

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA : Valentino Oktawijaya
NRP : 5110100071
DOSEN WALI : Ahmad Saikhu, S.Si., MT.
DOSEN PEMBIMBING : 1. Arya Yudhi Wijaya, S.Kom., M.Kom.
2. Rully Soelaiman, S.Kom., M.Kom.

2. JUDUL TUGAS AKHIR

“Implementasi *Dynamic Travelling Salesman Problem* dengan Menggunakan Algoritma *Random Immigrants Ant Colony Optimization*”

3. LATAR BELAKANG

Travelling Salesman Problem (TSP) merupakan sebuah permasalahan optimasi yang umum termasuk dalam kategori *NP-complete Combinatorial Optimization Problems* (COPs). TSP dapat dijelaskan sebagai sebuah permasalahan dalam mencari jarak minimal sebuah perjalanan terhadap sejumlah n kota dan kota-kota yang ada hanya diperbolehkan untuk dikunjungi sekali dalam suatu kesempatan [1]. Dalam penelitian yang dilakukan oleh M. Dorigo dan L. M. Gambardella pada tahun 1977 dalam penyelesaian permasalahan TSP, terbukti bahwa terbukti dari beberapa algoritma yang diusulkan untuk menyelesaikan permasalahan tersebut, *Ant Colony System* (ACS) berhasil mendapatkan hasil yang terbaik dibandingkan dengan algoritma *Genetic Algorithm* (GA), *Evolutionary Programming* (EP), *Simulated Annealing* (SA), dan *Annealing-Genetic Algorithm* (AG) [2].

Pemanfaatan TSP dalam dunia nyata sangat banyak, di antaranya adalah dapat digunakan untuk proses perencanaan, logistik, *routing*, dan sebagainya. Secara tradisional para peneliti hanya berfokus terhadap *Static TSP*, akan tetapi pada

kenyataannya hal tersebut kini sudah tidak relevan lagi dengan keadaan di dunia nyata. Implementasi TSP akan menjadi lebih realistis jika lingkungannya dinamis. Bagaimanapun juga, banyak aplikasi TSP dipengaruhi oleh banyak hal, terutama oleh faktor lingkungan, sehingga kini permasalahan pada TSP tidak hanya mencari solusi global yang optimal dan efisien tetapi juga bagaimana caranya supaya bisa menangani suatu permasalahan yang dinamis [3].

Dynamic TSP (DTSP) merupakan suatu keadaan di mana permasalahan *TSP* turut serta dipengaruhi oleh keadaan lingkungannya, dapat digambarkan sebagai keadaan di mana keadaan kota yang mempunyai perubahan dalam lingkungannya, seperti contohnya macet yang sedang terjadi. Bagaimanapun juga algoritma tradisional *Ant Colony Optimization (ACO)* diusulkan untuk menyelesaikan suatu masalah optimasi stasioner tentunya mempunyai batasan dalam menangani permasalahan yang dinamis dan mungkin tidak sesuai jika digunakan untuk *Dynamic Optimization Problems (DOPs)* sehingga akan butuh beberapa penyesuaian ataupun sedikit modifikasi terhadap algoritma dasar ACO untuk dapat menyelesaikannya [4]. Hal ini disebabkan karena pada faktanya jejak *pheromone* yang dihasilkan tidak masuk akal bisa digunakan kembali pada lingkungan baru setelah terjadi perubahan. Permasalahan tersebut bisa diakali dengan cara melakukan inisialisasi ulang jejak *pheromone* dan mempertimbangkan setiap terjadinya perubahan sebagai suatu masalah baru yang harus dipecahkan dari awal, tetapi hal tersebut sangatlah tidak efisien [3].

Di dunia nyata, semut dapat menemukan jalur terpendek dari sarang ke sumber makanan dengan memanfaatkan zat dari tubuh mereka yang disebut *pheromone*, mereka juga dapat beradaptasi dengan adanya perubahan lingkungan yang terjadi dengan menggunakan zat tersebut. Semut dapat mencari jalur terpendek yang baru setelah jalur yang lama sudah tidak efektif karena bukan termasuk jalur yang paling pendek dengan cara mengikuti jalur yang mengandung *pheromone* terbanyak. Karena keunikan dari proses semut tersebut, kemudian para ilmuwan mencoba mengimplementasikannya dalam algoritma ACO. Proses/algoritma tersebut dapat dan umum digunakan untuk menyelesaikan permasalahan TSP dan diyakini mempunyai penyelesaian yang lebih baik daripada algoritma GA, EP, SA, ataupun AG [4].

Pada Tugas Akhir ini diusulkan algoritma ACO dengan menggunakan metode *Random Immigrants ACO (RIACO)* agar dapat digunakan untuk menyelesaikan permasalahan DTSP. Adapun sebagai bahan evaluasi, terdapat dua jenis permasalahan yang digunakan, yaitu permasalahan pada DTSP dengan *random traffic factor* serta DTSP dengan *cyclic traffic factor*. Adapun untuk permasalahan *random traffic factor* ini dapat dijelaskan dengan turut diikutsertakannya faktor lalu lintas yang dihasilkan secara acak pada kasus DTSP di antara kota i dan j . Sedangkan untuk variasi DTSP *cyclic traffic factor*, suatu perubahan yang terjadi dengan pola bersiklus seperti diibaratkan seperti keadaan jalan yang mempunyai pola kemacetan tertentu pada waktu pagi hari, siang hari, dan sore hingga malam hari. Dengan kata lain, lingkungan yang telah ada sebelumnya dijamin akan muncul lagi tetapi dengan adanya sedikit perubahan, hal tersebut bisa dikatakan lebih realistis daripada pola acak [4].

Hasil keluaran yang diperoleh dari implementasi metode RIACO ini dapat dijadikan sebagai bahan evaluasi dari penyelesaian suatu permasalahan DTSP.

4. RUMUSAN MASALAH

Tugas Akhir ini mengangkat beberapa rumusan masalah sebagai berikut:

- a. Melakukan perancangan desain dan analisis algoritma *Ant Colony Optimization*.
- b. Melakukan implementasi algoritma *Ant Colony Optimization*.
- c. Melakukan perancangan dan analisis metode *Random Immigrants* pada algoritma *Ant Colony Optimization*.
- d. Melakukan implementasi metode *Random Immigrants* pada algoritma *Ant Colony Optimization*.

5. BATASAN MASALAH

Tugas Akhir ini mempunyai banyak permasalahan, tetapi tidak semua permasalahan tersebut akan terselesaikan karena terdapat keterbatasan sumber daya. Maka karena itu, dibuat batasan-batasan permasalahan yang ada, antara lain batasannya sebagai berikut:

- a. Implementasi algoritma yang diusulkan menggunakan bahasa pemrograman C++.
- b. Algoritma RIACO yang digunakan untuk menyelesaikan masalah DTSP yang dihasilkan dari *dataset* TSP yang didapatkan dari TSPLIB (<http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/>). Data uji yang digunakan adalah kroA100, kroA150, kroA200, di mana data tersebut masing-masing merepresentasikan kasus permasalahan dengan skala kecil (100 kota), menengah (150 kota), dan besar (200 kota).
- c. Diberikan dua buah metode DTSP untuk dilakukan percobaan, yaitu DTSP dengan *random traffic factor* dan *cyclic traffic factor*. Untuk metode *random traffic factor* menggunakan 100 iterasi, 5 *frequency of dynamic change* (f), 0.1 dan 0.75 *magnitude of dynamic changing* (m). Sedangkan untuk metode *cyclic traffic factor* menggunakan 1000 iterasi, $f = 100$, $m = 0.1$ dan 0.75.

6. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari Tugas Akhir ini antara lain:

- a. Mempelajari modifikasi algoritma dasar ACO untuk diimplementasikan pada suatu permasalahan DTSP.
- b. Memahami penerapan algoritma RIACO untuk menyelesaikan suatu permasalahan DTSP.
- c. Mencari penyelesaian dari DTSP dengan mengimplementasikan algoritma RIACO.
- d. Melakukan evaluasi algoritma RIACO dengan cara melakukan uji coba.

7. MANFAAT TUGAS AKHIR

Manfaat yang diharapkan dari Tugas Akhir ini antara lain untuk dapat membuktikan bahwa algoritma ACO dengan menggunakan metode *Random Immigrants* bisa digunakan untuk penyelesaian suatu permasalahan dalam DTSP sehingga bisa diterapkan pada ruas jalan yang utamanya mengalami permasalahan *cyclic traffic factor* yang sesuai dengan kondisi jalanan di kota Surabaya.

8. TINJAUAN PUSTAKA

1. *Dynamic Travelling Salesman Problem (DTSP)*

Travelling Salesman Problem (TSP) merupakan salah satu contoh permasalahan yang umum dipelajari dalam *combinatorial optimization*. TSP dapat dinyatakan sebagai sebuah permasalahan dalam mencari jarak tercepat dalam sebuah perjalanan terhadap sejumlah n kota dan kota-kota yang ada hanya diperbolehkan untuk dikunjungi sekali dalam suatu kesempatan perjalanan [1].

Namun pada kenyataannya, dalam banyak aplikasi dunia nyata banyak permasalahan tentang *Dynamic Optimization Problems (DOPs)* sehingga permasalahan yang ada pada suatu perjalanan yang dilakukan tidak hanya statis, akan tetapi permasalahan dari TSP itu sendiri akan menjadi lebih realistis jika lingkungannya dinamis [4]. Sebagai contoh seorang penjual ingin mendistribusikan barang-barangnya di berbagai kota mulai dari kota tempat dia tinggal dan kembali setelah ia mengunjungi semua kota ke kota dan kembali lagi ke kota asalnya lagi. Permasalahan yang muncul ialah bagaimana mengoptimalkan waktu dan merencanakan perjalanan seefisien mungkin dengan mempertimbangkan jarak antar kota sehingga menghasilkan rute terbaik. Tetapi sulit jika mempertimbangkan keterlambatan karena lalu lintas sehingga dapat mempengaruhi rute terbaiknya. Faktor lalu lintas dapat mengubah perencanaan yang telah dilakukan sebelumnya dan si penjual akan membutuhkan alternatif rute baru yang tercepat untuk menghindari penundaan yang diakibatkan oleh lalu lintas dan mengoptimalkan waktunya lagi [4].

Berdasarkan pada gambaran di atas sudah cukup jelas perbedaan antara *Static TSP* dengan *Dynamic TSP*, terutama pada kondisi lingkungan yang berubah secara dinamis. Jika diumpamakan, jarak antara kota a , b , c , dan d mempunyai nilai tertentu, kemudian seseorang ketika ingin mendapatkan waktu perjalanan optimal untuk mengunjunginya, nilai antara kota c dan d sudah tidak sama lagi ketika orang tersebut melakukan perjalanan dari kota a ke b . Hal tersebut tentunya berbeda dengan keadaan *Static TSP* yang di mana nilai perjalanan antar kota tetap dan tidak berubah ketika ada orang yang akan melakukan perjalanan maupun ketika telah melakukan perjalanan [4].

2. *Ant Colony Optimization* (ACO)

ACO merupakan sebuah algoritma yang diadaptasi dari perilaku semut dalam menemukan jalur terpendek dari sarang ke sumber makanan. Dengan memanfaatkan zat dari tubuh mereka yang disebut *pheromone*, mereka dapat mengenali lingkungannya untuk mencari makan dan kembali ke sarangnya. Dengan zat yang ditinggalkan dari tubuh mereka pula, semut dapat beradaptasi dengan adanya perubahan lingkungan yang terjadi. Jejak *pheromone* yang ditinggalkan oleh semut akan menguap apabila jalur tersebut tidak dilalui oleh semut dalam perjalanan pulang pergi dari sarang menuju tempat makanan [2].

Seiring dengan berjalannya waktu, jejak *pheromone* yang tadinya dikeluarkan oleh semut lama kelamaan akan menguap dan hilang sehingga dapat mempengaruhi daya tarik untuk melewatinya, terutama jika sudah hilang sama sekali. Jika semut lebih cepat melewati sebuah rute dari sarangnya menuju tempat makanan dan sebaliknya maka zat *pheromone* yang menguap akan lebih sedikit sehingga rute tersebut cenderung menarik semut lain untuk melewatinya. Sebaliknya apa bila suatu rute lebih lama untuk dilakukan perjalanan pulang pergi, maka lama kelamaan zat *pheromone* akan menguap lebih banyak dan mengurangi daya tarik rute tersebut.

9. RINGKASAN ISI TUGAS AKHIR

Tugas Akhir ini akan melakukan penerapan terhadap algoritma *Ant Colony Optimization* dengan menggunakan metode *Random Immigrants* untuk menyelesaikan suatu permasalahan *Dynamic Travelling Salesman Problem* [4]. Secara garis besar, proses implementasi yang akan dilakukan pada Tugas Akhir ini ditunjukkan tahapan-tahapan yang digambarkan pada diagram alur pada Gambar 1.



Gambar 1. Bagan Proses Pengerjaan Tugas Akhir

a. Praproses data

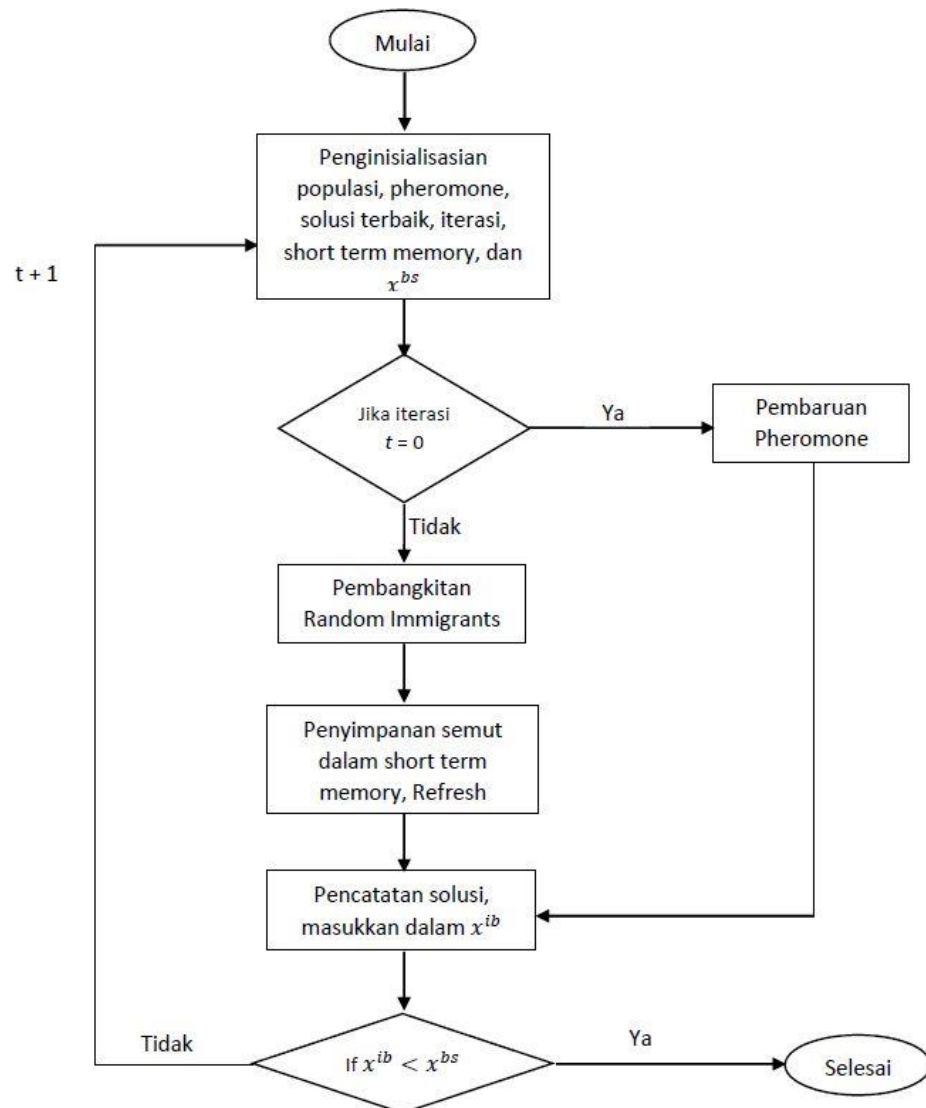
Tahap praproses data dilakukan untuk mendapatkan data uji. Data uji didapatkan dari TSPLIB [4].

b. Inisialisasi

Pada tahap ini dilakukan pemberian nilai awal dari populasi (μ), *pheromone trails* (τ_0), dan solusi kosong yang sesuai dengan referensi yang telah didapat [4].

c. Implementasi algoritma RIACO

Alur algoritma RIACO ini dapat dijelaskan dalam bagan seperti pada Gambar 2.



Gambar 2. Bagan Implementasi Algoritma RIACO

Pada algoritma RIACO sistem imigran acak dipercaya dapat menyelesaikan permasalahan DTSP karena metode tersebut mempertahankan tingkatan tertentu dari keanekaragaman dari pelaksanaannya. Pada prinsipnya yaitu memperkenalkan semut imigran baru yang dihasilkan secara acak dan mengganti semut yang terburuk pada *short term memory*, yang dinotasikan sebagai $k_{short}(t)$. Semua semut dari iterasi yang sedang terjadi mengganti yang lama, bukan hanya menggantikan yang terlama seperti pada *Population-based ACO* (P-ACO) [3].

Ketika ada semut yang dihilangkan, maka *update negative* dilakukan untuk jejak *pheromone* mereka seperti dalam Persamaan 1.

$$\tau_{ij} \leftarrow \tau_{ij} - \Delta\tau_{ij}^k, \forall (i,j) \in T \quad (1)$$

T^K merepresentasikan perjalanan dari semut ke k dan $\Delta\tau_{ij}^k = (\tau_{max} - \tau_0)/K_s$, di mana τ_{max} dan τ_0 masing-masing melambangkan nilai maksimum dan nilai *pheromone* awal. Dan jika ada semut yang ditambahkan, maka *update positif* dilakukan untuk jejak *pheromone* seperti pada Persamaan 2.

$$\tau_{ij} \leftarrow \tau_{ij} + \Delta\tau_{ij}^k, \forall (i,j) \in T \quad (2)$$

T^K dan $\Delta\tau_{ij}^k$ didefinisikan seperti pada Persamaan 1. Mekanisme ini berfungsi untuk tetap menjaga jejak *pheromone* di antara nilai tertentu dari τ_{min} , yang sama dengan nilai τ_0 , dan nilai dari τ_{max} , di mana dapat dihitung dengan menggunakan $\tau_0 + \sum_{k=1}^{K_s} \Delta\tau_{ij}^k$.

Namun, sebelum jejak *pheromone* diperbarui, satu set S_{ri} dari $r \times K_s$ imigran dihasilkan secara acak untuk menggantikan semut lain dalam $k_{short}(t)$, di mana r disebut sebagai tingkat penggantian [4].

d. Pengujian dan evaluasi

Tahap ini dilakukan pengajian dan analisis hasil yang berasal dari hasil implementasi algoritma RIACO sesuai dengan metode uji coba yang telah dijelaskan pada rumusan masalah bagian c, menggunakan *dataset* yang diambil TSPLIB [4].

10. METODOLOGI

a. Penyusunan proposal Tugas Akhir

Proposal Tugas Akhir ditulis untuk mengajukan ide atas pengerjaan Tugas Akhir. Proposal Tugas Akhir juga mengandung proyeksi hasil dari ide Tugas Akhir yang diajukan.

b. Studi literatur

Pada proses ini dilakukan studi lebih lanjut terhadap konsep-konsep yang terdapat pada jurnal, buku, artikel, dan literatur lain yang menunjang dapat menunjang pengerjaan Tugas Akhir. Studi dilakukan untuk mendalami konsep algoritma *Ant Colony*, *Random Immigrants Ant Colony Optimization*, serta konsep lain yang berguna untuk menyelesaikan permasalahan yang mungkin bisa muncul pada proses pengerjaan Tugas Akhir.

c. Implementasi algoritma

Implementasi merupakan tahapan di mana dilakukannya pembangunan sistem yang sesuai dengan algoritma. Adapun algoritma yang akan diimplementasikan yaitu algoritma *Random Immigrants Ant Colony Optimization*. Implementasinya sendiri akan dilakukan dengan menggunakan bahasa pemrograman C++.

d. Pengujian dan evaluasi

Performa dari algoritma yang telah diimplementasikan dan diterapkan akan dievaluasi menggunakan beberapa *dataset* yang didapat dari TSPLIB (<http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/>).

e. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam Tugas Akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Adapun sistematika penulisan buku Tugas Akhir secara garis besar antara lain:

1. Pendahuluan
 - a. Latar Belakang
 - b. Rumusan Masalah
 - c. Batasan Tugas Akhir
 - d. Tujuan
 - e. Metodologi
 - f. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

11. JADWAL KEGIATAN

Untuk pengerjaan Tugas Akhir, mempunyai perencanaan yang digambarkan seperti pada Tabel 1.

Table 1. Jadwal Kegiatan Pengerjaan Tugas Akhir

Tahapan	Tahun																	
	Februari				Maret				April				Mei				Juni	
Penyusunan proposal																		
Studi literatur																		
Implementasi																		
Pengujian dan evaluasi																		
Penyusunan buku																		

12. DAFTAR PUSTAKA

- [1] Johannes Marinus Kan, Yin Zhang, "Application of an Improved Ant Colony Optimization on Generalized Travelling Salesman Problem," *Energy Procedia*, vol. 17, pp. 319-325, 2012.
- [2] Marco Dorigo, Luca Maria Gambardella, "Ant Colonies for The Travelling Salesman Problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53-66, 1997.
- [3] Michalis Mavrovouniotis, Shengxiang Yang, "Ant Colony Optimization with Immigrants Schemes for Dynamic Environments," *Springer-Verlag Berlin Heidelberg*, pp. 371-380, 2010.
- [4] Michalis Mavrovouniotis, Shengxiang Yang, "Ant Colony Optimization with Immigrants Schemes for the Dynamic Travelling Salesman Problem with Traffic Factors," *Applied Soft Computing*, vol. 13, no. 10, pp. 4023-4037, 2013.