

**PENYELESAIAN KASUS *ASYMMETRIC TRAVELING SALESMAN PROBLEM* UNTUK MEMINIMASI JARAK TEMPUH DENGAN MENGGUNAKAN ALGORITMA *ELEPHANT HERDING OPTIMIZATION***

**SKRIPSI**

Diajukan untuk memenuhi salah satu syarat guna mencapai gelar  
Sarjana dalam bidang ilmu Teknik Industri

**Disusun oleh:**

**Nama : Ida Bagus Deva Ardha Nareswara Santosa**

**NPM : 2013610009**



**PROGRAM STUDI TEKNIK INDUSTRI  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS KATOLIK PARAHYANGAN**

**2017**

**FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS KATOLIK PARAHYANGAN  
BANDUNG**



Nama : Ida Bagus Deva Ardha Nareswara Santosa  
NPM : 2013610009  
Program Studi : Teknik Industri  
Judul Skripsi : *PENYELESAIAN ASYMMETRIC TRAVELLING  
SALESMAN PROBLEM UNTUK MEMINIMASI JARAK  
TEMPUH DENGAN MENGGUNAKAN ALGORITMA  
ELEPHANT HERDING OPTIMIZATION*

**TANDA PERSETUJUAN SKRIPSI**

Bandung, 7 Juli 2017

**Ketua Program Studi Teknik Industri**

(Dr. Carles Sitompul, S.T., M.T., M.I.M.)

**Pembimbing**

(Dr. Carles Sitompul, S.T., M.T., M.I.M.)





Program Studi Teknik Industri  
Fakultas Teknologi Industri  
Universitas Katolik Parahyangan



### **Pernyataan Tidak Mencontek atau Melakukan Tindakan Plagiat**

Saya, yang bertanda tangan  
dibawah ini,

Nama : Ida Bagus Deva Ardha Nareswara Santosa  
NPM : 2013610009

dengan ini menyatakan bahwa Skripsi dengan judul :

*"Penyelesaian Asymmetric Travelling Salesman Problem Untuk Meminimasi Jarak Tempuh Dengan Menggunakan Algoritma Elephant Herding Optimization"*

adalah hasil pekerjaan saya dan seluruh ide, pendapat atau materi dari sumber lain telah dikutip dengan cara penulisan referensi yang sesuai.

Pernyataan ini saya buat dengan sebenar-benarnya dan jika pernyataan ini tidak sesuai dengan kenyataan, maka saya bersedia menanggung sanksi yang akan dikenakan kepada saya.

Bandung, 7 Juli 2017

Ida Bagus Deva Ardha Nareswara Santosa  
NPM : 2013610009

## ABSTRAK

*Asymmetric Traveling Salesman Problem* (ATSP) merupakan masalah pencarian rute untuk mengunjungi seluruh kota yang perlu ia kunjungi dengan jarak terpendek. Pada permasalahan ATSP, seorang *salesman* hanya boleh mengunjungi masing-masing kota yang ada satu kali dan kembali ke kota pertama setelah seluruh kota telah dikunjungi. Persoalan ATSP ini merupakan variasi dari persoalan *Traveling Salesman Problem* yang memiliki perbedaan berupa matriks jarak antarkota yang asimetrik. Asimetrik berarti jarak dari kota X menuju kota Y dapat berbeda dengan jarak dari kota Y menuju kota X.

Dalam penelitian ini, persoalan ATSP diselesaikan dengan menggunakan *Elephant Herding Optimization* (EHO). EHO merupakan algoritma metaheuristik yang terinspirasi dari bagaimana sebuah populasi gajah berkumpul dan bergerak. Arah dan besar pergerakan gajah dalam satu *clan* dipengaruhi oleh seorang gajah tertua yang disebut *Matriarch* dan seluruh posisi gajah yang ada di dalam *clan* tersebut. Setiap periode tertentu gajah laki-laki termuda pada *clan* akan pergi untuk hidup sendiri. Terdapat 3 buah parameter pada EHO, yaitu *Alfa* yang menunjukkan pengaruh dari *Matriarch*, *Beta* yang menunjukkan pengaruh dari posisi gajah, dan *Clan* yang menunjukkan berapa gajah setiap periode yang akan meninggalkan populasi.

Dalam penelitian ini, EHO telah dirancang untuk dapat menyelesaikan kasus ATSP dan diimplementasikan pada 5 buah kasus *benchmark* dengan menggunakan 8 kombinasi parameter yang berbeda-beda. Perbandingan EHO dilakukan dengan *New Genetic Algorithm* (NGA), *Improved Discrete Bat Algorithm* (IDBA), dan *Harmony Search Algorithm* (HSA). Kasus dengan jumlah kota 17 hingga 56 menunjukkan hasil yang mencapai solusi optimal, sedangkan untuk kasus yang memiliki jumlah kota 71, algoritma NGA menghasilkan solusi yang lebih baik dari EHO. Semua parameter diuji pengaruhnya dan interaksinya dengan menggunakan ANOVA *multifactor*. Hasil dari pengujian pengaruh parameter tersebut menunjukkan adanya pengaruh dari interaksi parameter pada semua kasus yang terdapat pada penelitian ini.

## **ABSTRACT**

*Asymmetric Traveling Salesman Problem (ATSP) is a problem of finding a route for a salesman to visit all the cities that need to be visited with a minimum total distance. In ATSP, a salesman may only visit each city once and he will return to the first city after the entire city has been visited. ATSP is a variation of the Traveling Salesman Problem (TSP) which has differences of an ATSP's distance matrix is asymmetric. Asymmetric means that the distance between city X to city Y is different from the distance between city Y to city X.*

*In this research, ATSP is solved by using Elephant Herding Optimization(EHO). EHO is a metaheuristic algorithm that is inspired by how a population of elephants gather and move. Direction and magnitude of the movements of elephants in a clan is influenced by the oldest elephant called Matriarch and all positions of the elephant within the clan. Any given period, the youngest male elephant will leave the clan to live solitary. There are 3 parameters in EHO, Alfa shows the magnitude of influence given by Matriarch, Beta shows the magnitude of influence given by all of the elephants' position, and Clan shows how many young male elephant in any given period will leave the clan.*

*In this research, EHO has been designed to solve ATSP and implemented in 5 benchmark case using 8 different combinations of parameters. EHO will be compared with the New Genetic Algorithm (NGA), Improved Discrete Bat Algorithm (IDBA), and Harmony Search Algorithm (HSA). For the case with 17 to 56 cities, EHO reached the optimal result, while cases with 71 cities, NGA produces better solutions than EHO. All parameters are tested in order to find influence effect and interaction using multifactor ANOVA. The results of testing the effect of these parameters indicate that the influence of interaction parameter in all cases found in this research.*

## KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas berkat, rahmat, dan kekuatan yang diberikan untuk menyelesaikan laporan skripsi dengan judul *Penyelesaian Asymmetric Travelling Salesman Problem* untuk Meminimasi Jarak Tempuh dengan Menggunakan Algoritma *Elephant Herding Optimization*. Selama pelaksanaan dan proses penyusunan laporan ini, penulis mendapatkan banyak pengalaman dan bimbingan dari berbagai pihak. Pada kesempatan ini, penulis ingin mengucapkan terima kasih kepada pihak-pihak yang telah memberikan dukungan dan bantuan secara langsung maupun tidak langsung, yaitu;

1. Orang Tua penulis yang telah memberikan dukungan dan motivasi.
2. Bapak Dr. Carles Sitompul, S.T., M.T., M.i.M sebagai dosen pembimbing yang telah memberikan motivasi, masukan, dan bimbingan yang penuh kesabaran dalam penulisan laporan skripsi ini.
3. Ibu Cynthia Prithadevi Juwono, Ir., M.S. sebagai dosen penguji proposal skripsi yang telah memberikan masukan dan saran terhadap proposal skripsi.
4. Bapak Fran Setiawan S.T., M.Sc. sebagai dosen penguji proposal skripsi yang telah memberikan masukan dan saran terhadap proposal skripsi.
5. Petrus Kevin, Adrianus Vincent, Arnold Raharja, Ricky Nugraha Tendi, Agnes Monica, Felix Arya, Marcelinus Rico, dan Christian Setiardo yang telah saling mendukung sesama topik algoritma, memberikan saran serta masukan, melatih simulasi seminar, dan membantu mencari referensi-referensi yang berkaitan baik langsung maupun tidak langsung dalam pembuatan skripsi ini.
6. Ranggi Maharani sebagai orang yang telah bersedia mendampingi, mendengarkan, mendukung, dan memotivasi dalam pembuatan skripsi ini.
7. Hans Junius, Adrian Hartanto, Ardianto Mahadi, Vincentius Nugraha, Ivan Reynaldi, Bonifasius Alvin, dan Jong William yang telah membantu penulis dalam hal simulasi seminar dan memberikan masukan-masukan selama pembuatan.
8. Seluruh teman-teman kelas A.
9. Seluruh teman-teman TI 2013.

10. Semua dosen yang mengajar di TI UNPAR.
11. Semua staff dan pekarya UNPAR khususnya di FTI.
12. Semua pihak lain yang terlibat dalam pembuatan laporan skripsi yang tidak dapat disebutkan satu per satu.

Penulis menyadari keterbatasan yang menyebabkan masih terdapat ketidaksempurnaan dalam penulisan laporan ini. Untuk itu penulis dengan hati terbuka menerima kritik dan saran yang dapat digunakan sebagai perbaikan laporan ini. Semoga laporan ini dapat berguna bagi pembacanya.

Bandung, 9 Juni 2017

Ida Bagus Deva Ardha Nareswara Santosa

## DAFTAR ISI

<b>ABSTRAK</b> .....	i
<b>ABSTRACT</b> .....	ii
<b>KATA PENGANTAR</b> .....	iii
<b>DAFTAR ISI</b> .....	v
<b>DAFTAR TABEL</b> .....	vii
<b>DAFTAR GAMBAR</b> .....	ix
<b>DAFTAR LAMPIRAN</b> .....	xi
<b>BAB I PENDAHULUAN</b> .....	I-1
I.1 Latar Belakang Masalah.....	I-1
I.2 Identifikasi dan Perumusan Masalah .....	I-2
I.3 Batasan Masalah .....	I-7
I.4 Tujuan Penelitian .....	I-7
I.5 Manfaat Penelitian.....	I-8
I.6 Metodologi Penelitian .....	I-8
<b>BAB II TINJAUAN PUSTAKA</b> .....	II-1
II.1 <i>Asymmetric Traveling Salesman Problem</i> .....	II-1
II.2 Metode Pembuatan Urutan Kota ( <i>Tour Construction</i> ).....	II-2
II.2.1 Metode <i>Exact</i> .....	II-2
II.2.2 Metode <i>Heuristic</i> .....	II-2
II.2.3 Metode <i>Metaheuristic</i> .....	II-3
II.2.3.1 <i>Elephant Herding Optimization</i> .....	II-3
II.3 <i>Encoding dan Decoding</i> .....	II-6
II.4 <i>Local Search Algorithm 2-Opt</i> .....	II-6
II.4 Perancangan Eksperimen.....	II-7
<b>BAB III PERANCANGAN ALGORITMA</b> .....	III-1
III.1 <i>Encoding dan Decoding</i> .....	III-1
III.2 Perancangan Algoritma.....	III-4
III.2.1 Notasi Algoritma .....	III-5
III.2.2 Algoritma Utama EHO .....	III-6
III.2.3 Algoritma Perhitungan Pembagi (Algoritma A) .....	III-8



III.2.4	Algoritma Penentuan Posisi Awal Gajah (Algoritma B) .....	III-9
III.2.5	Algoritma <i>Clan Updating Operator</i> .....	III-10
III.2.6	Algoritma <i>Clan Separating Operator</i> .....	III-17
III.2.7	Algoritma Pengisian Urutan (Algoritma C) .....	III-25
III.2.8	Algoritma Perbandingan (Algoritma E) .....	III-27
III.2.9	Algoritma Perhitungan Nilai Jarak (Algoritma G) .....	III-29
III.2.10	Algoritma 2-Opt .....	III-32
III.3	Verifikasi dan Validasi Algoritma .....	III-30
<b>BAB IV</b>	<b>IMPLEMENTASI ALGORITMA</b> .....	IV-1
IV.1	Verifikasi dan Validasi Program Komputer .....	IV-1
IV.2	Implementasi <i>Elephant Herding Optimization</i> (EHO) .....	IV-4
IV.2.1	Penentuan Parameter <i>Elephant Herding Optimization</i> (EHO) .....	IV-4
IV.2.2	Penerapan EHO pada Kasus <i>Benchmark</i> .....	IV-12
IV.3	Pengujian Parameter <i>Elephant Herding Optimization</i> .....	IV-16
IV.4	<i>Interaction Plot</i> (IP) .....	IV-20
IV.5	Uji Beda ( <i>Tukey Test</i> ) .....	IV-22
IV.6	Perbandingan <i>Elephant Herding Optimization</i> dengan <i>New Genetic Algorithm, Improved Discrete Bat Algorithm, dan Harmony Search Algorithm</i> .....	IV-24
<b>BAB V</b>	<b>ANALISIS</b> .....	V-1
V.1	Analisis <i>Encoding</i> dan <i>Decoding</i> .....	V-1
V.2	Analisis <i>Local Search Algorithm</i> .....	V-2
V.3	Analisis Parameter EHO .....	V-4
V.3.1	Analisis Parameter <i>Alfa</i> .....	V-4
V.3.2	Analisis Parameter <i>Beta</i> .....	V-5
V.3.3	Analisis Parameter Jumlah <i>Clan</i> .....	V-5
V.3.4	Analisis Interaksi Pengaruh Parameter .....	V-6
<b>BAB VI</b>	<b>KESIMPULAN DAN SARAN</b> .....	VI-1
V.1	Kesimpulan .....	VI-1
V.2	Saran .....	VI-2
<b>DAFTAR PUSTAKA</b>		
<b>LAMPIRAN</b>		

## DAFTAR TABEL

Tabel II.1	Tabel Perhitungan ANOVA.....	II-10
Tabel III.1	Contoh Hasil Pengacakan untuk 4 Kota pada Gajah 1 .....	III-1
Tabel III.2	<i>From/To</i> Matrix pada Persoalan <i>Asymmetric Travelling Salesman Problem</i> .....	III-2
Tabel III.3	Contoh Hasil Pengacakan Bilangan Sebagai Dasar untuk Solusi Awal .....	III-3
Tabel III.4	Hasil Pengurutan Bilangan Acak beserta Urutan Kota yang Baru .....	
Tabel III.5	Perhitungan Jarak.....	III-4
Tabel III.6	Matriks Jarak.....	III-33
Tabel III.7	Hasil Random pada Tabel CS .....	III-34
Tabel III.8	Hasil Random pada Tabel RKey .....	III-34
Tabel III.9	<i>Assign</i> Nilai 1 Hingga Jumlah Kota pada Tabel Urut .....	III-35
Tabel III.10	Tabel Urutan dan RKey CUO .....	III-36
Tabel III.11	Tabel CU Setelah CUO .....	III-40
Tabel III.12	Tabel RKey pada AP (1) .....	III-40
Tabel III.13	Tabel Urutan dan RKey pada AP .....	III-41
Tabel III.14	Tabel FV pada AP.....	III-42
Tabel III.15	Tabel RKey pada 2-Opt (1).....	III-43
Tabel III.16	Tabel Urutan Kota pada 2-Opt (1) .....	III-43
Tabel III.17	Tabel FV pada 2-Opt (1) .....	III-43
Tabel III.18	Tabel RKey pada 2-Opt (2).....	III-44
Tabel III.19	Tabel Urutan Kota pada 2-Opt (2) .....	III-44
Tabel III.20	Tabel FV pada 2-Opt (2) .....	III-44
Tabel III.21	Tabel RKey pada 2-Opt (3).....	III-45
Tabel III.22	Tabel Urutan Kota pada 2-Opt (3) .....	III-45
Tabel III.23	Tabel FV pada 2-Opt (3) .....	III-45
Tabel III.24	Tabel RKey pada 2-Opt (4).....	III-45
Tabel III.25	Tabel Urutan Kota pada CS.....	III-46
Tabel III.26	Tabel RKey pada CS .....	III-51
Tabel III.27	Tabel RKey pada AP (2) .....	III-51
Tabel III.28	Tabel RKey Hasil .....	III-51

Tabel III.29	Tabel RKey dan Interpretasinya.....	III-52
Tabel IV.1	Data Kasus ATSP untuk 4 kota.....	IV-2
Tabel IV.2	Rekapitulasi Kasus ATSP .....	IV-4
Tabel IV.3	Rekapitulasi Hasil Jumlah Iterasi yang Terbaik untuk Setiap Kasus ..	
Tabel IV.4	Rekapitulasi Hasil Jumlah Iterasi dan Gajah yang Terbaik untuk Setiap Kasus.....	IV-10
Tabel IV.5	Rekapitulasi Kombinasi Parameter dengan <i>Orthogonal Array</i> L8.....	
Tabel IV.6	Rekapitulasi Hasil Kasus BR17.....	IV-12
Tabel IV.7	Rekapitulasi Hasil Kasus FTV33.....	IV-13
Tabel IV.8	Rekapitulasi Hasil Kasus FTV44.....	IV-13
Tabel IV.9	Rekapitulasi Hasil Kasus FTV55.....	IV-14
Tabel IV.10	Rekapitulasi Hasil Kasus FTV70.....	IV-15
Tabel IV.11	Rekapitulasi Pengujian ANOVA.....	IV-19
Tabel IV.12	Rekapitulasi Nilai Parameter Terbaik.....	IV-23
Tabel IV.13	Perbandingan Hasil Solusi Terbaik Kasus ATSP .....	IV-24
Tabel V.1	Perbandingan Antara EHO tanpa 2-Opt, EHO dengan 2-Opt, dan EHO dengan <i>Modified</i> 2-Opt.....	V-3

## DAFTAR GAMBAR

Gambar I.1	<i>Flowchart</i> Metodologi Penelitian .....	I-10
Gambar II.1	Operasi 2-Opt.....	II-6
Gambar III.1	Sistem Pengurutan Kota pada <i>Random-Key Representation</i> .....	III-2
Gambar III.2	Algoritma Utama EHO .....	III-7
Gambar III.3	Algoritma Perhitungan Pembagi .....	III-9
Gambar III.4	Algoritma Penentuan Posisi Awal Gajah .....	III-10
Gambar III.5	Algoritma <i>Clan Updating Operator</i> .....	III-14
Gambar III.6	Algoritma <i>Clan Separating Operator</i> .....	III-22
Gambar III.7	Algoritma Perhitungan Jarak untuk <i>Clan Updating</i> .....	III-27
Gambar III.8	Algoritma Perbandingan .....	III-28
Gambar III.9	Algoritma Perhitungan Nilai Jarak.....	III-29
Gambar III.10	Algoritma 2-Opt .....	III-32
Gambar IV.1	<i>Input</i> serta <i>Output Program</i> EHO.....	IV-3
Gambar IV.2	Plot Jumlah Iterasi untuk Kasus BR17.....	IV-5
Gambar IV.3	Plot Jumlah Iterasi untuk Kasus FTV33.....	IV-5
Gambar IV.4	Plot Jumlah Iterasi untuk Kasus FTV44.....	IV-6
Gambar IV.5	Plot Jumlah Iterasi untuk Kasus FTV55.....	IV-6
Gambar IV.6	Plot Jumlah Iterasi untuk Kasus FTV70.....	IV-7
Gambar IV.7	Plot Jumlah Gajah untuk Kasus BR17.....	IV-8
Gambar IV.8	Plot Jumlah Gajah untuk Kasus FTV33.....	IV-8
Gambar IV.9	Plot Jumlah Gajah untuk Kasus FTV44.....	IV-9
Gambar IV.10	Plot Jumlah Gajah untuk Kasus FTV55.....	IV-9
Gambar IV.11	Plot Jumlah Gajah untuk Kasus FTV70.....	IV-10
Gambar IV.12	Hasil Uji ANOVA untuk Kasus BR17 .....	IV-17
Gambar IV.13	Hasil Uji ANOVA untuk Kasus FTV33.....	IV-17
Gambar IV.14	Hasil Uji ANOVA untuk Kasus FTV44 .....	IV-18
Gambar IV.15	Hasil Uji ANOVA untuk Kasus FTV55.....	IV-18
Gambar IV.16	Hasil Uji ANOVA untuk Kasus FTV70.....	IV-19
Gambar IV.17	<i>Interaction Plot</i> Kasus BR17 .....	IV-20
Gambar IV.18	<i>Interaction Plot</i> Kasus FTV33 .....	IV-21

Gambar IV.19 <i>Interaction Plot</i> Kasus FTV44 .....	IV-21
Gambar IV.20 <i>Interaction Plot</i> Kasus FTV70 .....	IV-22
Gambar IV.21 <i>Tukey Test</i> pada Kasus FTV70 .....	IV-23

## DAFTAR LAMPIRAN

LAMPIRAN A: *Benchmark* Kasus ATSP



# BAB I

## PENDAHULUAN

Bagian pertama dari penelitian ini berisi tentang pendahuluan mengenai penelitian dalam menerapkan *Elephant Herding Optimization* (EHO) untuk menyelesaikan permasalahan *Asymmetric Travelling Salesman Problem* (ATSP). Pada bab ini terdapat beberapa bagian, yaitu latar belakang masalah, identifikasi dan rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan diakhiri dengan metodologi penelitian.

### I.1 Latar Belakang Masalah

Dalam kehidupan seorang *salesman* yang setiap hari berjalan mengunjungi tempat-tempat yang telah ditentukan kantor pusat untuk penawaran produk yang dibuat kepada konsumen, pemilihan destinasi untuk tempat penawaran produk adalah suatu masalah yang setiap hari harus diputuskan *salesman*. Keputusan akan pemilihan destinasi inilah menentukan *salesman* harus berjalan sejauh apa untuk setiap tempat yang harus dilalui. Semua tempat yang diberikan oleh kantor pusat harus ditempuh oleh *salesman* tersebut dan *salesman* harus kembali ke kantor pusat setelah melalui semua tempat yang ditentukan untuk mengirimkan laporan penjualan. Oleh karena itu, *salesman* harus menentukan lokasi yang memiliki total jarak terpendek untuk dapat mendatangi setiap tempat dalam satu kali jalan. Permasalahan dalam penentuan lokasi yang dikunjungi dapat dimodelkan menjadi kasus optimasi kombinatorial yang sering disebut *Travelling Salesman Problem*.

*Travelling Salesman Problem* (TSP) pertama kali di formulasikan secara matematis pada tahun 1800-an oleh Sir William Rowan Hamilton, seorang ahli matematika yang berasal dari Irlandia dan Thomas Kirkman, seorang ahli matematika juga yang berasal dari Inggris. Seiring berjalannya waktu, persoalan TSP lalu dikembangkan oleh Lawrer, Lenstra, Kan, dan Shmoys pada tahun 1985 (Lawrer, Lenstra, Kan, & Shmoys, 1985) dan dikembangkan kedalam jenis-jenis permasalahan yang berbeda-beda. Permasalahan TSP memiliki beberapa jenis, yaitu *symmetric* TSP atau disingkat TSP (persoalan TSP pada umumnya),

*Asymmetric TSP* atau disingkat *ATSP* (Gutin & Punnen, 2002), *Sequential Ordering Problem*, *Travelling Salesman Problem with Time Window*, dan lain-lain.

*Travelling Salesman Problem* merupakan suatu problem matematika berjenis diskrit dimana kasus ini menganalogikan seorang *salesman* yang harus mengunjungi setiap lokasi yang ditentukan dalam pekerjaannya. Setiap lokasi memiliki jarak antar lokasi dalam menempuh perjalanan menuju lokasi tersebut. Oleh karena itu, tujuan yang ingin *salesman* capai adalah menempuh perjalanan yang merupakan jarak terpendek yang ada. Contoh lain yang memiliki jenis masalah yang sama dengan *travelling salesman problem* adalah misalkan ada seorang kurir yang bekerja di perusahaan jasa pengiriman barang yang ingin melakukan pengiriman lima paket dari kantor pusat ke setiap lokasi yang ada. Setiap lokasi yang ada memiliki jarak antar lokasi masing-masing dan dalam pengiriman paket tersebut, kurir diharuskan memilih urutan lokasi yang memiliki total jarak terpendek. Urutan pemilihan lokasi yang ada dapat dimodelkan dalam *Travelling Salesman Problem*.

Pada penelitian ini, permasalahan yang digunakan untuk penerapan EHO adalah salah satu cabang permasalahan yang terdapat pada *Travelling Salesman Problem*, yaitu *Asymmetric Travelling Salesman Problem*. Sebagai contoh nyata yang ada, jarak dari satu kota A ke kota B tidak selalu sama dengan jarak dari kota B ke kota A. Perbedaan ini umumnya terjadi dikarenakan jalur antar kota tersebut hanya memiliki satu arah. Oleh karena itu, permasalahan *Asymmetric Travelling Salesman Problem* diharapkan untuk dapat memetakan kondisi yang lebih nyata jika dibandingkan dengan *Travelling Salesman Problem*.

*Asymmetric Travelling Salesman Problem* merupakan permasalahan *NP-Hard (Non-deterministic Polynomial-time Hard) problem* yang membutuhkan penyelesaian dengan banyak solusi (Ascheuer, Grotschel, & Abdel-Hamid, 1999) karena saat permasalahan *ATSP* diselesaikan dengan metode eksak, maka waktu yang dibutuhkan untuk menghasilkan solusi optimal sangat besar. Oleh karena itu, dibutuhkan pendekatan heuristik maupun metaheuristik untuk dapat menyelesaikan kasus *ATSP* ini.

## I.2 Identifikasi dan Rumusan Masalah

Asymmetric Travelling Salesman Problem (ATSP) adalah salah satu permasalahan TSP yang sering dijumpai di dunia nyata. Dimana permasalahan yang ada pada ATSP sama halnya dengan TSP pada umumnya, hanya saja jarak setiap tempat tidak berlaku untuk arah yang sebaliknya (jarak yang ditempuh *salesman* saat *salesman* ingin berjalan dari kota A menuju kota B tidak sama dengan jarak yang ditempuh *salesman* saat ingin berjalan dari kota B menuju kota A). Sebagai ilustrasi dari ATSP dalam dunia nyata adalah permasalahan yang dialami oleh kurir pengantar barang saat jalur yang harus hanya memiliki jalur satu arah. Tersebutlah suatu kabupaten yang ada di Jawa Barat, terdapat beberapa tempat yang harus dikunjungi oleh seorang kurir. Kurir tersebut hanya mengunjungi tempat-tempat tersebut sekali. Pada kenyataannya, tidak semua jarak antara kota  $i$  ke kota  $j$  dan jarak dari kota  $j$  ke kota  $i$  sama. Jarak antar kota tersebut ada yang berbeda karena jalur yang harus dilalui berbeda atau hanya ada jalur satu arah di kota tersebut. Berdasarkan persoalan kurir yang telah dijabarkan, permasalahan transportasi yang terjadi di dunia nyata ternyata lebih kompleks jika dibandingkan dengan persoalan TSP pada umumnya. Oleh karena itu, persoalan ATSP lebih mampu untuk merepresentasikan problem dunia nyata jika dibandingkan dengan TSP.

Dalam mencari solusi optimal dari permasalahan ATSP yang ada, terdapat dua metode yang dapat digunakan. Metode pertama adalah metode eksak dimana hasil dari metode ini adalah solusi optimal dari permasalahan ATSP, sedangkan metode kedua adalah metode heuristik/metaheuristik (pendekatan). Contoh metode eksak yang dapat digunakan untuk menyelesaikan permasalahan ATSP adalah *Branch and Bound* (Gutin & Punnen, 2002). Kekurangan dari metode eksak adalah waktu penyelesaian masalah yang sangat lama jika *node* (misalkan kota yang harus dilalui) dari soal ATSP semakin bertambah. Gutin dan Punnen (2002) sendiri pernah menerapkan metode heuristik untuk menyelesaikan permasalahan ATSP (*Nearest Neighbor Algorithm* dan *Greedy Algorithm*) tetapi menghasilkan solusi yang kurang baik jika dibandingkan dengan metode eksak *Branch and Bound*.

Seperti dikatakan sebelumnya, permasalahan ATSP merupakan salah satu permasalahan yang ada pada kategori *NP-Hard* dan untuk mendapatkan satu penyelesaian masalah yang optimal dibutuhkan waktu yang sangat lama,

oleh karena itu metode heuristik atau metaheuristik cocok untuk permasalahan ATSP meskipun metode ini tidak menjamin solusi yang didapat merupakan optimal tetapi mendekati optimal. Metode heuristik merupakan metode yang menitikberatkan pencarian solusi pada prinsip *trial and error*. Oleh karena itu, ambil contoh permasalahan ATSP, jika *node* atau kota yang harus dilewati semakin banyak maka semakin sulit juga untuk sebuah metode dalam mencari setiap kombinasi solusi yang ada karena *search space* yang dimiliki semakin besar. Sedangkan untuk metode metaheuristik, metode ini mencari kombinasi solusi dengan dua proses utama, yaitu *Exploration* dan *Exploitation*. *Exploration* adalah suatu proses mencari solusi-solusi yang ada di ruang lingkup masalah, sedangkan *Exploitation* adalah proses seleksi dari pencarian solusi yang ada (Eiben & Schippers, 1998). Jika dibandingkan, metode heuristik merupakan *Problem-dependent Technique* dimana pada umumnya, metode heuristik hanya bisa menyelesaikan satu kasus yang sama (seperti contoh *Dijkstra Algorithm* umumnya hanya bisa menyelesaikan kasus *Shortest Path*) dan kadang terjebak pada *local optimum*. Sedangkan di sisi lain, metode *metaheuristic* merupakan *Problem-independent Technique* dimana *metaheuristic* dapat dijadikan *blackbox* dengan *output* yang dapat berbeda jika *input* yang diberikan sama (tidak tergantung pada problem yang ingin diselesaikan). *Meta (beyond)* dapat diartikan lebih dari sekedar mendapatkan solusi. Metaheuristik bertujuan untuk mengeksplorasi ruang solusi untuk mendapatkan solusi terbaik.

Penggunaan metode metaheuristik diharapkan mendapatkan solusi yang mendekati optimal dengan waktu yang lebih singkat jika dibandingkan dengan metode eksak (Gemm, Kim, dan Loganathan, 2001). Ilmuan-ilmuan pada tahun 1970, sudah mulai mengembangkan metode metaheuristik untuk dapat menyelesaikan permasalahan-permasalahan yang ada lebih cepat meskipun hasil yang didapat tidak optimal.

Untuk permasalahan ATSP sendiri, sudah ada beberapa *metaheuristic algorithm* yang mencoba menyelesaikan permasalahan ATSP. Algoritma tersebut ada *A New Genetic Algorithm* (Nagata dan Soler, 2012), *Improved Discrete Bat Algorithm* (Osaba, Yang, Diaz, Garcia, & Carballedo, 2016), dan *Harmony Search Algorithm* (Kevin, 2016). Solusi yang dihasilkan oleh ketiga algoritma tersebut sudah baik, tetapi penyelesaian permasalahan ATSP sendiri tidak terbatas oleh metode-metode yang sudah ada. Jika sebuah algoritma sudah

masuk kedalam algoritma metaheuristik, maka algoritma tersebut dapat dikatakan mampu untuk menghasilkan solusi dari setiap permasalahan yang ada.

*Elephant Herding Optimization* atau disingkat EHO diperkenalkan oleh Gai-Ge Wang, Suash Deb, dan Leandro Coelho pada bulan Desember tahun 2015. EHO ini adalah algoritma yang masih tergolong baru sehingga belum banyak permasalahan-permasalahan yang diselesaikan oleh algoritma ini. EHO merupakan sebuah algoritma metaheuristik yang dikembangkan dengan meniru tingkah laku dari hewan gajah (*Elephant*) dalam bermigrasi berbentuk kelompok dari satu tempat ke tempat lain (*Herding*). Kelompok dari gajah-gajah ini digambarkan sebagai *clan* yang memiliki satu ketua/petinggi yang disebut *matriarch* yang pada umumnya adalah gajah yang tertua, dari setiap kelompok gajah akan menghasilkan gajah berjenis kelamin laki-laki yang akan meninggalkan kelompok saat masuk remaja, dan gajah yang meninggalkan kelompok akan digantikan dengan yang baru. Dalam penerapan EHO, nilai *Global optimum* adalah suatu nilai yang dicari. *Local Optimum* pada penerapan EHO dianalogikan sebagai posisi terbaik yang dimiliki oleh *matriach* pada sebuah clan yang ada dan diharapkan pada iterasi terakhir nilai *Local Optimum* yang dihasilkan sama dengan *Global Optimum* yang ada. Untuk mencapai posisi terbaik, setiap gajah yang ada di dalam sebuah clan akan bergerak mengikuti kemana gajah *matriach* berjalan dan terdapat gajah yang akan keluar dari *clan* yang ada setiap periodenya. Jumlah pergerakan dari setiap clan yang ada dapat dianalogikan sebagai jumlah iterasi dan jumlah terjadinya gajah yang keluar dari *clan* mewakili jumlah generasi yang ditetapkan. Proses pergerakan posisi dari gajah (baik pergerakan yang dikarenakan *matriarch* ini bertujuan untuk mendapatkan *Global optimum* dari permasalahan yang ada (Wang *et. al.*, 2015).

Gai-Ge Wang, Suash Deb, dan Leandro Coelho menjelaskan tentang dua proses yang menjadi dasar EHO, yaitu *Clan Updating Operator* dan *Separating Operator*. *Clan Updating Operator* adalah proses yang dilakukan untuk meng-*update* posisi *matriach* serta gajah (baik itu *matriach* atau bukan) yang berada di tengah2 *clan*, sedangkan *Separating Operator* adalah proses dimana gajah berjenis kelamin laki-laki akan meninggalkan *clan* saat remaja. Parameter Jumlah *clan* (*clan*) adalah parameter penentu berapa banyak gajah yang akan keluar dan digantikan pada setiap periode. Proses pemindahan posisi dari sebuah *clan* direpresentasikan oleh *influence* yang dapat diberikan oleh

gajah *matriach*, sedangkan posisi gajah *matriarch* sendiri ditentukan oleh posisi seluruh gajah dari clan tersebut. Oleh karena itu, *influence* dari gajah *matriach* ( $\alpha$ ) dan *influence* dari posisi gajah yang ada didalam *clan* ( $\beta$ ) adalah dua parameter yang menentukan hasil dari proses *Clan Updating Operator*.

*Elephant Herding Optimization* telah dilakukan *benchmark* dengan beberapa algoritma metaheuristik (*Biogeography-based Optimization*, *Differential Evolution Algorithm*, dan *Genetic Algorithm*) (Wang *et. al.*, 2015) pada *test function* atau *artificial landscapes*. *Test functions* ini digunakan untuk mengevaluasi karakteristik dari sebuah *optimization algorithm* berdasarkan kecepatan konvergensi, ketepatan, *robustness*, dan performansi secara *general* (Wang *et. al.*, 2015). Salah satu *benchmark* yang digunakan sebagai perbandingan EHO adalah *Genetic Algorithm* (GA). *Genetic Algorithm* pernah diterapkan untuk menyelesaikan kasus *Asymmetric Travelling Salesman Problem*. Berikut adalah Tabel I.1 hasil perbandingan EHO terhadap *benchmark* yang ada termasuk GA.

Tabel I.1 Perbandingan hasil EHO terhadap *benchmark* (Sumber: Wang *et. al.*, 2015)

<i>Test Function</i>	BBO	DE	EHO	GA
Ackley	4.08	15.82	<b>1.30E-03</b>	13.76
Alpine	0.37	9.97	<b>1.80E-04</b>	12.69
Brown	<b>2.20E-16</b>	1.63	2.30E-06	<b>2.20E-16</b>
Dixon & Price	4.80E+04	8.70E+05	<b>0.74</b>	9.30E+05
<i>Fletcher-Powell</i>	<b>2.60E+04</b>	1.30E+05	2.50E+05	8.50E+04
Griewank	3.52	13.09	<b>1</b>	10.27
Holzman 2	60	475.88	<b>7.60E-14</b>	1.60E+03
Ley	<b>0.4</b>	7.17	1.27	12.41
Pathological	4.34	<b>1.46</b>	3.2	4.07
Penalty #1	3.26	32.71	<b>0.23</b>	13.9
Penalty #2	118.5	2.60E+05	<b>1.33</b>	2.20E+05
Perm	6.00E+51	<b>5.80E+37</b>	4.00E+45	6.00E+51
Powell	23	334.18	<b>8.80E-07</b>	191
Quartic	<b>2.20E-16</b>	0.18	8.40E-16	<b>2.20E-16</b>
Rastrigin	<b>2.20E-16</b>	79	1.80E-05	9

Penerapan EHO pada beberapa *test function* membuktikan bahwa EHO menghasilkan hasil yang lebih baik daripada GA. Dapat dilihat pada tabel diatas, EHO berhasil lebih unggul pada 10 *test function* yang ada jika dibandingkan dengan *Genetic Algorithm* yang pernah menyelesaikan kasus *Asymmetric*



*Travelling Salesman Problem*. Hal ini memungkinkan untuk EHO dapat menyelesaikan *Asymmetric Travelling Salesman Problem* dan memiliki kemungkinan untuk dapat menghasilkan performansi yang bagus juga.

Metode metaheuristik yang digunakan dalam penelitian ini adalah *Elephant Herding Optimization* (EHO). Wang et. al. (2015) mengatakan bahwa algoritma ini dirancang untuk menyelesaikan berbagai masalah dengan jenis *global optimization* untuk menghasilkan solusi yang lebih baik jika dibandingkan dengan metode yang telah ada. Oleh karena itu, rumusan masalah dalam penelitian ini adalah sebagai berikut.

1. Bagaimana penerapan Algoritma *Elephant Herding Optimization* dalam menyelesaikan kasus *Asymmetric Travelling Salesman Problem*?
2. Bagaimana pengaruh dari parameter-parameter *Elephant Herding Optimization* terhadap performansinya?
3. Bagaimana perbandingan hasil solusi pada persoalan kasus *benchmark* yang pernah diselesaikan menggunakan *A New Genetic Algorithm* (Nagata & Soler, 2012), *Improved Discrete Bat Algorithm* (Osaba et. al., 2016), dan *Harmony Search Algorithm* (Kevin, 2016)?

### **I.3 Batasan Masalah**

Pembatasan masalah pada penelitian ini dibuat sesuai dengan identifikasi masalah. Pembatasan masalah yang digunakan untuk menyederhanakan masalah dalam penelitian ini adalah jenis permasalahan yang digunakan pada penelitian ini, hanya terbatas pada *benchmark* yang umum digunakan.

### **I.4 Tujuan Penelitian**

Berdasarkan rumusan masalah diatas, berikut dibawah ini adalah tujuan dari penelitian yang dilakukan.

1. Menyelesaikan permasalahan *Asymmetric Travelling Salesman Problem* (ATSP) dengan menggunakan algoritma *Elephant Herding Optimization* (EHO).
2. Mengetahui pengaruh parameter-parameter dari *Elephant Herding Optimization* terhadap performansinya.

3. Membandingkan hasil solusi pada permasalahan *Asymmetric Travelling Salesman Problem* yang menggunakan Algoritma *Elephant Herding Optimization* (EHO) dengan *A New Genetic Algorithm* (Nagata & Soler, 2012), *Improved Discrete Bat Algorithm* (Osaba et. al., 2016), dan *Harmony Search Algorithm* (Kevin, 2016).

### **I.7 Manfaat Penelitian**

Berdasarkan penelitian yang akan dilakukan, manfaat yang diharapkan dalam penelitian ini adalah sebagai berikut.

1. Menambah pengetahuan dan wawasan dalam menyelesaikan permasalahan *Asymmetric Travelling Salesman Problem*.
2. Menambah referensi untuk penelitian mengenai algoritma *Elephant Herding Optimization* dan *Asymmetric Travelling Salesman Problem*.

### **I.8 Metodologi Penelitian**

Dalam melakukan penelitian mengenai perancangan serta penerapan algoritma *Elephant Herding Optimization* kepada permasalahan *Asymmetric Travelling Salesman Problem*, diperlukan suatu metodologi penelitian. Berikut merupakan langkah-langkah yang dilakukan pada penelitian ini.

1. Studi Literatur  
Langkah pertama yang dilakukan pada penelitian ini adalah mencari serta mengumpulkan informasi mengenai *Elephant Herding Optimization* (EHO) serta *Asymmetric Travelling Salesman Problem* (ATSP) dari referensi jurnal serta buku yang terkait.
2. Identifikasi dan Perumusan Masalah  
Setelah studi literatur, dilanjutkan dengan identifikasi dan perumusan masalah. Identifikasi yang dimaksudkan adalah masalah ATSP beserta penerapan EHO. Perumusan masalah dilakukan berdasarkan identifikasi masalah yang telah dilakukan.
3. Penentuan Batasan Masalah dan Asumsi  
Setelah dilakukan identifikasi dan perumusan masalah, ditentukanlah batasan-batasan masalah serta asumsi yang digunakan dalam penelitian yang bertujuan untuk memfokuskan penelitian dan membantu proses penelitian yang ada.

4. Penentuan Tujuan dan Manfaat Penelitian  
Penelitian dalam bentuk apapun harus memiliki sebuah tujuan dan manfaat. Penentuan tujuan dan manfaat penelitian dibuat agar penelitian ini memiliki maksud dan manfaat yang jelas.
5. Perancangan *Elephant Herding Optimization* Pada Permasalahan *Asymmetric Travelling Salesman Problem*.  
Perancangan algoritma adalah tahap awal yang diperlukan dalam penyelesaian masalah ATSP agar hasil dari penerapan algoritma EHO menggambarkan solusi ATSP. Pembuatan program ini menggunakan *software* Netbeans IDE 8.1.
6. Verifikasi Algoritma  
Perancangan algoritma pada tahap sebelumnya harus dilakukan verifikasi untuk memastikan bahwa setiap komponen perancangan program sesuai dengan permasalahan yang ada.
7. Validasi Algoritma  
Perancangan algoritma yang telah melewati tahap verifikasi dilakukan proses validasi dengan melihat *output* yang dihasilkan oleh program.
8. Penyelesaian *Asymmetric Travelling Salesman Problem* (ATSP) Dengan Algoritma *Elephant Herding Optimization* (EHO)  
Pada tahap ini, program algoritma EHO akan dilakukan input beberapa kasus ATSP untuk mengetahui solusi permasalahan yang dihasilkan oleh algoritma EHO tersebut.
9. Evaluasi Pengaruh Parameter Pada Algoritma *Elephant Herding Optimization* (EHO)  
Pada tahap ini, parameter-parameter yang menjadi komponen EHO akan dievaluasi, lalu parameter tersebut akan dilakukan perhitungan pengaruh dari setiap parameter beserta interaksinya saat menyelesaikan kasus-kasus ATSP pada tahap sebelumnya.
10. Melakukan Perbandingan Hasil yang Didapat Menggunakan EHO Dengan Metode Penyelesaian Masalah Lainnya.  
Pada tahap ini dilakukan perbandingan mengenai *output* yang dihasilkan oleh algoritma EHO pada tahap sebelumnya dengan *output* yang dihasilkan oleh algoritma pembanding serta menghitung kedekatan dengan *Best Known Solution* pada kasus terkait.

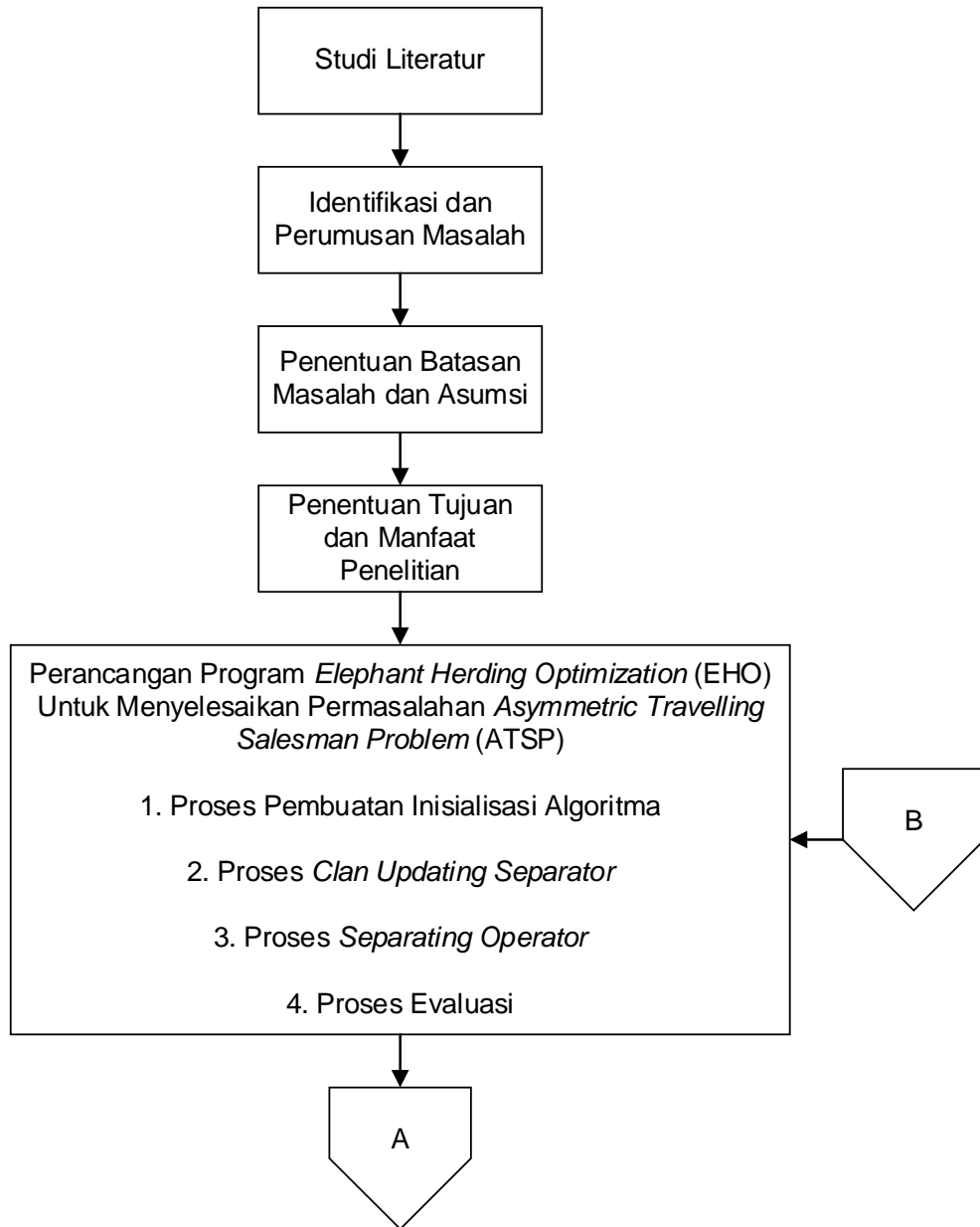
11. Analisis

Analisis dilakukan terhadap solusi yang dihasilkan algoritma EHO terhadap permasalahan ATSP beserta hasil *benchmarking* dari metode penyelesaian masalah lainnya. Analisis juga dilakukan untuk mengetahui parameter apa yang memiliki pengaruh yang paling signifikan dalam menyelesaikan permasalahan ATSP.

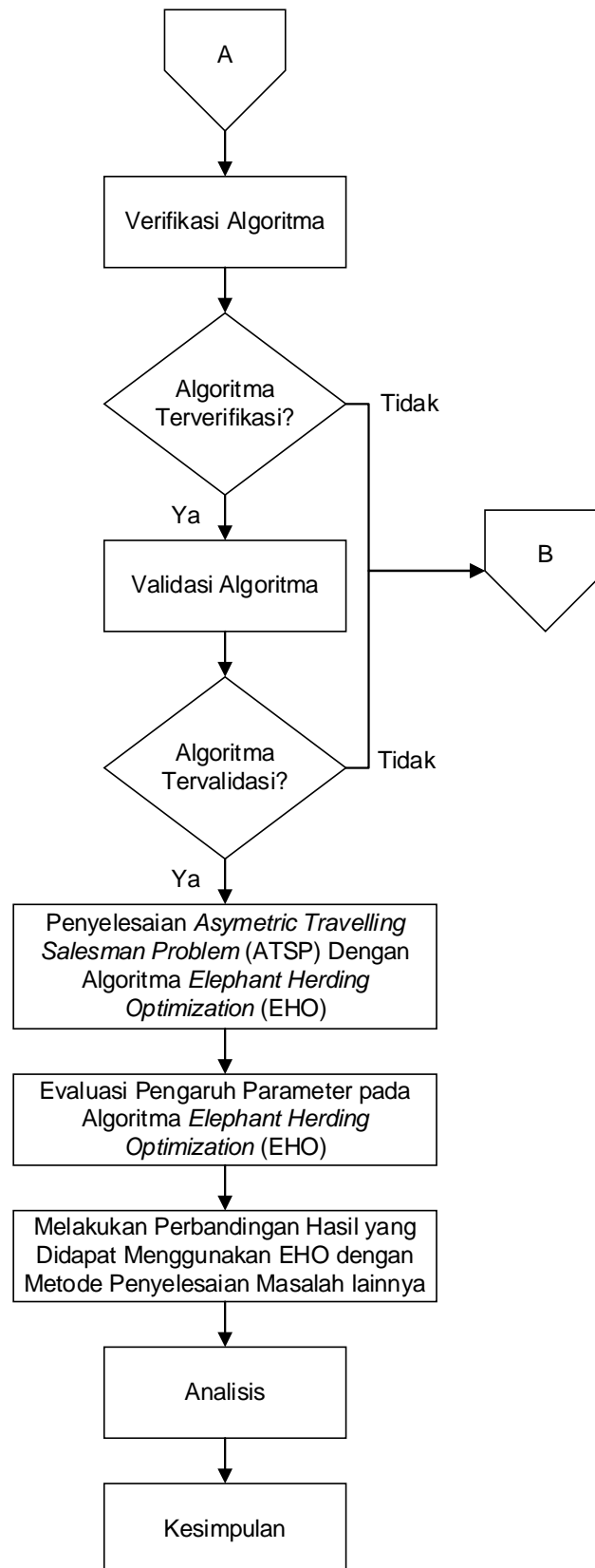
12. Kesimpulan

Penarikan kesimpulan dilakukan untuk menjawab rumusan masalah yang telah ditentukan sebelumnya.

Gambar I.1 merupakan *flowchart* dari metodologi penelitian yang dirancang.



Gambar I.1 *Flowchart* Metodologi Penelitian

Gambar I.1 *Flowchart* Metodologi Penelitian (lanjutan)