

**PENERAPAN *DRAGONFLY ALGORITHM* UNTUK
MENYELESAIKAN *VEHICLE ROUTING PROBLEM*
WITH *MULTI-TRIPS***

SKRIPSI

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

Disusun oleh:

Nama : Teresa Davina

NPM : 2016610023



**PROGRAM STUDI SARJANA TEKNIK INDUSTRI
JURUSAN TEKNIK INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS KATOLIK PARAHYANGAN
2021**

**FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS KATOLIK PARAHYANGAN
BANDUNG**



Nama : Teresa Davina
NPM : 2016610023
Jurusan : Teknik Industri
Judul Skripsi : PENERAPAN *DRAGONFLY ALGORITHM* UNTUK
MENYELESAIKAN *VEHICLE ROUTING PROBLEM WITH MULTI-TRIPS*

TANDA PERSETUJUAN SKRIPSI

Bandung, Agustus 2021

Ketua Program Studi Sarjana Teknik Industri

(Dr. Ceicalia Tesavrita, S.T., M.T.)

Pembimbing Pertama

23 Agustus 2021

(Fran Setiawan, S.T., M.Sc.)

Pembimbing Kedua

23 Agustus 2021

(Dr. Sugih Sudharma Tjandra, S.T., M.Si.)



Jurusan Teknik Industri
Fakultas Teknologi Industri
Universitas Katolik Parahyangan

Pernyataan Tidak Mencontek atau Melakukan Tindakan Plagiat

Saya, yang bertanda tangan di bawah ini,

Nama : Teresa Davina

NPM : 2016610023

dengan ini menyatakan bahwa skripsi dengan judul :

“PENERAPAN *DRAGONFLY ALGORITHM* UNTUK MENYELESAIKAN *VEHICLE ROUTING PROBLEM WITH MULTI-TRIPS*”

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.

Jakarta,

Teresa Davina
2016610023

ABSTRAK

Biaya *supply chain* rata-rata mencakup 55% harga produk, dimana transportasi memiliki peran terbesar dalam rantai logistik. Seiring perkembangan ekonomi dan globalisasi, iklim industri semakin kompetitif. Untuk bisa tetap bersaing dan berkembang, perusahaan-perusahaan terus mencari cara untuk meningkatkan efisiensi *supply chain*. Terutama di kota besar dan padat seperti Jakarta, dimana kendaraan-kendaraan besar seperti truk dan lori sulit menavigasi jalanan yang sempit. Oleh karena itu, lebih baik menggunakan kendaraan kecil yang bisa melakukan banyak trip dalam satu hari kerja. Permasalahan ini merupakan *Vehicle Routing Problem with Multi-Trip*. Meski sudah cukup banyak algoritma-algoritma metaheuristik yang dibuat untuk menyelesaikan permasalahan ini, belum ada yang berhasil menemukan solusi optimal dengan konsisten.

Pada penelitian ini, dilakukan pengembangan *Dragonfly Algorithm* untuk menyelesaikan kasus VRPMT, dengan menggunakan pendekatan yang terinspirasi dari proses *initialization* penelitian *Variable Neighborhood Search* untuk kasus VRPMT. Algoritma yang dibuat dijalankan dengan program Matlab.

Hasil dari *dragonfly algorithm* dengan proses *encoding decoding* yang dibuat tidak berhasil untuk menyelesaikan kasus VRPMT, khususnya pada populasi pelanggan yang besar. Karena tidak memaksimalkan kapasitas kendaraan dalam rute dan kurang optimalnya rute yang dibuat, batasan waktu terkadang tidak terpenuhi. Diperlukan penelitian lebih lanjut untuk mengembangkan algoritma, baik dengan menambahkan *local search*, atau dengan mengubah proses *encoding decoding* yang digunakan dengan proses yang lebih sesuai.

Kata Kunci: VRPMT, *Dragonfly Algorithm*, *Supply Chain*, Transportasi, *Variable Neighborhood Search*

ABSTRACT

The average supply chain cost covers 55% of the product price, of which transportation has the largest impact in the logistics chain. With the growing economic world and globalization, the industrial climate is increasingly competitive. In order to stay competitive and be able to thrive, companies are constantly looking for ways to improve supply chain efficiency. This is especially true in a big and congested city like Jakarta, where large vehicles such as trucks and lorries find it difficult to navigate the narrow streets. Therefore, it is better to use a small vehicle that can do many trips in one working day. This problem is called the Vehicle routing Problem with Multi-Trip. Although quite a number of metaheuristic algorithms have been developed to solve this problem, no one has been able to find the optimal solution consistently

In this study, the author developed a Dragonfly Algorithm to solve the VRPMT case, using an approach inspired by the Variable Neighborhood Search research initialization process for the VRPMT. The algorithm created is run with the Matlab software.

The results from the dragonfly algorithm with the encoding decoding process that was made did not succeed in solving the VRPMT case, especially in a large customer population. Due to not maximizing the vehicle capacity on the route and the lack of feasible routes created, time constraints are sometimes not met. Further research is needed to develop the algorithm, either by adding local search, or by changing the encoding decoding process used with a more suitable process.

Keywords: VRPMT, Dragonfly Algorithm, Supply Chain, Transportation, Variable Neighborhood Search

KATA PENGANTAR

Puji syukur kepada Tuhan yang Maha Esa atas berkat dan rahmat-Nya sehingga penelitian untuk skripsi yang berjudul “Penerapan *Dragonfly Algorithm* untuk *Vehicle Routing Problem with Multi-Trips*” dapat terselesaikan. Penelitian ini diajukan demi pemenuhan syarat akademik, untuk menyelesaikan program studi sarjana 1 teknik industri di Universitas katolik Parahyangan. Penulis menyadari bahwa penelitian ini jauh dari sempurna, karena keterbatasan kemampuan penulis sendiri.

Atas kekurangan-kekurangan dalam penelitian ini, penulis mengharapkan akan adanya masukan-masukan, atau akan adanya penelitian lain yang bisa melampaui kekurangan-kekurangannya. Penulis menghadapi banyak kesulitan dalam proses penulisan, yang hanya bisa penulis lewati dengan banyak bantuan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan rasa terimakasih yang sebesar-besarnya, khususnya kepada:

1. Pak Fran Setiawan, S.T., M.Sc. dan Pak Dr. Sugih Sudharma Tjandra, S.T., M.Si, yang telah membimbing, menyemangati, memberi ilmu, dan motivasi penulis dalam setiap tahap pembuatan skripsi, jauh melebihi kewajiban sebagai pembimbing I dan II.
2. Orang tua penulis, Ir. Irfan Natakesuma Chaidir dan Henny Chaidir, dan kakak penulis, Maria Tiffany, yang dengan rasa sabar, kasih sayang dan penuh pengertian, membantu penulis melewati masa-masa tersulit penulis selama proses penelitian dan penulisan.
3. Pak Alfian, S.T., M.T. sebagai dosen wali penulis, yang juga membantu penulis melewati masa-masa sulit dalam proses penelitian dan penulisan, dan Kak Adi Putra, yang membantu mengajari penulis membuat program.
4. Teman-teman seperjuangan angkatan 2016, terutama Vieri Gunawan yang dengan sabar menjawab semua pertanyaan-pertanyaan penelitian, teman-teman kelompok PST, serta Nisa Hafizha Amalia yang ikut menyemangati
5. Seluruh dosen-dosen dan staff yang membimbing penulis dalam masa belajar di Universitas Katolik Parahyangan, yang namanya tidak bisa

disebutkan satu-per-satu, yang membimbing dan membantu penulis selama ini.

Terima kasih atas semua bantuan dan dukungannya, masa-masa ini tidak akan pernah penulis lupakan.

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR TABEL	vii
DAFTAR GAMBAR	ix
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	I-1
I.1 Latar Belakang Masalah	I-1
I.2 Identifikasi dan Rumusan Masalah	I-4
I.3 Pembatasan Masalah dan Asumsi Penelitian	I-7
I.4 Tujuan Penelitian	I-8
I.5 Manfaat Penelitian	I-8
I.6 Metodologi Penelitian	I-8
I.7 Sistematika Penulisan	I-11
BAB II TINJAUAN PUSTAKA	II-1
II.1 <i>City Logistics</i>	II-1
II.2 <i>Vehicle Routing Problem with Multi-Trips</i>	II-2
II.3 <i>Dragonfly Algorithm</i>	II-4
BAB III PERANCANGAN ALGORITMA	III-1
III.1 <i>Encoding dan Decoding</i>	III-1
III.2 Pembaharuan Posisi Capung Menggunakan Algoritma <i>Dragonfly</i> ...	III-4
III.3 Penerapan <i>Dragonfly Algorithm</i> untuk Menyelesaikan VRPMT	III-8
III.3.1 Notasi Algoritma.....	III-8
III.3.2 Algoritma Utama.....	III-10
III.3.3 Algoritma Pembuatan Matriks Posisi Awal Capung	III-11
III.3.4 Algoritma <i>Decoding</i>	III-15
III.3.5 Algoritma Perhitungan <i>Fitness</i>	III-18
III.3.6 Algoritma Pencari <i>Food</i> dan <i>Enemy</i>	III-18
III.3.7 Algoritma Penghitung Matriks <i>Step Vector</i> dan Posisi	III-20

III.4 Verifikasi Algoritma	III-1
III.5 Validasi Algoritma	III-1
BAB IV IMPLEMENTASI ALGORITMA	IV-1
IV.1 Verifikasi dan Validasi Program	IV-1
IV.2 Pengerjaan Kasus Hipotetik.....	IV-3
IV.3 Pengerjaan Kasus CMT dengan Program.....	IV-7
BAB V ANALISIS	V-1
V.1 Analisis Proses <i>Encoding</i> dan <i>Decoding</i>	V-1
V.2 Analisis <i>Dragonfly Algorithm</i>	V-4
V.3 Analisis Pemilihan Kasus.....	V-5
BAB VI KESIMPULAN DAN SARAN	VI-1
DAFTAR PUSTAKA	
LAMPIRAN	
DAFTAR RIWAYAT HIDUP	

DAFTAR TABEL

Tabel III.1	Koordinat dan <i>Demand</i>	III-1
Tabel III.2	Jarak <i>Euclidean</i>	III-2
Tabel III.3	Jarak <i>Euclidean</i> Kendaraan dan Konsumen	III-2
Tabel III.4	Solusi Awal	III-4
Tabel III.5	Jarak <i>Euclidean</i> Antar Capung	III-5
Tabel III.6	Perhitungan Faktor Pengali Iterasi.....	III-7
Tabel III.7	Rute Terbaik Kasus Sederhana.....	III-8
Tabel III.8	Jarak <i>Euclidean</i> Capung dan Konsumen	III-24
Tabel III.9	Rute Iterasi	III-25
Tabel III.10	Jarak <i>Euclidean</i> antar Capung Iterasi 1	III-27
Tabel IV.1	Rute Program Kasus Hipotetik Enam Pelanggan.....	IV-3
Tabel IV.2	Data Kasus Hipotetik Delapan Pelanggan	IV-4
Tabel IV.3	Rute Program Kasus Hipotetik Delapan Pelanggan.....	IV-4
Tabel IV.4	Kombinasi Kasus Hipotetik	IV-5
Tabel IV.5	Kasus CMT1.....	IV-7
Tabel IV.6	Kasus CMT2.....	IV-8
Tabel IV.7	Kasus CMT3.....	IV-8
Tabel IV.8	Kasus CMT4.....	IV-9

DAFTAR GAMBAR

Gambar I.1	<i>Flowchart</i> Metodologi Penelitian.....	I-10
Gambar II.1	Pengerumunan Dinamis (Kiri) dan Statis (kanan).....	II-5
Gambar II.2	Pergerakan Capung.....	II-5
Gambar III.1	<i>Flowchart</i> Algoritma Utama.....	III-12
Gambar III.2	<i>Flowchart</i> Algoritma Pembuatan Matriks Posisi Awal Capung.....	III-13
Gambar III.3	<i>Flowchart</i> Algoritma <i>Decoding</i>	III-15
Gambar III.4	<i>Flowchart</i> Algoritma Perhitungan <i>Fitness</i>	III-19
Gambar III.5	<i>Flowchart</i> Algoritma Pencarian <i>Food</i> dan <i>Enemy</i>	III-20
Gambar III.6	<i>Flowchart</i> Algoritma Matriks <i>Step Vector</i> dan Posisi.....	III-22
Gambar IV.1	Program Pembuatan Matriks Posisi Awal.....	IV-1
Gambar IV.2	Program Batasan C dan TH.....	IV-2
Gambar IV.3	Input Program.....	IV-2
Gambar IV.4	Hasil Program.....	IV-2
Gambar IV.5	Kasus Hipotetik Enam Pelanggan.....	IV-3
Gambar IV.6	Kasus Hipotetik Delapan Pelanggan.....	IV-4
Gambar IV.7	Kombinasi Satu Kasus Enam Pelanggan.....	IV-5
Gambar IV.8	Detail Rute Kombinasi Satu.....	IV-5
Gambar IV.9	Tampilan <i>Error</i> Kombinasi Tiga.....	IV-6

DAFTAR LAMPIRAN

DATA KASUS CMT	A-1
----------------------	-----

BAB I

PENDAHULUAN

Pada bab ini akan disajikan latar belakang masalah, identifikasi dan rumusan masalah, pembatasan masalah dan asumsi penelitian, tujuan penelitian, manfaat penelitian, metodologi penelitian, serta sistematika penulisan yang akan dilakukan dalam penelitian ini.

I.1 Latar Belakang Masalah

Seiring perkembangan ekonomi dan globalisasi, dunia perindustrian harus terus berkembang agar bisa bertahan, bersaing, dan berkembang dalam iklim pasar yang semakin kompetitif. Untuk itu, setiap industri harus memiliki manajemen yang baik dalam setiap tahap produksi yang ada dalam *supply chain*. Biaya *supply chain* mencakup sekitar 55% dari harga total produk. Karena *supply chain* memiliki dampak besar terhadap harga produk akhir, proses perbaikan terus menerus diperlukan dalam setiap tahap dalam *supply chain*, sehingga mengurangi biaya keseluruhan (Lapinskaitė & Kuckailytė, 2014). Menurut Tseng, Yue, dan Taylor (2005), transportasi memainkan peran terbesar dalam rantai logistik dan dianggap sebagai komponen paling penting karena memerankan bagian terbesar dari *life cycle* produk. Masalah ini bisa diatasi dengan menentukan rute perjalanan angkutan pengiriman yang lebih baik.

Vehicle Routing Problem (VRP) merupakan pendesainan rute pengantaran berbiaya terendah melalui sekumpulan pelanggan yang tersebar secara geografis, dengan mempertimbangkan berbagai batasan. Permasalahan ini memiliki posisi utama dalam manajemen distribusi dan dihadapi sehari-hari oleh puluhan ribu *carrier* sedunia. Masalah muncul dalam beberapa bentuk karena berbagai macam batasan yang ditemukan saat penerapan (Laporte, Toth, & Vigo, 2013). Laporte et al. (2013) menjelaskan bahkan dalam kasus-kasus sederhana dengan batasan kapasitas (*capacitated VRP*), tetap sulit untuk menggunakan algoritma eksak untuk seratus atau dua ratus pelanggan. Dalam beberapa tahun terakhir, banyak usaha penelitian telah beralih ke pengembangan metaheuristik yang kuat. Perkembangan heuristik modern untuk VRP benar-benar dimulai sekitar tahun

1990 dengan munculnya metaheuristik. Bisa dibilang pembelajaran VRP telah menstimulasi pertumbuhan dan pengertian akan beberapa konsep metaheuristik yang kita kenal sekarang (Laporte et al., 2013).

Dalam permasalahan *Vehicle Routing Problem* biasa, setiap alat angkut hanya memiliki satu rute yang dijalani setiap harinya. Oleh karena itu, metode ini sulit diterapkan langsung ke kehidupan nyata karena tidak memperhitungkan konstrain-konstrain seperti modal untuk kendaraan, jumlah kendaraan yang dimiliki, kapasitas kendaraan yang digunakan, dan sebagainya. Batasan ini membuat pengiriman menggunakan alat angkut dengan kapasitas kecil kurang efisien, dengan utilisasi setiap kendaraannya rendah. Hal ini dikarenakan setiap barang yang diangkut di awal rute sudah habis, kendaraan tersebut dianggap tidak akan mengambil rute lain, dan tidak akan digunakan lagi untuk sisa hari tersebut. Untuk dapat mempertimbangkan batasan-batasan di kehidupan nyata tersebut, VRP dikembangkan menjadi beberapa variasi yang berfokus pada batasan-batasan yang ingin dilalui.

Logistik perkotaan bertujuan untuk mengurangi gangguan macet di jalan-jalan perkotaan yang diasosiasikan dengan truk dan mobil van sembari mendukung perkembangan ekonomi dan sosial kota-kota tersebut (Crainic, 2008). Untuk mendukung hal ini, pengiriman dalam kota sebaiknya menggunakan kendaraan kecil, baik mobil atau motor. Bila menentukan rute menggunakan VRP biasa, akan membutuhkan banyak kendaraan pengiriman bila memperhatikan logistic perkotaan, yang menyatakan bahwa kendaraan yang digunakan memiliki kapasitas kecil. Setiap kendaraan tersebut juga akan memiliki tingkat utilisasi yang rendah, karena setelah menyelesaikan rute akan dianggap tidak bisa digunakan lagi, meski rute yang dijalankan relatif lebih pendek karena kapasitas kendaraan kecil.

Vehicle Routing Problem with Multi-Trips merupakan salah satu varian dari VRP standar dimana sebuah kendaraan bisa ditugaskan ke beberapa rute dalam periode perencanaan (Taillard, Laporte, & Gendreau, 1995). Salah satu pertimbangan dari VRPMT adalah ukuran kapasitas kendaraan yang kecil. Oleh karena itu, bila sebuah kendaraan kecil sudah menyelesaikan satu rute, ia bisa kembali ke posisi awal untuk mengambil rute lainnya. Hal ini akan meningkatkan utilisasi kendaraan pengiriman, mengurangi modal yang dibutuhkan untuk

membeli lebih banyak kendaraan, serta meningkatkan efisiensi rantai pasok didalam kota.

Contoh nyata VRPMT adalah perusahaan yang mengantarkan produk-produk yang diproduksi ke berbagai *reseller* di dalam suatu kota. Untuk melakukan hal ini, perusahaan-perusahaan memiliki kendaraan transportasi barang sendiri dengan jumlah terbatas. Dalam satu hari perusahaan mungkin akan perlu mengirimkan barang ke sejumlah reseller, menggunakan rute-rute yang sudah ditentukan untuk memaksimalkan kapasitas kendaraan, dan meminimasi bahan bakar yang digunakan dengan setiap kendaraan yang ada. Oleh karena itu, kendaraan-kendaraan mungkin akan perlu menjalani lebih dari satu rute dalam satu hari kerja.

Carić & Gold (2008) menyatakan bahwa VRP merupakan permasalahan NP *hard combinatorial optimization* yang bisa diselesaikan secara *exact* hanya untuk kejadian kecil permasalahan, dan meski pendekatan heuristik tidak menjamin optimalitas, tetapi metode tersebut akan menghasilkan hasil terbaik. Caric & Gold (2008) juga mengatakan bahwa dalam 20 tahun terakhir (dari penerbitan tahun 2008) metode metaheuristik merupakan arah penelitian yang paling menjanjikan untuk berbagai variasi VRP.

Talbi (2009) menyatakan bahwa metode heuristik menemukan solusi-solusi “baik” untuk suatu permasalahan berukuran besar. Metode tersebut memberikan solusi dengan performansi dan yang bisa diterima. Mereka didesain untuk menyelesaikan suatu permasalahan atau instansi yang spesifik. Metaheuristik merupakan algoritma yang memiliki fungsi lebih general yang bisa digunakan untuk menyelesaikan hampir semua masalah optimasi. Algoritma-algoritma metaheuristik merupakan paradigma *computational intelligence* yang digunakan khususnya untuk memecahkan masalah-masalah optimasi (Abdel-Basset, Abdel-Fatah, & Sangaiah, 2018).

Yang (2010) menyatakan bahwa “dua komponen utama dari algoritma metaheuristik adalah intensifikasi dan diversifikasi, atau eksploitasi dan eksplorasi. Diversifikasi bertujuan untuk menggenerasi solusi-solusi beragam untuk eksplorasi ruang pencarian dalam skala global, sementara intensifikasi adalah untuk mengfokuskan pencarian di daerah lokal dengan mengeksploitasi informasi bahwa ada solusi “baik” yang ditemukan di daerah tersebut. Hal ini dikombinasikan dengan seleksi solusi terbaik. Pemilihan yang terbaik memastikan bahwa solusi-

solusi akan bertemu di optimalitas, sementara diversifikasi dari randomisasi menghindarkan solusi-solusi dari terjebak di optima lokal, serta di saat yang sama meningkatkan keragaman solusi-solusinya. Kombinasi baik dari dua komponen utama biasanya akan memastikan bahwa optimalitas global bisa dicapai.” Oleh karena itu, penelitian yang dilakukan akan menggunakan metode metaheuristik.

I.2 Identifikasi dan Rumusan Masalah

Algoritma-algoritma metaheuristik dibagi berdasarkan *metaphor based* dan *non-metaphor based* untuk mendiferensiasi dalam skema pencarian dan menjelaskan bagaimana algoritma yang berdasarkan metafora mensimulasikan perilaku fenomena yang dipilih di area pencarian (Abdel-Basset et al., 2018). Contoh algoritma berbasis non-metafora adalah *Tabu Search* (TS), dan *Variable Neighborhood Search* (VNS). Contoh algoritma-algoritma dalam kategori metafora adalah *Genetic Algorithm* (GA), *Particle Swarm Optimization* (PSO), *Water Waves Optimization* (WWO), *Clonal Selection Algorithm* (CLONALG), *Chemical Reaction Optimization* (CRO), *Harmony Search* (HS), *Sine Cosine Algorithm* (SCA), *Simulated Annealing* (SA), *Teaching–Learning-Based Optimization* (TLBO), *League Championship Algorithm* (LCA), dan sebagainya.

Permasalahan VRPMT sudah pernah diselesaikan menggunakan berbagai metode. VRPMT dapat diselesaikan menggunakan algoritma *exact* dengan *two set-partitioning-like formulations* (Mingozzi, Roberti, & Toth, 2012), dalam 42 dari 52 kasus dengan batasan maksimal 120 pelanggan. Dalam kategori algoritma heuristik, VRPMT pernah diselesaikan dengan metode *multi-phase constructive heuristic* (Petch & Salhi, 2004). Hasil dari penelitian tersebut cukup baik, dengan memberikan solusi populasi, tetapi Petch dan Salhi (2004) melanjutkan bahwa untuk memilih solusi, sebaiknya mengintegrasikan *genetic algorithm* yang merupakan bagian dari kelompok metaheuristik. Metode-metode metaheuristik yang sudah pernah digunakan untuk VRPMT (atau nama-nama lain yang digunakan) adalah *variable neighborhood search algorithm* (Cheikh, Ratli, Mkaouar, & Jarboui, 2015), *hybrid genetic algorithm* (Salhi & Petch, 2007), *memetic algorithm* (Cattaruzza, Absi, Feillet, & Vidal, 2014), *adaptive memory algorithm* (Olivera & Viera, 2007), *iterative local search with variable neighborhood search* (Belver, Gomez, Lopez, De la Fuente, & Ponte, 2017) dan *adaptive large neighborhood search* (Azi, Gendreau, Potvin, & Centre-ville, 2014). Dari penelitian-

penelitian yang dilakukan, belum ada algoritma yang berhasil menemukan *global optima* untuk kasus VRPMT, meski beberapa berhasil mendekati global optima di sebagian kejadian. Oleh karena itu, masih ada kesempatan untuk penelitian penyelesaian kasus VRPMT dengan algoritma lain, untuk mencari algoritma yang paling sesuai menyelesaikan kasus VRPMT.

Metaheuristik merupakan metode optimasi yang memberikan solusi yang memenuhi parameter-parameter masalah, dan bisa diterapkan pada berbagai masalah yang berbeda, tidak seperti heuristik, dimana algoritma untuk satu masalah tidak dapat diterapkan pada masalah lain.

Metaheuristik dapat dibagi menjadi dua klasifikasi, *single-solution based* dan *population-based*. Metode berbasis *single-solution*, juga disebut metode *trajectory*, memanipulasi solusi tunggal. Metode berbasis *population* mengulang dan memanipulasi banyak solusi sekaligus. Teknik berbasis solusi tunggal (misalnya *tabu search*, *simulated annealing* dan *local search*) berorientasi pada intensifikasi, sedangkan teknik berbasis populasi (misalnya *genetic algorithm* dan *particle swarm algorithm*) lebih difokuskan pada eksplorasi ruang pencarian (Gogna, & Tayal, 2013). Gogna dan Tayal (2013) lanjut menjelaskan bahwa metaheuristik berbasis populasi menghasilkan kelompok solusi baru dalam setiap iterasi, sementara *single-solution based* melakukan generasi dan penggantian hanya akan satu solusi. Setiap solusi yang digenerasi akan dimanipulasi untuk menghasilkan kumpulan solusi baru secara terus menerus hingga kriteria-kriteria yang ditentukan terpenuhi. Metaheuristik berbasis populasi memberikan banyak solusi yang bisa memenuhi kriteria. Sebagian besar metaheuristik berbasis populasi terinspirasi dari alam.

Salah satu cabang dari metaheuristik *population based* yang terinspirasi dari alam merupakan algoritma-algoritma *swarm intelligence* (SI). SI didasarkan pada perilaku kolektif dari sistem-sistem yang mandiri. Contoh skema SI adalah *particle swarm optimization*, *ant colony system*, *stochastic diffusion search*, *bacteria foraging*, *the artificial bee colony*, dan sebagainya (Chu, Huang, Roddick, & Pan, 2011). Sebagian besar terinspirasi dari sistem-sistem biologis, SI mengadopsi perilaku kolektif kelompok hewan, selagi mereka berusaha bertahan hidup (Chakraborty & Kar, 2017). *Dragonfly algorithm* merupakan algoritma metaheuristik baru yang termasuk dalam kategori SI. Inspirasi utama algoritma DA berasal dari perilaku capung yang statis dan dinamis di alam. Dua fase penting

dari optimasi, eksplorasi dan eksploitasi, dirancang dengan memodelkan interaksi sosial capung dalam navigasi, mencari makanan, dan menghindari musuh saat mengerumuni secara dinamis atau statistik (Mirjalili, 2015).

Dalam penelitian Mirjalili, ia membandingkan algoritma *dragonfly* dengan PSO yang dianggap sebagai algoritma terbaik dalam algoritma kategori *swarm-based*, dan GA sebagai algoritma terbaik dalam kategori *evolutionary*. Ia melakukan perbandingan dalam 7 kasus *unimodal* (memiliki satu optimum, baik untuk menguji eksploitasi dan *convergence*), 6 kasus *multi-modal* (memiliki banyak optimum, dimana hanya satu merupakan *global optima*. Baik untuk menguji eksplorasi, dan penghindaran *local optima*), dan 6 kasus *composite* (menggabungkan *test function* dari *unimodal* dan *multimodal*. Baik untuk menguji keseimbangan dari eksplorasi dan eksploitasi). Hasil penelitian menemukan bahwa algoritma DA memiliki performa lebih baik dibandingkan GA dan PSO di sebagian besar kasus *unimodal* yang diuji, sehingga bisa disimpulkan bahwa DA memiliki kapabilitas eksploitasi yang tinggi. Dari pengujian *multi-modal*, ditemukan bahwa DA memberikan hasil yang kompetitif dengan PSO, dan lebih baik yang signifikan dibandingkan GA, dan bisa disimpulkan bahwa DA memiliki kemampuan eksplorasi yang tinggi, serta memiliki kapabilitas penghindaran *local optima* yang baik. Dalam pengujian *composite*, hasil dari DA kompetitif dengan PSO dan GA, dan lebih baik di beberapa kasus, dan disimpulkan bahwa DA bisa menyeimbangkan kemampuan eksplorasi dan eksploitasi, sehingga DA kemungkinan bisa menyelesaikan permasalahan-permasalahan optimasi yang sulit di keadaan nyata.

Gunawan (2020) menggunakan algoritma *dragonfly* untuk menyelesaikan permasalahan *capacitated vehicle routing problem with time windows*, dimana dilakukan *benchmark* dengan algoritma *elephant herding optimization*. Dari 6 kasus yang dibandingkan, DA berhasil menemukan *best known solution*, sama seperti EHO, dalam 1 kasus. DA juga berhasil menemukan solusi yang lebih baik dalam 4 kasus, dan 1 solusi yang lebih buruk dalam 1 kasus. Dengan menggunakan *dragonfly algorithm*, diharapkan dapat memberikan performansi VRPMT yang lebih baik dari algoritma-algoritma lain yang sudah diuji sebelumnya. Oleh karena itu, penelitian ini akan melakukan penerapan *dragonfly algorithm* untuk menyelesaikan *vehicle routing problem with multi-trips*.

Dari identifikasi masalah yang sudah dilakukan, berikut perumusan masalah dalam penelitian ini:

1. Bagaimana penerapan algoritma *Dragonfly* untuk menyelesaikan permasalahan *Vehicle Routing Problem with Multi Trips*?
2. Bagaimana performansi algoritma *Dragonfly* dibandingkan dengan *genetic algorithm* (Salhi, & Petch, 2007), *memetic algorithm*, dan *memetic algorithm with complete local search* (Cattaruzza et al., 2014).

I.3 Pembatasan Masalah dan Asumsi Penelitian

Setelah identifikasi dan perumusan masalah dilakukan, selanjutnya akan ditentukan penentuan batasan masalah dan asumsi pada penelitian yang akan dilakukan. Pembatasan masalah dan asumsi dalam penelitian ini adalah :

1. Nilai *fitness* hasil terbaik Algoritma *dragonfly* akan dibandingkan dengan nilai *fitness benchmark*.
2. Hanya menggunakan perbandingan performansi dari *genetic algorithm* (Salhi, & Petch, 2007), *memetic algorithm*, dan *memetic algorithm with complete local search* (Cattaruzza et al., 2014).
3. Hanya menggunakan *dataset benchmark* CMT1, CMT2, CMT3, dan CMT4.

I.4 Tujuan Penelitian

Dari rumusan masalah yang telah dibuat, akan ditentukan tujuan dari penelitian yang akan dilakukan. Berikut tujuan-tujuan tersebut:

1. Menerapkan *Dragonfly Algorithm* dalam menyelesaikan permasalahan *Vehicle Routing Problem with Multi-Trips*.
2. Membandingkan performansi algoritma *Dragonfly* dengan *genetic algorithm* (Salhi, & Petch, 2007) dan *memetic algorithm* (Cattaruzza et al., 2014).

I.5 Manfaat Penelitian

Dari penelitian penerapan *Dragonfly Algorithm* dalam menyelesaikan permasalahan *Vehicle Routing Problem with Multi-Trips*, diharapkan penelitian akan bisa memberikan beberapa manfaat sebagai berikut:

1. Mengembangkan penelitian akan *Dragonfly Algorithm* dan *Vehicle Routing Problem with Multi-Trips*.

2. Memberikan alternatif penyelesaian untuk *Vehicle Routing Problem with Multi-Trips*.
3. Menambah pengetahuan pembaca mengenai penerapan *Dragonfly Algorithm* dalam menyelesaikan *Vehicle Routing Problem with Multi-Trips*.
4. Menambah referensi untuk penelitian yang berkaitan dengan *Dragonfly Algorithm* dan *Vehicle Routing Problem with Multi-Trips*.

I.6 Metodologi Penelitian

Untuk kelancaran penelitian, pertama-tama akan dibahas metodologi dari penelitian yang akan dilakukan. *Flow chart* digunakan untuk menggambarkan metodologi penelitian untuk memudahkan pemahaman akan tahap-tahap yang akan dilakukan dalam penelitian yang akan dilakukan. *Flow chart* dapat dilihat pada Gambar I.1 Berikut penjelasan dari setiap tahap.

1. Studi Literatur
Pada tahap ini akan dikumpulkan pengumpulan sumber dan referensi dari jurnal-jurnal penelitian atau buku-buku yang berkaitan dengan *Vehicle Routing Problem with Multi-Trips* dan *Dragonfly Algorithm* untuk membantu pemahaman akan metode dan masalah dalam penelitian.
2. Identifikasi dan Perumusan Masalah
Pada tahap ini akan diidentifikasi masalah yang ada, mengapa VRPMT bisa menyelesaikan masalah tersebut, mengapa digunakan *Dragonfly Algorithm* untuk menyelesaikan masalahnya, serta dibuat perumusan masalah dari identifikasi masalah yang telah dilakukan.
3. Pembatasan Masalah
Tahap ini dilakukan dengan tujuan agar penelitian lebih terfokus, dan memaparkan variable-variabel yang tidak menjadi pertimbangan dalam penelitian, atau hasil penelitian.
4. Penentuan Tujuan dan Manfaat Penelitian
Tahap ini dilakukan untuk memberi arahan akan penelitian, dan menyatakan manfaat yang bisa diambil berbagai pihak dari penelitian yang dilakukan.
5. Perancangan *Dragonfly Algorithm* untuk *Vehicle Routing Problem with Multi-Trips*

Pada tahap ini akan dilakukan perancangan dan penerapan algoritma DA untuk VRPMT berdasarkan hasil studi literatur yang telah dilakukan.

6. Verifikasi dan Validasi Rancangan Algoritma

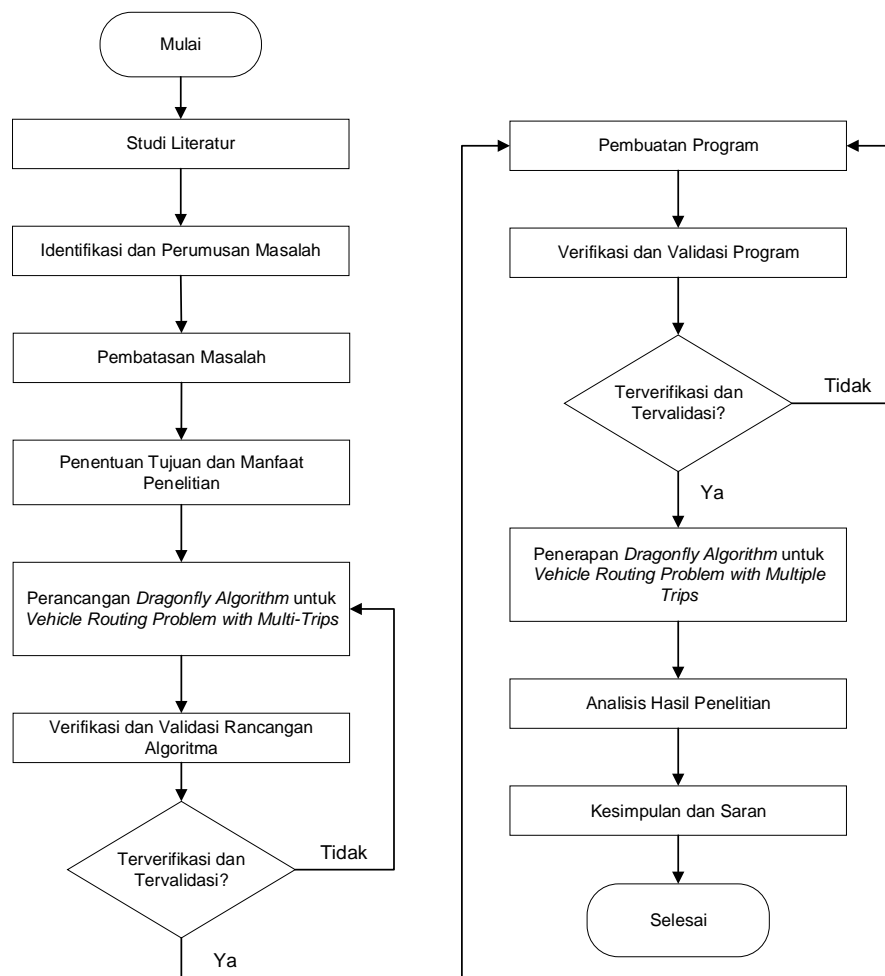
Tahap ini dilakukan untuk memastikan bahwa algoritma yang sudah dibuat sesuai dengan *Dragonfly Algorithm*, serta dapat menyelesaikan permasalahan VRPMT.

7. Pembuatan Program

Pada tahap ini akan dilakukan pembuatan program berdasarkan algoritma yang telah dirancang dan tervalidasi untuk menyelesaikan permasalahan VRPMT. Program dibuat menggunakan *software* Matlab.

8. Verifikasi dan Validasi Program

Tahap ini dilakukan untuk memverifikasi dan validasi bahwa program Matlab yang telah dibuat sesuai dengan algoritma yang sudah dirancang untuk menyelesaikan permasalahan VRPMT.



Gambar I.1 Flowchart Metodologi Penelitian

9. Penerapan *Dragonfly Algorithm* untuk *Vehicle Routing Problem with Multiple Trips*
Pada tahap ini program yang telah terverifikasi dan tervalidasi kemudian akan diimplementasikan terhadap kasus-kasus yang telah ditentukan dalam identifikasi dan perumusan masalah.
10. Analisis Hasil Penelitian
Pada tahap ini dilakukan analisis terhadap hasil penerapan program algoritma *Dragonfly* akan permasalahan VRPMT, serta performa algoritma tersebut bila dibandingkan dengan algoritma-algoritma lain yang telah ditetapkan sebagai *benchmark* penelitian.
11. Kesimpulan dan Saran
Pada tahap ini akan disajikan kesimpulan dari penelitian, serta pemberian saran akan penelitian-penelitian lain di masa depan yang berkaitan dengan penelitian yang telah dilakukan.

I.7 Sistematika Penulisan

Berikut sistematika penulisan yang memaparkan urutan penulisan laporan dari penelitian yang dilakukan.

BAB I PENDAHULUAN

Bab I memaparkan latar belakang masalah, identifikasi dan rumusan masalah, pembatasan masalah dan asumsi penelitian, tujuan penelitian, manfaat penelitian, metodologi penelitian, serta sistematika penulisan yang akan dilakukan dalam penelitian ini.

BAB II TINJAUAN PUSTAKA

Bab II memaparkan teori-teori yang akan digunakan dalam proses penelitian. Teori-teori yang dijabarkan adalah Hal-hal yang akan dijabarkan adalah teori mengenai *city logistics*, VRPMT, dan *Dragonfly Algorithm*.

BAB III PERANCANGAN ALGORITMA

Bab III memaparkan proses perancangan *dragonfly algorithm* untuk menyelesaikan permasalahan *Vehicle Routing Problem with Multi Trips*, proses *encoding* dan *decoding* yang dilakukan, proses pembaharuan posisi *dragonfly*, pembuatan algoritma, serta proses verifikasi dan validasi algoritma yang dibuat dengan cara penerapan algoritma yang telah dibuat.

BAB IV IMPLEMENTASI ALGORITMA

Bab IV memaparkan proses verifikasi dan validasi program yang dibuat dengan bahasa Matlab, implementasi *dragonfly algorithm* untuk menyelesaikan kasus hipotetik permasalahan *Vehicle Routing Problem with Multi Trips*, serta perbandingan performa *dragonfly algorithm* akan hasil penelitian-penelitian *benchmark genetic algorithm* dan *memetic algorithm*.

BAB V ANALISIS

Bab V memaparkan analisis dari hasil penelitian *dragonfly algorithm* untuk menyelesaikan permasalahan VRPMT. Hal-hal yang dianalisis adalah proses *encoding* dan *decoding* yang dilakukan, serta *Dragonfly Algorithm*.

BAB VI KESIMPULAN DAN SARAN

Bab VI memaparkan kesimpulan dari penelitian, serta memberikan saran untuk penelitian-penelitian di masa depan yang berkaitan dengan penelitian ini.