Islam Malang	Universitas Islam Malang (UNISMA)
5	Kampus UMM 3 Malang	Universitas Muhammadiyah Malang
6	Kampus STIKI	STIKI Malang
7	Perempatan Dinoyo	Universitas Malang (UM)
8	Kampus ITN Malang	ITN MALANG
9	Pertigaan STIKMA	STIKMA Malang
10	Komplek Araya	BINUS Malang

Adapun daftar jalur atau rute kampus dengan kampus-kampus lain yang diproyeksi pada gambar di bawah ini:



**Gambar 5** Daftar Rute Kampus

### 3.4 Studi Kasus

Misalkan *node* awal dari monumen bundaran monumen pesawat suhat (STMIK ASIA Malang) untuk ke kampus Muhammadiyah Malang Dihitung dengan menggunakan rumus probabilitas:

$$P_{ij}^k = \frac{[\tau_{ij}]^\alpha \cdot [n_{ij}]^\beta}{\sum [\tau_{ij}]^\alpha \cdot [n_{ij}]^\beta}$$

Keterangan:  
Alfa ( $\alpha$ ) = 1

Beta ( $\beta$ ) = 1  
 $T_{ij}$  awal = 0,01

$n_{ij} = 1/d_{ij}$  d: node yang mungkin dilalui

Berikut adalah cara untuk mencari probabilitasnya seperti di bawah ini:

1. Bundaran Pesawat Suhat ke Rumah Sakit Permata Bunda ( jarak 0.26 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.26}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.26}\right]^1} = \frac{0.038}{0.038} = 1$$

Jadi probabilitas Bundaran Pesawat Suhat ke Rumah Sakit Permata Bunda = 1

2. Rumah Sakit Permata Bunda ke Perempatan Pizza Hut Suhat ( jarak 1.5 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{1.5}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{1.5}\right]^1} = \frac{0.006}{0.006} = 1$$

Jadi probabilitas Rumah Sakit Permata Bunda ke Perempatan Pizza Hut Suhat = 1

3. Perempatan Pizza Hut Suhat ke Pertigaan Brawijaya ( jarak 0.75 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.75}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.75}\right]^1} = \frac{0.013}{0.013} = 1$$

Jadi probabilitas Perempatan Pizza Hut Suhat ke Pertigaan Brawijaya = 1

4. Pertigaan Brawijaya ke Pertigaan Dinoyo ( jarak 0.95 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.95}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.95}\right]^1} = \frac{0.010}{0.010} = 1$$

Jadi probabilitas Pertigaan Brawijaya ke Pertigaan Dinoyo = 1

5. Pertigaan Dinoyo ke Universitas Islam Malang (UNISMA) ( jarak 0.4 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.4}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.4}\right]^1} = \frac{0.025}{0.025} = 1$$

Jadi probabilitas Pertigaan Dinoyo ke Universitas Islam Malang (UNISMA) = 1

6. Universitas Islam Malang ke MAN 1 Malang ( jarak 1.1 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{1.1}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{1.1}\right]^1}$$

$$= \frac{0.009}{0.009} = 1$$

Jadi probabilitas Universitas Islam Malang ke MAN 1 Malang = 1

7. MAN 1 Malang ke Baiduri Ballroom Cafe And Resto ( jarak 0.35 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.35}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.35}\right]^1} = \frac{0.028}{0.028} = 1$$

Jadi probabilitas MAN 1 Malang ke Baiduri Ballroom Cafe And Resto = 1

8. Baiduri Ballroom Cafe And Cafe ke SMK PGRI 3 Malang ( jarak 0.26 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.26}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.26}\right]^1} = \frac{0.038}{0.038} = 1$$

Jadi probabilitas Baiduri Ballroom Cafe And Cafe ke SMK PGRI 3 Malang = 1

9. SMK PGRI 3 Malang ke Rumah Sakit Muhammadiyah Malang ( jarak 0.35 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.35}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.35}\right]^1} = \frac{0.028}{0.028} = 1$$

Jadi probabilitas SMK PGRI 3 Malang ke Rumah Sakit Muhammadiyah Malang = 1

10. Rumah Sakit Universitas Muhammadiyah Malang ke Kampus Muhammadiyah Malang ( jarak 0.65 km )

$$= \frac{[0.01]^2 \cdot \left[\frac{1}{0.65}\right]^1}{[0.01]^2 \cdot \left[\frac{1}{0.65}\right]^1} = \frac{0.015}{0.015} = 1$$

Jadi probabilitas Rumah Sakit Universitas Muhammadiyah Malang ke Kampus Muhammadiyah Malang = 1

Jadi telah didapatkan jalur yaitu melalui Bundaran Pesawat Suhat - Rumah Sakit Permata Bunda - Perempatan Pizza Hut Suhat - Pertigaan Brawijaya - Pertigaan Dinoyo - Universitas Islam Malang (UNISMA) - MAN 1 Malang - Baiduri Ballroom And Cafe - SMK PGRI 3 Malang - Rumah Sakit Muhammadiyah Malang - Kampus Muhammadiyah Malang = 0.26 + 1.5 + 0.75 + 0.95 + 0.04 + 1.1 + 0.35 + 0.26 + 0.35 + 0.65 = 6.57. Jadi jarak sebenarnya sebesar 6.57 Km dan dinamakan dengan rute 1.

Kemudian hitung probabilitas dari Perempatan ITN ke Kampus Muhammadiyah



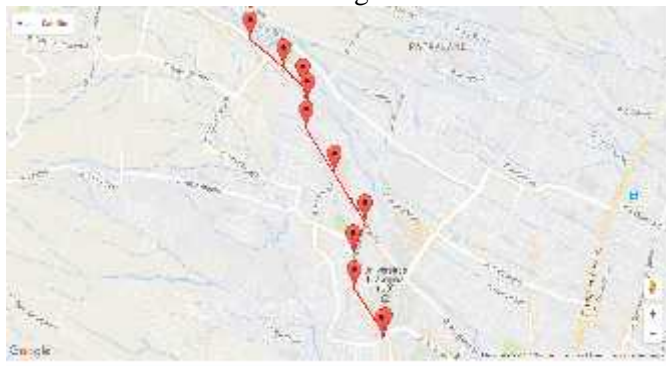
Malang sebagai rute 2 yang melewati beberapa *node-node*. Cara perhitungan sama dengan perhitungan rute 1 diatas. Sampai didapatkan jalur yaitu melalui Perempatan ITN – Kampus UIN – Pertigaan Sardo Swalayan – Pertigaan Dinoyo – Universitas Islam Malang (UNISMA) – MAN 1 Malang – Baiduri *Ballroom Cafe And Resto* – SMK PGRI 3 Malang – Rumah Sakit Muhammadiyah Malang – Kampus Muhammadiyah Malang =  $0.13 + 0.8 + 0.45 + 0.45 + 0.4 + 0.1 + 0.35 + 0.26 + 0.35 + 0.65 = 4.94$ . Jadi jarak sebenarnya sebesar 4.94 Km dan dinamakan dengan rute 2..

Jadi dapat disimpulkan bahwa rute terdekat menuju kampus Muhammadiyah Malang adalah rute 2. Dikarenakan rute 2 adalah jarak yang paling dekat daripada rute 1.

Rute dari ITN MALANG (KAMPUS ITN MALANG) menuju UNIVERSITAS MUHAMMADIYAH MALANG (KAMPUS UIN MALANG) menggunakan jarak 4.94 Km

NO	LOKASI	JARAK	WAKTU
1	PEREMPATAN ITN MALANG	0.13	0.00
2	KAMPUS UIN MALANG	0.80	0.00
3	PERTIGAAN SARDO SWALAYAN	0.45	0.00
4	PERTIGAAN DINOYO	0.45	0.00
5	UNIVERSITAS ISLAM MALANG	0.40	0.00
6	MAN 1 MALANG	0.10	0.00
7	BALURAN	0.35	0.00
8	SMK PGRI 3 MALANG	0.26	0.00
9	RUMAH SAKIT MUHAMMADIYAH MALANG	0.35	0.00
10	KAMPUS MUHAMMADIYAH MALANG	0.65	0.00

**Gambar 6** Node Terpendek Kampus Muhammadiyah Malang



**Gambar 7** Jalur Rute Terpendek

**4. Penutup**

**4.1 Kesimpulan**

Dari hasil pembahasan pembuatan aplikasi dalam menentukan rute terpendek kampus teknik informatika di kota Malang dengan menggunakan metode *ant colony* ini dapat diambil kesimpulan bahwa rute terpendek didapatkan dari *tabulist* yang diperoleh masing-masing semut.

**4.2 Saran**

Adapun saran dalam pembuatan aplikasi ini adalah :

1. Data *graph* yang digunakan lebih kompleks dan lebih detail
2. Bobot antar titik tidak hanya berupa bobot jarak saja.
3. Permasalahan yang akan di selesaikan lebih dimodifikasi tidak hanya jalur terpendek tapi juga untuk menghasilkan keluaran yang lain.
4. Disarankan dalam penelitian selanjutnya tidak hanya menampilkan satu kota saja, tetapi bisa menggunakan lebih dari satu kota misalnya dalam seluruh kabupaten.
5. Untuk penelitian selanjutnya tampilan rute terpendeknya ditampilkan menggunakan graf dengan jarak.

**DAFTAR PUSTAKA**

Demers, M.N., *Fundamental of Geographic Information Systems*, John Wileys & Sons, Inc., New York. 1997  
 ESRI, "Understanding GIS : The Arc/ Info Method Environmental System", Research Institute, Redlands, California. 1990  
 Chrisman, Nicholas. *Exploring Geographic Information System*. John Willey & Sons, Inc., New York. 1997  
 Foote, K.E. Good map. Dalam *Cartographic Communication* , 2002  
 Purwadhi, F, *Pengantar Interpretasi Citra Penginderaan Jauh*. LAPAN-UNNES. 2008