

# Penerapan Algoritma *Differential Evolution* untuk Penyelesaian Permasalahan *Vehicle Routing Problem with Delivery and Pick-up*

Ika Ayu Fajarwati dan Wiwik Anggraeni

Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111

E-mail: wiwik@is.its.ac.id

**Abstrak**—*Vehicle Routing Problem (VRP)* merupakan permasalahan optimasi kombinatorial kompleks yang memiliki peranan penting dalam manajemen sistem distribusi dengan tujuan meminimalkan biaya yang diperlukan, dimana penentuan biaya berkaitan dengan jarak dari rute yang ditempuh oleh armada distribusi. Ciri dari VRP yaitu penggunaan armada dengan kapasitas tertentu dan kegiatannya berpusat pada satu titik depot untuk melayani pelanggan pada titik-titik tertentu dengan jumlah permintaan yang diketahui. Kasus distribusi yang menggabungkan aktifitas pengiriman dan pengambilan produk termasuk dalam salah satu jenis VRP yaitu *Vehicle Routing Problem Delivery and Pick-Up (VRP-DP)*. Banyak metode yang dapat digunakan untuk menyelesaikan permasalahan VRP-DP, salah satunya adalah metode optimasi metaheuristik yaitu Algoritma *Differential Evolution* yang akan diperkenalkan dalam penelitian ini. Hasil yang diharapkan nantinya adalah rute distribusi optimal untuk armada perusahaan sehingga menghasilkan jarak tempuh dan tentunya total biaya yang minimal dalam memenuhi semua permintaan pelanggan.

**Kata Kunci**—*Vehicle Routing Problem Delivery Pick-Up, Differential Evolution Algorithm.*

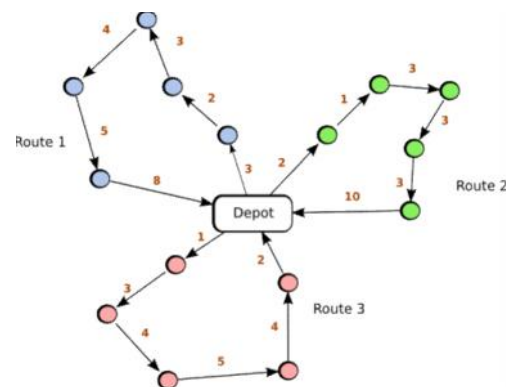
## I. PENDAHULUAN

**P**ERMASALAHAN *Vehicle Routing Problem (VRP)* adalah sebuah permasalahan optimasi kombinatorial yang kompleks, yang didefinisikan sebagai berikut: pencarian cara penggunaan sejumlah armada (kendaraan) secara efisien yang harus melakukan perjalanan untuk untuk mengantar dan/atau menjemput orang/barang pada sejumlah tempat. Setiap tujuan hanya boleh dilayani oleh satu armada saja. Hal ini dilakukan dengan mempertimbangkan kapasitas kendaraan dalam satu kali angkut, untuk meminimalkan biaya yang diperlukan. Asumsi bahwa penentuan biaya minimal erat kaitannya dengan jarak yang minimal.

Dalam praktiknya tidak sedikit kasus distribusi yang menggabungkan aktifitas pengiriman dengan pengambilan produk sekaligus. Beberapa contoh kasus distribusi yang menggabungkan kedua aktifitas pengiriman dan pengambilan produk contohnya distribusi *Liquefied Petroleum Gas (LPG)*, Air Minum Dalam Kemasan (AMDK), dan lain-lain. Model VRP yang menggabungkan kedua aktifitas pengiriman dan pengambilan produk dinamakan *Vehicle Routing Problem Delivery and Pick-Up (VRP-DP)*.

Beberapa metode atau pendekatan sudah banyak diajukan sebagai alternatif pemecahan masalah VRP untuk mendapatkan ongkos total minimum, diantaranya adalah

*integer programming* [1], *mixed integer programming* [2], *Tabu Search* [3], *genetic algorithm (GA)* [4], *simulated annealing* [5], *ant colony* [6] dan lain-lain. Dalam menggunakan exact optimization seperti integer programming, akan diperlukan waktu komputasi yang sangat lama terutama untuk problem berukuran besar (jika jumlah titik yang dilayani



Gambar 1. Visualisasi *Vehicle Routing Problem*

cukup banyak). Gambar 1 di bawah ini adalah visualisasi dari kasus *Vehicle Routing Problem*.

Dalam pengerjaan penelitian ini akan memperkenalkan satu metode optimasi lagi untuk menyelesaikan VRP yaitu algoritma *Differential Evolution (DE)*. DE termasuk dalam keluarga *Evolutionary Algorithm (EA)* yaitu *evolutionary population-based algorithm* dimana prinsip dan filosofi algoritmanya meniru perilaku evolusi biologi. DE diperkenalkan oleh Storn dan Price pada 1995. Meskipun begitu DE berbeda dengan algoritma EA dalam cara menentukan jarak dan arah proses pencarian dari populasi/solusi yang sekarang.

Menurut Storn dan Price, umumnya teknik optimasi praktis harus memenuhi tiga persyaratan. Pertama, metode harus menemukan nilai global optimum, terlepas dari sistem nilai-nilai parameter awal. Kedua, konvergensi harus cepat. Ketiga, program harus memiliki minimal parameter kontrol sehingga akan mudah untuk digunakan. Ketiga persyaratan tersebut yang mendasari munculnya DE. [7]

DE menyempurnakan kekurangan dari algoritma evolusi lainnya dengan strategi optimasi sederhana untuk proses optimasi yang cepat (waktu perhitungan cepat dengan iterasi yang sedikit untuk menemukan optimal global solution). Metode ini merupakan evolusi dari *Genetic Algorithm (GA)* dengan mengganti operator logika dengan operator matematis.

Penggantian operator ini membuat metode DE menjadi jauh lebih sederhana dan lebih mudah dalam pemakaiannya. DE juga memiliki kelebihan lain yaitu adanya evolusi yang dialami oleh setiap individu dalam populasi dimana diferensiasi dan crossover terjadi secara berurutan pada setiap individu yang terpilih acak dari populasi setiap waktu. Oleh karena itu DE lebih baik digunakan dalam penyelesaian permasalahan kompleks dibanding algoritma evolusioner lainnya. DE juga lebih efisien mengeksplorasi ruang keputusan dari GA ketika mengoptimasikan *multiple objective* [8].

Kinerja DE sebagai algoritma optimasi global dalam permasalahan continuous numerical minimization problems telah banyak dieksplorasi. DE menjadi alat yang powerful untuk memecahkan permasalahan seperti *fitting sophisticated models* [9], *performing model selection* [10], dan *optimizing portfolios under non-convex settings* [11].

Pertanyaan utama yang akan dijawab dalam penelitian ini adalah bagaimana menyelesaikan permasalahan VRP-DP yaitu menentukan rute distribusi yang optimal untuk armada perusahaan sehingga menghasilkan jarak tempuh dan tentunya total biaya yang minimal dalam memenuhi semua permintaan pelanggan.

## II. METODOLOGI PENELITIAN

### A. Identifikasi Masalah

Permasalahan yang diangkat dalam penelitian ini adalah penyelesaian permasalahan penentuan rute distribusi produk dengan menggunakan algoritma *Differential Evolution* (DE). VRP adalah problem pemrograman integer yang masuk kategori *NP-Hard Problem*, yang berarti usaha komputasi yang digunakan akan semakin sulit dan banyak seiring dengan meningkatnya ruang lingkup masalah, dalam hal ini adanya peningkatan jumlah titik yang akan dilayani oleh kendaraan. Oleh karena itu dibutuhkan metode optimisasi yang kompetitif untuk menyelesaikan permasalahan ini. DE sudah diaplikasikan dengan berhasil pada permasalahan-permasalahan penentuan rute perjalanan maupun bidang keteknikan lain. Oleh karena itu diharapkan DE juga akan berhasil menyelesaikan permasalahan penentuan rute distribusi produk.

### B. Studi Pustaka

Pada tahap ini akan dilakukan pemahaman teori-teori yang mendukung penelitian maupun informasi lain yang menunjang penelitian mengenai *Vehicle Routing Problem with Delivery and Pick-up*, *Differential Evolution Algorithm*, dan lain-lain. Teori dan informasi diatas dicari dari berbagai sumber seperti jurnal, buku panduan, tesis, tugas akhir, seminar konferensi atau prosiding nasional dan internasional. Hasil yang didapatkan dari studi literatur yaitu informasi mengenai beberapa studi kasus permasalahan VRP-DP yang telah berhasil diselesaikan oleh metode-metode tertentu oleh peneliti-peneliti terdahulu.

### C. Pengumpulan Data

Tahap ini bertujuan untuk mengumpulkan data yang nantinya digunakan untuk pengujian algoritma. Data yang

diambil adalah sebuah data set yang berisi data permintaan dari 21 pelanggan beserta matriks jarak dari lokasi depot dan pelanggan. Data set tersebut didapatkan dari penelitian sebelumnya yang menyelesaikan permasalahan dengan kasus dan data yang sama menggunakan algoritma *Dethloff's insertion* dan *Tabu Search* [12]. Kemudian terdapat juga data dengan jumlah node yang lebih banyak yaitu sebanyak 101 node untuk uji coba pada kasus besar.

### D. Permodelan Matematis dari Permasalahan

Model matematis yang disusun akan memuat aspirasi/kriteria-kriteria sebagai fungsi tujuan dan beberapa fungsi kendala. Fungsi tujuannya adalah minimasi total biaya perjalanan oleh armada. Beberapa kendala yang harus dipertimbangkan adalah kapasitas kendaraan, besar permintaan dari pelanggan dan lain-lain.

### E. Perancangan Algoritma

Tidak sama dengan GA pada umumnya, DE bergerak dari solusi awal random sebanyak populasi tertentu dengan performansi kurang baik yang diubah melalui proses mutasi dan crossover. Setiap iterasinya akan dipilih solusi terbaik sebagai induk dengan proses seleksi untuk membangkitkan solusi anak pada iterasi berikutnya. Pemrograman komputer perlu dilakukan karena problem routing adalah problem yang besar yang dalam penyelesaiannya membutuhkan banyak iterasi dan tidak mungkin dilakukan secara manual. Implementasi dilakukan dengan memasukkan data yang dibutuhkan sebagai masalah yang akan diselesaikan.

### F. Uji coba Algoritma

Uji coba dilakukan dengan pengujian algoritma terhadap data set yang telah ditentukan dengan cara me-running program dengan memasukkan input parameter yang berbeda-beda dan dilakukan berulang kali. Selain itu, pengujian dikatakan berhasil jika semua batasan terpenuhi.

### G. Analisa dan Interpretasi Hasil

Analisis hasil berisi penjelasan tentang semua hasil output yang didapat dari ujicoba yang dilakukan dengan input parameter yang berbeda-beda. Kemudian dibandingkan dengan hasil dari penelitian sebelumnya.

## III. DESAIN DAN IMPLEMENTASI ALGORITMA

### A. Deskripsi Data

Berikut ini adalah daftar variable input dari beserta asumsinya yang akan digunakan pada saat menjalankan algoritma:

1. Data set demand dan pickup untuk-untuk masing-masing node yang ditunjukkan pada tabel 1.
2. Data jarak antar node dalam bentuk matriks seperti yang ditunjukkan pada lampiran A.
3. Kendaraan angkut adalah Mitsubishi Colt Diesel 100PS 2.4 ton / 7.9 m3, dengan biaya sewa Rp. 500.000,- [13] dengan asumsi sewa per rute.

Tabel 1.  
Data node beserta jumlah delivery dan pickup

Node	Delivery Ammount	Pickup Ammount
1 (Depot)	0	0
2	6	8
3	7	5
4	5	2
5	2	0
6	3	4
7	6	9
8	1	6
9	2	6
10	9	15
11	12	3
12	15	14
13	6	15
14	7	3
15	8	16
16	10	5
17	11	17
18	13	13
19	15	17
20	10	6
21	7	9

4. Kapasitas maksimal kendaraan angkut yaitu 40 unit tabung gas.
5. Biaya BBM kendaraan tersebut per liter Rp.4500,- dengan asumsi konsumsi kendaraan tersebut per-liter mampu menempuh jarak 13 km [14].

**B. Perancangan Permodelan Permasalahan**

Sebelum merancang algoritma *Differential Evolution*, dibutuhkan perhitungan-perhitungan fungsi tujuan, kendala dan perhitungan asumsi yang dibutuhkan seperti perhitungan total biaya sewa kendaraan, biaya perjalanan dan total biaya perjalanan. Berikut adalah perhitungannya :

1. Biaya sewa kendaraan = jumlah kendaraan x biaya sewa per-kendaraan dengan jumlah kendaraan jumlah route solusi yang terbentuk oleh algoritma.

$$2. \text{Biaya Perjalanan} = \left( \frac{\text{Total Jarak}}{\text{Kemampuan Tempuh per-liter}} \right) \times \text{Harga BBM per-liter}$$

3. Total Biaya=Biaya Sewa Kendaraan + Biaya Perjalanan

**C. Perancangan Algoritma Differential Evolution**

Perancangan algoritma *Differential Evolution* yang diusulkan dalam penelitian ini memiliki empat tahapan utama, yaitu tahap Inisialisasi, Mutasi, Crossover dan Seleksi.

**1. Inisialisasi**

Tahap ini merupakan tahap pencarian solusi awal permasalahan VRPDP dengan membangkitkan gen berupa angka 0-1 secara acak sebanyak jumlah node menjadi kromosom berulang kali sampai dengan ukuran populasi yang diinginkan. Pada tahap inisialisasi dibutuhkan variable input berupa :

- a. Ukuran populasi yang diinginkan atau Population Size atau biasa di notasi kan sebagai N.

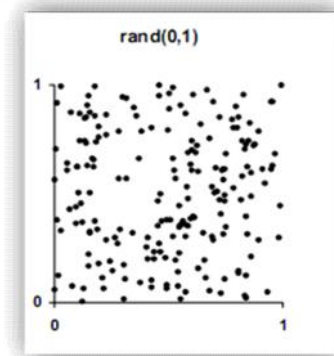
- b. Matriks jarak, untuk mencari jumlah node keseluruhan. Jumlah node dapat dibaca dari dimensi matriks jarak.

$[a \ b] = \text{size}(\text{jarak\_node})$  maka  $a = \text{jumlah node}$ .

Populasi solusi awal merupakan kumpulan kromosom dengan index angka yang terambil secara acak sebanyak jumlah node. Ketentuan dari kromosom yang akan dihasilkan adalah sebagai berikut :

- Gen yang dibangkitkan yaitu 0-1 dan diurutkan
- Pembentukan kromosom harus dimulai dari angka 1 dan diakhiri dengan angka 1 lagi.
- Tidak memilih kembali angka yang sudah terpilih.

Populasi solusi awal hasil inisialisasi (populasi parents) yang telah dibangkitkan kemudian dievaluasi dengan cara menghitung nilai fungsinya. Nilai fungsi (fitness) populasi parents tersebut nantinya akan diseleksi dengan nilai fungsi yang dihasilkan oleh populasi mutan (populasi hasil mutasi). Visualisasi tahapan inisialisasi individu yang tersebar secara random ditunjukkan pada gambar 2.



Gambar. 2. Inisialisasi Populasi Fungsi Random

**2. Mutasi**

Pembentukan vector muatan ( $V_i$ ) dilakukan dengan mengkombinasikan 3 vektor yang dipilih secara acak dari populasi vector yang telah ada dengan F sebagai factor skala pembeda. Nilai vector solusi pertama dipilih sekali untuk semua vector yang ada di populasi yang sama, sedangkan populasi vector kedua dan ketiga dipilih untuk setiap vector muatan yang akan dibentuk. Proses pembentukan vector mutan ini dilakukan sebanyak jumlah populasi.

**3. Kawin-Silang**

Crossover atau kawin silang menggabungkan vector dari populasi solusi awal (populasi parents/  $X_i$ ) dengan populasi vector mutan (populasi hasil mutasi/  $V_i$ ), kemudian dilanjutkan dengan membangkitkan bilangan random sebanyak jumlah aset. Apabila nilai CR yang telah ditentukan pada inisialisasi lebih besar dari nilai random, maka vector  $i$  akan tetap berisi  $X_i$ . Begitu pula sebaliknya bila CR kurang dari atau sama dengan nilai rand, maka vector  $i$  akan berisi  $V_i$ . Mekanisme crossover yang ditentukan oleh fungsi random tersebut ternyata memiliki kelemahan yaitu sangat memungkinkan individu terbaik dari hasil tahapan sebelumnya terlewatkan karena tidak

terpilih berdasarkan fungsi random untuk mengisi populasi baru [15].

Gambar 3 di atas menunjukkan ilustrasi perkawinan silang yang dilakukan dalam tahapan ini yaitu dengan mengambil 2 parents kromosom dari populasi mutan kemudian pada tiap persilangan akan menghasilkan 4 anak dengan penjelasan sebagai berikut [16]:

Parent A	A1	A2
Parent B	B1	B2
anak 1	A1	B2
anak 2	B1	A2
anak 3	B2	A1
anak 4	A2	B1

Gambar. 3. Ilustrasi perkawinan silang

Diawali dengan memilih secara acak 2 kromosom (parent1 dan parent2) di dalam satu populasi yang akan di kawin-silang dan batas pemotongan. Batas pemotongan minimal yang terjadi haruslah antara 1 sampai dengan jumlah\_node dikurangi 3 (kk-3). Variabel *idxkrom* adalah penggabungan antara hasil perpotongan dari parent 1 dengan hasil perpotongan dari parent 2. Setelah perkawinan silang selesai sampai dengan *pop\_size*, barulah semua hasil perkawin-silangan tersebut di kumpulkan jadi satu.

Kumpulan hasil crossover tersebut kemudian disatukan dengan *pop\_campur1* dan di beri nama variable *pop\_campur2* dimana populasi parents, populasi mutan dan populasi crossover digabungkan menjadi satu.

4. Seleksi Akhir

Kumpulan populasi yang telah disatukan kemudian di urutkan sesuai total biaya hasil perhitungan fitness masing-masing kromosom secara *ascending* (dari yang terkecil). Kemudian diambil kromosom terbaik sebanyak ukuran populasi.

D. Kriteria Pemberhentian

Iterasi pencarian nilai fungsi minimum akan terus berjalan hingga kriteria pemberhentian iterasi terpenuhi. Dalam kasus ini menggunakan 2 stopping criterion yaitu :

- Berhenti jika selisih total biaya kromosom terburuk dengan total biaya kromosom terbaik dari populasi gabungan adalah epsilon, yaitu bilangan yang sangat kecil dan mendekati nol.
- Berhenti hingga maksimal iterasi yang di inginkan

Dua kriteria pemberhentian di atas diterapkan dalam perancangan algoritma DE dalam penelitian ini, diharapkan lebih optimal dalam pencarian solusi.

IV. UJICOBAL ALGORITMA DAN ANALISA HASIL

Algoritma *Differential Evolution* memiliki kelebihan yaitu terdapat proses crossover yang menghasilkan 4 anak, sehingga alternatif solusi yang ditawarkan muncul lebih banyak. Selain itu pada tahap seleksi, populasi yang dihasilkan dari tiap

tahap algoritma dikumpulkan kembali untuk diambil solusi terbaik, sehingga tidak ada solusi terbaik yang terlewatkan. Dari hasil ujicoba algoritma yang dilakukan berulang kali tanpa batasan iterasi atau mencapai konvergen, sehingga mendapatkan solusi yang paling optimal, didapatkan hasil yang ditunjukkan pada tabel 2 di bawah ini :

Tabel 2.

Solusi optimal oleh Algoritma *Differential Evolution*

Algoritma <i>Differential Evolution</i>		
Kendaraan ke-	Rute	Jarak Tempuh (km)
1	1 – 10 – 4 – 8 – 5 – 19 – 1	43.25
2	1 – 18 – 14 – 13 – 21 – 1	25.35
3	1 – 3 – 2 – 20 – 6 – 15 – 1	32.6
4	1 – 9 – 17 – 12 – 1	25.74
5	1 – 11 – 16 – 7 – 1	9.6
<b>Biaya Sewa Kendaraan (Rp.)</b>		2.500.000
<b>Total Jarak Tempuh (km)</b>		136.54
<b>Biaya Perjalanan (Rp.)</b>		47.789
<b>Total Biaya (Rp.)</b>		2.547.789

Untuk lebih mengetahui keunggulan algoritma, maka perlu adanya perbandingan solusi yang dihasilkan oleh algoritma DE dengan algoritma lainnya. Pada penelitian sebelumnya, kasus dan dataset yang sama juga diselesaikan oleh algoritma lain yaitu *Dethloff's insertion* dan algoritma *Tabu Search*. Tabel 3 di bawah menunjukkan perbandingan solusi yang dihasilkan oleh algoritma *Dethloff's insertion* dan algoritma *Tabu Search*.

Tabel 3.

Solusi optimal oleh Algoritma *Dethloff's insertion* dan *Tabu Search*

Algoritma <i>Dethloff's insertion</i>		
Kendaraan ke-	Rute	Jarak Tempuh (km)
1	1 – 3 – 7 – 4 – 18 – 20 – 1	41.5
2	1 – 13 – 5 – 2 – 19 – 1	34.6
3	1 – 17 – 12 – 1	24.12
4	1 – 9 – 16 – 1	18.28
5	1 – 8 – 14 – 11 – 1	20.36
6	1 – 6 – 10 – 15 – 1	9.63
<b>Biaya Sewa Kendaraan (Rp.)</b>		3.000.000
<b>Total Jarak Tempuh (km)</b>		148.53
<b>Biaya Perjalanan (Rp.)</b>		51.414
<b>Total Biaya (Rp.)</b>		3.051.414
Algoritma <i>Tabu Search</i>		
Kendaraan ke-	Rute	Jarak Tempuh (km)
1	1 – 10 – 15 – 6 – 14 – 1	17.15
2	1 – 10 – 15 – 6 – 14 – 1	17.15
3	1 – 17 – 13 – 12 – 20 – 1	25.35
4	1 – 19 – 5 – 2 – 1	32.49
5	1 – 3 – 7 – 9 – 16 – 1	35.96
<b>Biaya Sewa Kendaraan (Rp.)</b>		2.500.000
<b>Total Jarak Tempuh (km)</b>		151.8
<b>Biaya Perjalanan (Rp.)</b>		52.546
<b>Total Biaya (Rp.)</b>		2.552.546

Berikut tabel 4 menunjukkan perbandingan solusi antara ketiga algoritma yang dibandingkan :

Tabel 4.  
Perbandingan solusi antar tiga algoritma

	Algoritma		
	<i>Dethloff's insertion</i>	<i>Tabu Search</i>	<i>Differential Evolution</i>
<b>Jumlah Rute/Jumlah Kendaraan Sewa</b>	6	5	5
<b>Total Jarak Tempuh</b>	148,53	151,8	136,54
<b>Total Biaya Perjalanan</b>	3.051.414	2.552.546	2.547.789

Perbandingan solusi oleh *Dethloff's insertion* dengan solusi oleh *Differential Evolution* :

1. Jumlah kendaraan yang dihasilkan oleh algoritma *Dethloff's insertion* adalah sebanyak 6 unit kendaraan dengan 1 rute untuk masing-masing kendaraan, sedangkan dengan algoritma *Differential Evolution*, dibutuhkan 5 kendaraan untuk menempuh perjalanan distribusi. Maka terjadi penurunan jumlah kendaraan sebanyak 1 unit kendaraan. Dengan kata lain rute yang berhasil dibentuk juga berkurang sebanyak 1 rute.
2. Total jarak tempuh yang dihasilkan oleh algoritma *Dethloff's insertion* adalah 148.53 km, sedangkan dengan algoritma *Differential Evolution* total jarak tempuhnya adalah 136,54 km. Maka terjadi penurunan angka pada total jarak tempuh sebesar 11,99 km. Persentase penurunan total jarak tempuh :

$$\frac{148.53 - 136.54}{136.54} \times 100\% = 0.09\%$$

Maka, persentase penurunan total jarak tempuh yang terjadi adalah 0.09 %.

3. Total biaya perjalanan yang dihasilkan oleh algoritma *Dethloff's insertion* adalah Rp. 3.051.414,-, sedangkan dengan algoritma *Differential Evolution* total biaya perjalanannya adalah Rp. 2.547.789,-. Maka terjadi penurunan nominal total biaya perjalanan yaitu sebesar Rp. 503.625,-. Persentase penurunan total biaya perjalanan :

$$\frac{503.625}{3.051.414} \times 100\% = 16.5\%$$

Maka, persentase penurunan total biaya perjalanan yang terjadi adalah 16.5 %.

Perbandingan solusi oleh *Tabu Search* dengan *Differential Evolution* :

1. Jumlah kendaraan yang dihasilkan oleh algoritma *Tabu Search* adalah sebanyak 5 unit kendaraan dengan 1 rute untuk masing-masing kendaraan, dan dengan algoritma *Differential Evolution* juga sebanyak 5 kendaraan. Maka tidak terjadi perubahan untuk jumlah kendaraan.
2. Total jarak tempuh yang dihasilkan oleh algoritma *Tabu Search* adalah 151.8 km, sedangkan dengan algoritma *Differential Evolution* total jarak tempuhnya adalah 136,54 km. Maka terjadi penurunan angka pada total jarak tempuh sebesar 15,26 km. Persentase penurunan total jarak tempuh :

$$\frac{151.8 - 136.54}{136.54} \times 100\% = 0.1117\%$$

Maka, persentase penurunan total jarak tempuh yang terjadi adalah 0.1117 %.

3. Total biaya perjalanan yang dihasilkan oleh algoritma *Tabu Search* adalah Rp. 2.552.546,-, sedangkan dengan algoritma *Differential Evolution* total biaya perjalanannya adalah Rp. 2.547.789,-. Maka terjadi penurunan nominal total biaya perjalanan yaitu sebesar Rp. 4.757,-. Perbandingan nya sangat kecil karena kendaraan yang digunakan juga 5 unit.

$$\frac{4.757}{2.552.546} \times 100\% = 0.18\%$$

Maka, persentase penurunan total biaya perjalanan yang terjadi adalah 0.18 %.

## V. KESIMPULAN

Beberapa hal yang dapat disimpulkan dari pengerjaan penelitian ini adalah sebagai berikut :

1. Dalam penyelesaian permasalahan *Vehicle Routing Problem with Delivery and Pick-up* (VRP-DP) dengan dataset 21 pelanggan, algoritma *Differential Evolution* yang diusulkan menghasilkan total 5 route (5 unit kendaraan sewa), total jarak 136,54 km dan total biaya perjalanan sebesar Rp. 2.547.789,00.
2. Algoritma *Differential Evolution* mampu mengurangi jumlah rute dari solusi yang dihasilkan oleh algoritma *Dethloff's insertion* sebesar 1 unit, menurunkan total jarak tempuh sebesar 0.09 %, dan menghasilkan total biaya perjalanan yang lebih rendah dengan selisih Rp. 503.625,- dengan presentase sebesar 16.5 %.
3. Algoritma *Differential Evolution* mampu menurunkan total jarak tempuh sebesar 0.117 %, dan menghasilkan total biaya perjalanan yang lebih rendah dengan selisih Rp. 4.757,- dengan presentase penurunan sebesar 0.18 %. Selisih yang dihasilkan tidak terlalu besar dikarenakan jumlah route dan jumlah kendaraan yang harus disewa, yang dihasilkan oleh Algoritma *Differential Evolution* tidak mengalami perubahan.
4. Algoritma *Differential Evolution* dapat menghasilkan solusi yang lebih optimal dibanding kedua algoritma pembandingan karena terdapat proses crossover yang menghasilkan 4 anak, sehingga alternatif solusi yang ditawarkan muncul lebih banyak. Selain itu pada tahap seleksi, populasi yang dihasilkan dari tiap tahapan algoritma dikumpulkan kembali untuk diambil solusi terbaik, sehingga tidak ada solusi terbaik yang terlewatkan. Dengan begitu Algoritma *Differential Evolution* terbukti dapat diimplementasikan dalam kasus NP-hard.



## UCAPAN TERIMA KASIH

Penulis I.A.F mengucapkan terima kasih kepada kepala jurusan dan dosen Sistem Informasi, yang telah membina dan memberikan fasilitas untuk penelitian, serta semua pihak yang telah membantu penulis dalam menyelesaikan penelitian ini.

## DAFTAR PUSTAKA

- [1] Kulkarni R.V., Bhawe P.R. "Integer programming formulations of *Vehicle Routing Problems*". *European Journal of Operational Research*, Vol. 20/1, (1985) 58-67.
- [2] Longo H., De Aragao M.P., Uchoa E. "Solving capacitated arc routing problems using a transformation to the CVRP". *Computers and Operations Research*, Vol. 33/6, (2006) 1823-1837.
- [3] Montane F.A.T., Galvao R.D. "A *Tabu Search* algorithm for the *Vehicle Routing Problem* with simultaneous pick-up and delivery service". *Computers and Operations Research*, Vol. 33/3, (2006)595-619.
- [4] Baker B.M., Ayeche M.A. "A genetic algorithm for the *Vehicle Routing Problem*". *Computers and Operations Research*, Vol. 30/5, (2003) 787-800.
- [5] Tavakkoli-Moghaddam R., Safaei N., Kah M.M.O., Rabbani M. "A New Capacitated *Vehicle Routing Problem* with Split Service for Minimizing Fleet Cost by Simulated Annealing". *Journal of the Franklin Institute*, Vol. 344/5, (2007)406-425.
- [6] Bell J.E., McMullen P.R. "Ant colony optimization techniques for the *Vehicle Routing Problem*". *Advanced Engineering Informatics*, Vol. 18/1, (2004) 41-48.
- [7] Storn, R and Price, K " *Differential Evolution - A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*". Technical Report, International Computer Science Institute, Berkley, (1995).
- [8] Tusar, Tea, Bodgan F. " *Differential Evolution* Versus Genetic Algorithms in Multi-objective Optimization". *Department of Intelligent Systems : Springer*, (2007) 257-271.
- [9] M. Gilli and E. Schumann "Heuristic Optimization in Financial Modeling". COMISEF wps-007 09/02/2009, (2009).
- [10] G. Maringer and M. Meyer. "Smooth transition autoregressive models: New approaches to the model selection problem". *Studies in Nonlinear Dynamics & Econometrics*, Vol. 12/1, (2008)1-19.
- [11] Krink and S. Paterlini. "Multiobjective optimization using *Differential Evolution* for real-world portfolio optimization". *Computational Management Science*, (2011) 157-179.
- [12] Setiawan, Maramis. " *Vehicle Routing Problem* With Simultaneous Pick-Ups And Deliveries Menggunakan Algoritma *Tabu Search* dengan Bantuan Google Maps". Tugas Akhir, Jurusan Teknik Industri, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember, Surabaya, (2009).
- [13] Tarif Sewa Mobil. [http://www.sewamobilbox.com/?page\\_id=6](http://www.sewamobilbox.com/?page_id=6) diakses April 2012
- [14] Esti, Neri Anggrita. 2007. Perancangan Sistem Distribusi PT. TRIJAYA di Surabaya. <http://digilib.petra.ac.id/viewer.php?submit15.x=25&submit.y=24&submit=prev&page=2&qual=high&submitval=prev&fname=%2Fjunkpe%2Fs1%2Ftmi%2F2007%2Fjunkpe-ns-s1-2007-25401096-10487-trijaya-chapter4.pdf> diakses April 2012.
- [15] Awalul, Heri. Pengembangan Algoritma *Differential Evolution* Untuk Penyelesaian Permasalahan *Vehicle Routing Problem* Simultaneous Deliveries Pick-Up With Time Windows (Vrpsdptw). Master theses, Jurusan Teknik Industri, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember, Surabaya, (2011).
- [16] Salim, Emmil. Algoritma Genetika. <http://www.scribd.com/doc/95737964/ALGORITMA-GENETIKA> diakses Juni, 2012