

# PENERAPAN TRAVELING SALESMAN PROBLEM PADA PENYEBARAN BROSUR PENERIMAAN MAHASISWA BARU SEKOLAH TINGGI TEKNOLOGI NURUL JADID MENGGUNAKAN ANT COLONY OPTIMIZATION

Olief Imandira Ratu Farisi<sup>1)</sup>, Gulpi Qorik Oktagalu Pratamasunu<sup>2)</sup>, dan  
Khairul Anas Nur Islam Hadi<sup>3)</sup>

<sup>1, 2, 3)</sup> Teknik Informatika, Universitas Nurul Jadid  
Paiton, Probolinggo

e-mail: olief.ilmandira@gmail.com<sup>1)</sup>, gulpi.qorik@gmail.com<sup>2)</sup>, khairul.id28@gmail.com<sup>3)</sup>

## ABSTRAK

Penyebaran brosur Penerimaan Mahasiswa Baru (PMB) Sekolah Tinggi Teknologi Nurul Jadid (STTNJ) memiliki rute kunjungan ke 37 Sekolah Menengah Atas (SMA/SMK/MA) di Probolinggo. Setiap sekolah hanya dapat dikunjungi satu kali dan setelah selesai tim penyebar akan kembali lagi ke STTNJ. Permasalahan ini sesuai dengan konsep Traveling Salesman Problem (TSP), dimana tujuannya adalah mencari rute yang paling optimal, sehingga penyebaran brosur menjadi lebih efektif dan efisien. Untuk menyelesaikan permasalahan ini, diusulkan penggunaan metode Ant Colony Optimization (ACO). ACO adalah metode yang didesain untuk menyelesaikan kasus TSP, terinspirasi dari perilaku koloni semut dalam menemukan jalur terpendek dari sarang menuju sumber makanan. Uji coba dilakukan untuk menentukan parameter ACO dengan waktu komputasi yang lebih cepat dan hasil yang mendekati optimal. Dari hasil uji coba didapat nilai:  $\alpha = 2$ ,  $\beta = 5$ ,  $\rho = 0,2$ , iterasi = 320 dan semut = 15. Selanjutnya dilakukan percobaan sebanyak 30 kali pada kasus TSP penyebaran brosur PMB, dan mendapat hasil solusi terbaik 181,6 km dengan waktu komputasi 86,9 detik. Rata-rata hasil solusi dari 30 kali percobaan adalah 187,28 km dengan rata-rata waktu komputasi 103,34 detik. Hasil penelitian menunjukkan metode ACO dapat mencari rute optimal dengan waktu yang singkat.

**Kata Kunci:** Penyebaran brosur, Traveling Salesman Problem, Ant Colony Optimization.

## ABSTRACT

The distribution of the New Student Admission brochure of Sekolah Tinggi Teknologi Nurul Jadid (STTNJ) has a route to visit 37 high schools in Probolinggo. Each school can be visited exactly one and after completing the tour, the distribution team will return to STTNJ. This problem is in accordance with the concept of Traveling Salesman Problem (TSP), where the goal is finding the optimal route, so the distribution of brochures be more effective and efficient. To solve this problem, we propose Ant Colony Optimization (ACO). ACO is a method designed to solve the TSP, inspired by the behavior of ant colonies in finding the shortest path from the nest to the food source. Experiments were conducted to determine the ACO parameters with faster computational time and the optimal solution. The experimental results obtained the value of  $\alpha = 2$ ,  $\beta = 5$ ,  $\rho = 0.2$ , the number of iteration = 320 and ant = 15. From 30 experiments in the case of TSP's distribution of brochures, we get the best solution results 181.6 km with a computational time of 86.9 seconds. The average solution result of 30 experiments is 187.28 km with an average computational time of 103.34 seconds. The results showed that ACO method can find the optimal route with a short time.

**Keywords:** Distribution of brochure, Traveling Salesman Problem, Ant Colony Optimization.

## I. PENDAHULUAN

**T**RAVELING Salesman Problem (TSP) adalah salah satu masalah optimasi yang penting dalam sistem distribusi. TSP secara umum digambarkan sebagai suatu kasus dimana seseorang harus mengunjungi sejumlah kota dari suatu pusat fasilitas dan kembali lagi ke tempat pemberangkatan semula. Tujuan dari masalah ini adalah untuk meminimalkan total jarak tempuh salesman. Meskipun

TSP sepertinya mudah dinyatakan, akan tetapi sangat sulit untuk diselesaikan terutama untuk persoalan dengan jumlah kota yang banyak. Berdasarkan hal tersebut, banyak peneliti lebih memusatkan kepada pengembangan metode-metode pendekatan (*heuristic*) seperti *simulated annealing*, algoritma semut (*Ant Algorithm*), Algoritma Genetika, *Tabu search*, dan lain sebagainya [1].

Saat ini penyebaran brosur penerimaan mahasiswa baru (PMB) di Sekolah Tinggi Teknologi Nurul Jadid dilakukan secara subyektif dan tidak berdasarkan jarak

terdekat sekolah menengah atas (SMA/SMK/MA) yang akan dikunjungi. Padahal, di Probolinggo saja terdapat 37 sekolah yang harus dikunjungi. Hal ini tentu akan memakan waktu yang sangat lama jika terdapat kesalahan dalam penentuan urutan kunjungan. Oleh karena itu, dibutuhkan suatu metode yang dapat mencari urutan kunjungan sekolah yang memiliki rute terpendek.

Pada penelitian ini diusulkan penggunaan metode *ant colony optimization* (ACO) untuk menghitung rute terpendek dari sekolah yang akan dikunjungi. ACO diaopsi dari perilaku koloni semut yang dikenal sebagai sistem semut. Seekor semut dapat menjangkau sumber makanan dengan rute terdekat dari sarangnya dengan memanfaatkan material kimia yang disebut feromon, yaitu alat komunikasi berupa hormon yang dikeluarkan oleh semut sebagai penunjuk jalan bagi semut yang lain. Pada awalnya semut-semut tersebut akan melalui semua jalur yang memungkinkan secara acak sambil melepaskan feromon, feromon tersebut akan menarik perhatian semut lain untuk mengikuti suatu rute. Semakin banyak jumlah feromon yang ada pada suatu rute, semakin potensial rute tersebut untuk diikuti oleh semut-semut lainnya [2].

Dengan menggunakan metode ACO, diharapkan rute optimal dalam penyebaran brosur dapat ditemukan. Sehingga proses distribusi brosur dapat meningkat efektifitas dan efisiensinya.

## II. ANT COLONY OPTIMIZATION DALAM TSP

Awalnya ACO diterapkan oleh penemunya, Marco Dorigo untuk menyelesaikan persoalan TSP. Dalam TSP [3]. Setiap semut memulai rutenya melalui sebuah kota awal yang sama yaitu 1 dan akan kembali lagi ke kota 1. Secara berulang, setiap semut akan memilih kota berikutnya untuk dikunjungi sehingga setiap semut akan menemukan rute yang dilaluinya sampai kembali ke kota 1 lagi. Pemilihan kota-kota yang akan dilaluinya didasarkan pada suatu fungsi probabilitas (tingkat kemungkinan), dengan mempertimbangkan jarak antar kota tersebut dan jumlah feromon yang terdapat pada ruas yang menghubungkan kota sekarang ke kota selanjutnya. Semut-semut akan memilih untuk bergerak menuju ke kota-kota yang dihubungkan dengan ruas yang pendek dan memiliki tingkat feromon yang tinggi [4].

Perlu diketahui bahwa setiap semut memiliki sebuah memori yang dinamai tabu list, berisi semua kota yang telah dikunjunginya pada setiap tur. Tabu list ini mencegah semut untuk mengunjungi kota-kota yang sebelumnya telah dikunjungi selama rute tersebut berlangsung, sehingga tidak akan ada satu kota dikunjungi lebih dari satu kali. Setelah satu iterasi selesai atau semua semut telah menyelesaikan rutenya, akan dilakukan pembaruan feromon secara global yang

meliputi penguapan pada semua sisi dan penambahan feromon pada sisi-sisi yang merupakan bagian dari tur. Dan setiap semut akan dihitung panjang rute yang telah ditempuhnya. Semakin pendek sebuah rute yang dihasilkan oleh seekor semut, semakin besar jumlah feromon yang diletakkan pada ruas-ruas yang dilaluinya. Dengan kata lain, ruas-ruas yang merupakan bagian dari rute yang pendek adalah ruas-ruas yang menerima jumlah feromon yang lebih besar. Hal ini menyebabkan ruas-ruas yang diberi feromon lebih banyak akan lebih disukai pada iterasi selanjutnya. Lalu jalur terpendek yang ditemukan oleh semut akan disimpan, dan semua tabu list yang ada dikosongkan kembali untuk menyimpan kota-kota pada iterasi berikutnya [4].

Proses di atas kemudian diulangi sampai memenuhi jumlah iterasi maksimum yang ditentukan user atau tidak terjadi lagi perubahan solusi. Aturan transisi status atau probabilitas semut- $k$  dari kota- $i$  memilih kota- $j$  ditunjukkan dalam Persamaan 2.1.

$$p_{ij} = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{u \in U_k} [\tau_{iu}(t)]^\alpha [\eta_{iu}(t)]^\beta}, & j \in U_k \\ 0, & \text{lainnya} \end{cases} \quad (2.1)$$

dimana  $\tau_{ij}(t)$  adalah intensitas trail, yaitu jumlah feromon yang tersebar pada sisi  $(i, j)$  pada waktu  $t$ ,  $\eta_{ij}$  adalah visibilitas yang merupakan invers jarak antara kota- $i$  ke kota- $j$  ( $\eta_{ij} = 1/d_{ij}$ ), dan  $U_k$  adalah himpunan kota-kota yang belum dikunjungi oleh semut- $k$ . Dari Persamaan 2.1 ini ruas yang lebih pendek dan memiliki jumlah feromon yang lebih banyak akan lebih besar probabilitasnya untuk dipilih dalam rute.

Setelah satu iterasi, yaitu dimana semua semut telah berhasil menyelesaikan turnya, intensitas trail akan diperbarui dengan aturan pembaruan feromon global pada Persamaan 2.2 berikut.

$$\tau_{ij}(t+n) = \rho \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij}^k \quad (2.2)$$

dimana  $(1-\rho)$  adalah tingkat penguapan feromon dan  $\Delta \tau_{ij}^k$  adalah jumlah feromon yang dihasilkan oleh semut- $k$  pada sisi  $(i, j)$  yang diformulasikan sebagai berikut.

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{if semut-}k \text{ melewati sisi } (i, j) \\ 0, & \text{lainnya} \end{cases} \quad (2.3)$$

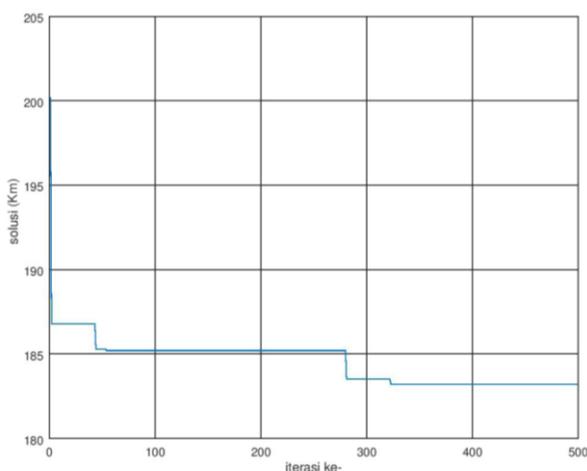
dengan  $Q$  adalah bilangan konstant dan  $L_k$  adalah panjang tur dari semut- $k$ .

### III. HASIL UJI COBA DAN PEMBAHASAN

Uji coba dilakukan menggunakan data jarak 37 kota yang ada di daerah Probolinggo. Data ini berisi jarak masing-masing kota ke kota yang lain yang dikumpulkan satu persatu dari peta online yang ada. Data tersebut disimpan dalam bentuk tabel yang berekstensi “xls”. Selanjutnya data tersebut akan digunakan sebagai data uji parameter optimal metode ACO untuk TSP.

Tahap pertama dalam uji coba metode ACO adalah menentukan banyak iterasi minimum yang diperlukan untuk mencapai solusi optimum. Parameter yang digunakan pada uji coba ini antara lain  $\alpha = 1$ ,  $\beta = 5$ ,  $\rho = 0,5$ ,  $Q = 100$  dan semut sebanyak jumlah kota (Dorigo, 1996). Pada studi kasus ini jumlah semut yang digunakan sebanyak 37 semut. Metode ACO dijalankan dengan 500 iterasi. Hasil uji coba penentuan banyak iterasi minimum ditunjukkan pada Gambar 2.1. Didapat bahwa iterasi minimal yang diperlukan ACO untuk mencapai konvergensi adalah 320. Banyak iterasi ini akan digunakan pada uji coba selanjutnya.

Tahap kedua adalah menentukan parameter  $\alpha$  dan  $\beta$  yang mempengaruhi probabilitas penentuan sekolah yang akan dikunjungi selanjutnya. Nilai yang akan diujicobakan adalah  $\alpha \in \{0,5,1,2\}$  dan  $\beta \in \{1,2,5\}$  [3]. Pada uji coba ini, pengaturan default yang digunakan antara lain  $\rho = 0,5, Q = 100$ , semut sebanyak 37, dan banyak iterasi adalah 320.



**Gambar 2.1** Grafik hasil uji coba penentuan banyak iterasi minimum

**Tabel 2.1.** Penentuan nilai  $\alpha$  dan  $\beta$

| A   | B | Hasil (km) | Waktu (detik) |
|-----|---|------------|---------------|
| 0,5 | 1 | 289        | 314,51        |
|     | 2 | 208,9      | 272,06        |
|     | 5 | 185,9      | 239,47        |
| 1   | 1 | 199,4      | 276,87        |
|     | 2 | 188,5      | 318,62        |
|     | 5 | 186,1      | 268,55        |
| 2   | 1 | 196,5      | 286,43        |
|     | 2 | 189,6      | 260,82        |
|     | 5 | 183,9      | 241,48        |

**Tabel 2.2.** Nilai koefisien  $\rho$

| $\rho$ | Hasil (km) | Waktu (detik) |
|--------|------------|---------------|
| 0,1    | 186,7      | 220,78        |
| 0,2    | 182        | 290,23        |
| 0,3    | 186,6      | 289,98        |
| 0,4    | 183,6      | 287,21        |
| 0,5    | 182,4      | 274,84        |
| 0,6    | 185        | 295,37        |
| 0,7    | 185,7      | 269,65        |
| 0,8    | 185,7      | 286,12        |
| 0,9    | 182,6      | 232,37        |

Berdasarkan Tabel 2.1, hasil terbaik diperoleh jika  $\alpha = 2$  dan  $\beta = 5$ . Hal ini berarti pada studi kasus ini, visibilitas memiliki prioritas yang lebih besar dibandingkan dengan feromon. Dari 3 nilai  $\alpha$  yang diujicobakan, hasil optimum didapat ketika  $\beta = 5$ . Waktu komputasi terbaik diperoleh ketika  $\alpha = 0,5$  dan  $\beta = 5$ . Dengan mempertimbangkan solusi yang dihasilkan dan waktu komputasi, maka pada studi kasus ini, dipilih  $\alpha = 2$  dan  $\beta = 5$ .

Tahap ketiga uji coba ACO adalah mencari parameter  $\rho$  dimana  $(1 - \rho)$  menyatakan penguapan feromon. Nilai parameter  $\rho$  yang diujicobakan memiliki interval (0,1). Pada uji coba ini, pengaturan default yang digunakan antara lain  $\alpha = 2$ ,  $\beta = 5, Q = 100$ , semut = 37 dan banyak iterasi adalah 320.

Dari sembilan nilai  $\rho$ , hasil terbaik diperoleh ketika  $\rho = 0,2$ . Berdasarkan hasil uji coba pada Tabel 2.2 waktu komputasi terbaik 220,78 detik diperoleh ketika  $\rho = 0,1$ . Pada uji coba tersebut, untuk semua nilai  $\rho$  yang diujicobakan tidak terdapat perbedaan waktu komputasi yang cukup signifikan. Sehingga, pada studi kasus ini, dipilih  $\rho = 0,2$ .

Tahap selanjutnya adalah menentukan banyak semut yang optimal. Nilai parameter  $\alpha = 2$ ,  $\beta = 5$ ,  $\rho = 0,2$ ,

dan  $Q = 100$ . Banyak  $n$  semut yang diujicobakan adalah  $n \in \{5,10,15,20,25,30,35\}$  dengan iterasi sebanyak 320 kali. Hasil uji coba penentuan banyak semut untuk menjalankan ACO pada studi kasus ditunjukkan pada Tabel 2.3 berikut.

**Tabel 2.3.** Penentuan banyak semut

| Semut | Hasil (km) | Waktu (detik) |
|-------|------------|---------------|
| 5     | 195,5      | 27,256        |
| 10    | 188,8      | 54,565        |
| 15    | 183,2      | 86,848        |
| 20    | 183,5      | 117,92        |
| 25    | 185,4      | 178,19        |
| 30    | 184,1      | 213,29        |
| 35    | 183,7      | 223,86        |

Dari Tabel 2.3 terlihat bahwa banyak semut minimal yang dapat menghasilkan solusi terbaik adalah 15 semut. Waktu komputasi yang diperlukan adalah 86,848 detik. Sehingga nilai banyak semut ini akan digunakan sebagai nilai awal dalam penyelesaian TSP penyebaran brosur menggunakan ACO.

Setelah parameter awal yang dibutuhkan oleh metode ACO didapatkan di tahapan sebelumnya, langkah selanjutnya adalah dengan melakukan uji coba untuk menemukan rute atau solusi terbaik pada penyebaran brosur ke tiga puluh tujuh sekolah di Probolinggo. Uji coba dilakukan dengan menjalankan metode ACO sebanyak 30 kali percobaan. Berdasarkan

beberapa uji coba di atas, metode ACO yang terbaik diinisialisasi sebagai berikut.

1. Parameter standar yang digunakan pada uji coba adalah  $\alpha = 2$ ,  $\beta = 5$ ,  $\rho = 0,2$ , dan  $Q = 100$ .

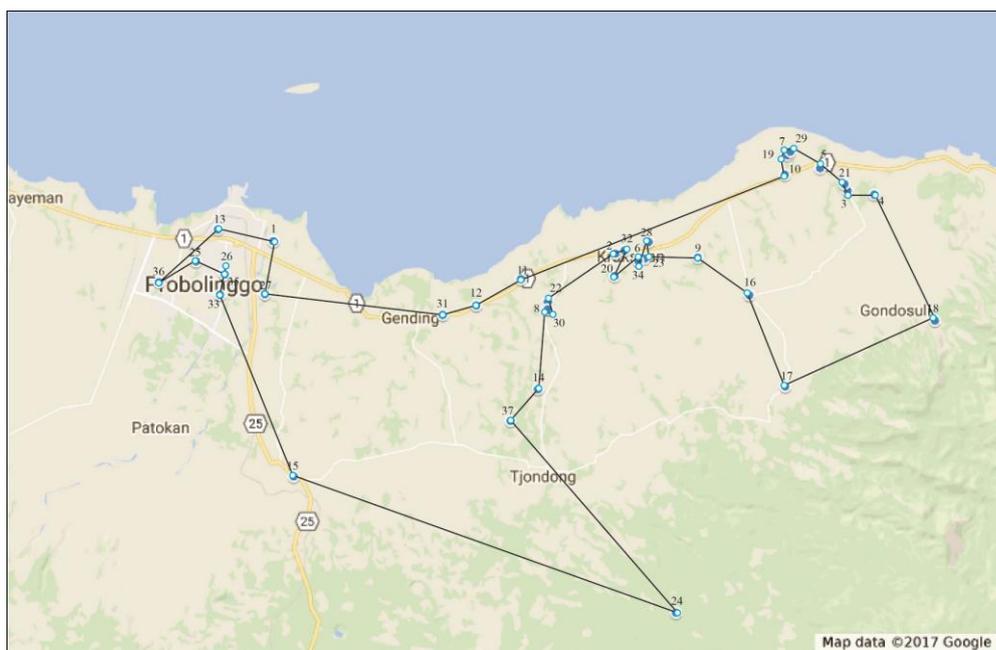
2. Banyak semut yang diperlukan untuk menghasilkan solusi optimum studi kasus adalah 15.

3. Iterasi minimal yang diperlukan untuk menghasilkan solusi optimum studi kasus sebanyak 320 iterasi.

Hasil Uji coba ACO pada penyebaran brosur penerimaan mahasiswa baru Sekolah Tinggi Teknologi Nurul Jadid untuk 30 kali percobaan ditunjukkan pada Tabel 2.4. Pada tabel ini terlihat bahwa metode ACO dapat mencari rute-rute tercepat dalam waktu yang cepat. Kecepatan komputasi masing-masing percobaan bervariasi.

Hasil terbaik yang didapat dengan metode ACO untuk 30 kali percobaan adalah 181,6 km dengan waktu komputasi 86,9 detik. Rata-rata solusi dari 30 percobaan adalah 187,28 km dengan rata-rata waktu komputasi 103,34 detik. Rute terbaik yang dihasilkan dapat dilihat pada Gambar 2.2.

Dari Gambar 2.2 terlihat bahwa rute yang dihasilkan oleh metode ACO sesuai dengan logika pemikiran manusia dengan mengambil rute yang tidak berbelit-belit, memilih lokasi terdekat dari tempat yang terakhir dikunjungi dan terorganisir dengan baik. Hal ini juga dibuktikan dengan hasil jarak rute terpendek yang dihasilkan.



**Gambar 2.2** Peta jalur penyebaran brosur PMB hasil solusi terbaik dari 30 kali percobaan

**Tabel 2.4.** Hasil percobaan ACO dengan parameter yang telah dioptimasi

| No | Hasil (km) | Waktu (detik) |
|----|------------|---------------|
| 1  | 186,5      | 99,65         |
| 2  | 187,7      | 83,42         |
| 3  | 185,1      | 87,857        |
| 4  | 184,5      | 87,205        |
| 5  | 186,5      | 90,545        |
| 6  | 193,9      | 88,699        |
| 7  | 187,7      | 99,908        |
| 8  | 188,6      | 84,905        |
| 9  | 186,4      | 83,303        |
| 10 | 190        | 90,839        |
| 11 | 187,1      | 87,297        |
| 12 | 187,5      | 87,374        |
| 13 | 186,7      | 82,9          |
| 14 | 181,6      | 86,9          |
| 15 | 185        | 81,54         |
| 16 | 186,2      | 83,803        |
| 17 | 185,4      | 82,637        |
| 18 | 189,6      | 91,157        |
| 19 | 187,6      | 274,05        |
| 20 | 186,2      | 113,95        |
| 21 | 185,6      | 115,12        |
| 22 | 188,8      | 113,8         |
| 23 | 186,7      | 114,58        |
| 24 | 189,9      | 114,1         |
| 25 | 188,1      | 115,54        |
| 26 | 186,2      | 115,17        |
| 27 | 192,4      | 126,52        |
| 28 | 187,5      | 107,66        |
| 29 | 188,8      | 106,91        |
| 30 | 184,8      | 103,09        |

#### IV. KESIMPULAN

Dari pembahasan yang telah diuraikan pada bab sebelumnya, dapat diambil suatu kesimpulan tentang algoritma Ant Colony Optimization (ACO) untuk menyelesaikan masalah Traveling Salesman Problem (TSP) pada studi kasus penyebaran brosur penerimaan mahasiswa baru Sekolah Tinggi Teknologi Nurul Jadid (STTNJ). Sebagai berikut:

1. Algoritma ACO dapat digunakan untuk mencari solusi jarak minimal pada studi kasus.

2. Parameter yang diuji memiliki pengaruh pada waktu komputasi dan hasil solusi yang cukup signifikan.

3. Hasil ujicoba pada jumlah iterasi menunjukkan bahwa semakin sedikit iterasi semakin cepat proses pencarian solusi ACO.

4. Hasil uji coba pada parameter  $\alpha$  dan  $\beta$  menunjukkan hasil solusi dengan hasil terbaik didapat ketika  $\beta = 5$ .

5. Pada uji coba nilai  $\rho$ , hasil solusi yang didapat lebih bervariasi, namun dengan waktu komputasi yang tidak banyak berbeda.

6. Hasil uji coba pada jumlah banyak semut, hasil solusi terbaik didapat ketika semut berjumlah 15. Semakin sedikit semut semakin cepat waktu proses yang diperlukan ACO.

7. Hasil terbaik yang didapat pada 30 kali percobaan adalah 181,6 Km dengan waktu komputasi 86,9 detik.

8. Rata-rata hasil solusi dari 30 kali percobaan adalah 187,28 Km dan waktu komputasi rata-rata 103,34 detik.

#### DAFTAR PUSTAKA

- [1] Leksono, A. (2009). Algoritma Ant Colony Optimization (ACO) untuk Menyelesaikan Traveling Salesman Problem (TSP). Skripsi Jurusan Matematika Universitas Diponegoro.
- [2] Budi Santosa, P. (n.d.). Ant Colony Optimization. Lab Komputasi dan Optimasi Industri ITS Surabaya.
- [3] Dorigo, M., Maniezzo, V, and Colomi, A. (1996). The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on System, Man, and Cybernetics-Part B: Cybernetics*, 26(1): 29-41.
- [4] Farisi, O. I. (2015). Penyelesaian Multi-Depot Multiple Traveling Salesman Problem Menggunakan Hybrid Firefly Algorithm – Ant Colony Optimization. Tesis Magister Jurusan Matematika Institut Teknologi Sepuluh Nopember surabaya.