

Implementasi *artificial bee colony* pada *mix fleet vehicle routing problem* (MFVRP) untuk optimalisasi pendistribusian produk

Elis Dwi Wulandari¹, Darmawan Satyananda²
Universitas Negeri Malang
vin_elboygirl@yahoo.com
darmawan.satyananda.fmipa@um.ac.id

ABTRAK: *Mix Fleet Vehicle Routing Problem* (MFVRP) merupakan bagian dari permasalahan *Capacitated Vehicle Routing Problem*. Tujuan utama dari MFVRP adalah meminimalisasi jarak tempuh dan banyak kendaraan dengan memperhatikan setiap kendaraan yang memiliki kapasitas berbeda. *Artificial Bee Colony* merupakan algoritma yang didasarkan kecerdasan berkelompok dari lebah dalam mencari sumber makanan. Terdapat tiga bagian dalam kelompok lebah yaitu Lebah Pekerja (*Employed*), Lebah Penjaga (*Onlooker*) dan Lebah Pengintai (*Scout*). Implementasi Algoritma *Artificial Bee Colony* untuk memudahkan menyelesaikan permasalahan MFVRP menggunakan Borland Delphi 7.

Kata kunci : *Graph, Vehicle Routing Problem (VRP), Mix Fleet Vehicle Routing Problem (MFVRP), Artificial Bee Colony, distribusi.*

ABSTRACT: *Mix Fleet Vehicle Routing Problem* (MFVRP) is part of the *Capacitated Vehicle Routing Problem*. The main objective of MFVRP is minimizing the travel distance and the number of vehicles considering vehicle that has a different capacity. *Artificial Bee Colony* algorithm is based on swarm intelligence of bees searching for food sources. There are three sections in the group of bees which has a different task, named *Bee Workers (Employed)*, *Bee Keeper (Onlooker)* and *Bee Surveillance (Scout)*. Implementation of *Artificial Bee Colony* Algorithm is to facilitate solving the problem uses Borland Delphi 7.

Keywords: *Graph, Vehicle Routing Problem (VRP), Mix Fleet Vehicle Routing Problem (MFVRP), Artificial Bee Colony, distribution.*

Pendistribusian atau pemindahan barang yang efektif dan efisien yaitu sampainya barang kepada konsumen di tempat dan waktu yang tepat akan mempengaruhi kepuasan dan rasa tertarik konsumen atau *customer*. Hal ini akan meningkatkan volume penjualan bagi perusahaan sehingga diperoleh pendapatan bagi perusahaan, pencapaian tujuan dan mencapai laba yang maksimal. Jumlah konsumen yang tidak sedikit dan letak tempat tujuan distribusi yang menyebar sering kali mengakibatkan jalur atau rute distribusi yang tidak optimal. Permasalahan distribusi dapat digambarkan sebagai graph dengan titik dan sisi serta dapat diselesaikan dengan *Vehicle Routing Problem (VRP)*. *Vehicle Routing Problem (VRP)* merupakan suatu permasalahan optimalisasi untuk menentukan beberapa rute untuk sekumpulan kendaraan yang harus melayani sejumlah *customer* dari depot pusat. Salah satu varian VRP adalah CVRP. CVRP (*Capacitated Vehicle Routing Problem*) digolongkan menjadi 2 macam yaitu *Homogenous Vehicle Routing Problem (or Uniform Fleet VRP)* dan *Heterogenous Vehicle Routing Problem (or Mixed Fleet VRP)*. Dalam *Mix Fleet VRP* dimana pada

-
1. Elis Dwi Wulandari adalah mahasiswa Jurusan Matematika FMIPA Universitas Negeri Malang
 2. Darmawan Satyananda adalah dosen Jurusan Matematika FMIPA Universitas Negeri Malang

saat pengiriman barang, depot memiliki kendaraan dengan kapasitas barang yang berbeda-beda (Suthikarnnarunai, 2008). Salah satu metode untuk menyelesaikan permasalahan MFVRP adalah *Artificial Bee Colony (ABC)*. *Artificial Bee Colony (ABC)* adalah sebuah metode optimalisasi yang terinspirasi oleh perilaku nyata lebah madu menemukan dan berbagi informasi sumber-sumber makanan kepada lebah lain di sarangnya, diperkenalkan oleh Karaboga pada tahun 2005 (Karaboga, 2005). Akan dibahas algoritma *Artificial Bee Colony (ABC)* yang memberikan alternatif kemungkinan dalam menyelesaikan masalah MFVRP.

HASIL YANG DIHARAPKAN

1. Mengimplementasikan langkah-langkah Algoritma *Artificial Bee Colony* dalam penyelesaian permasalahan *Mix Fleet Vehicle Routing Problem* MFVRP yaitu permasalahan pendistribusian barang.
2. Implementasi langkah-langkah Algoritma *Artificial Bee Colony* dengan bahasa pemrograman dalam pembuatan program ABC untuk MFVRP.

HASIL DAN PEMBAHASAN

Graph G didefinisikan sebagai pasangan himpunan (V, E) , ditulis dengan notasi $G = (V, E)$, yang dalam hal ini V adalah himpunan tidak kosong dari titik-titik (*vertex*) dan E adalah himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang titik (Wilson, 2000: 39). Pada permasalahan VRP, graph G dianggap sebagai peta yang menjelaskan kemungkinan jalur yang dapat dilewati dengan setiap titik mewakili depot dan *customer*, setiap sisi pada graph menunjukkan jalan yang menghubungkan antar titik dan setiap bobot pada sisi mewakili jarak. Graph berbobot diterapkan dalam permasalahan MFVRP, yaitu setiap jalan (sisi pada graph) yang menghubungkan antar *customer* memiliki bobot/ nilai jarak dalam satuan tertentu.

- ***Mix Fleet Vehicle Routing Problem***

MFVRP didasarkan pada himpunan yang berisi dari suatu graph G dimana suatu titik mewakili sebuah depot dan titik-titik yang lain mewakili beberapa *customer*. Menurut Prive (2005), suatu himpunan n *customer* dilayani oleh k jenis kendaraan, dengan $k = 1, 2, \dots, K$ dan banyak kendaraan dengan jenis k jenis dinotasikan dengan m_k . Setiap *customer* mempunyai permintaan pengiriman barang sejumlah d_i dengan $i = 1, 2, \dots, n$. Setiap kendaraan berangkat dari depot dan kembali lagi ke depot. Sehingga solusi yang dihasilkan dari permasalahan MFVRP adalah himpunan rute dimana setiap *customer* dikunjungi oleh satu kendaraan dan total permintaan dari seluruh *customer* di sebarang rute paling banyak adalah dengan total jarak tempuh yang minimum. Kapasitas maksimum setiap kendaraan berbeda-beda tergantung jenis kendaraan, jumlah kendaraan yang dibutuhkan depot untuk mendistribusikan barang diasumsikan selalu tersedia dengan kapasitas yang berbeda-beda tergantung jenis kendaraan yang dipakai. Komoditi yang didistribusikan sejenis. Tiap rute dilayani oleh kendaraan yang berbeda sehingga jumlah rute sama dengan jumlah kendaraan yang dibutuhkan.

Untuk memperkecil jumlah dari himpunan rute, maka setiap *customer* harus dilayani tepat satu kali oleh tepat satu kendaraan. Status bahwa kendaraan melewati *i* ke *j* dinyatakan sebagai berikut :

$$x_{ij}^k = \begin{cases} 1, & \text{jika kendaraan } k \text{ dijalankan dari titik } i \text{ ke } j, i \neq j \\ 0, & \text{untuk yang lain} \end{cases}$$

Menurut Sarwadi dan Krismi (2004) dan Brajevic (2011) serta Suthikarnnarunai (2008) formulasi matematika dari VRP dengan sejumlah *customer* yang bertujuan meminimumkan total jarak tempuh sesuai jumlah kendaraan yang tersedia adalah sebagai berikut :

Fungsi tujuan

$$\text{Min } \sum_{(i,j) \in N} c_{ij} \sum_{k=1}^K x_{ij}^k, \text{ dengan } i \neq j$$

Dengan:

$$\begin{aligned} \text{Batasan 1 : } \sum_{k=1}^K x_{ij}^k &= 1, \forall i \in N, j \in N \setminus \{0\}, i \neq j \\ \sum_{k=1}^K x_{ij}^k &= 1, \forall i \in N \setminus \{0\}, j \in N, i \neq j \end{aligned}$$

Setiap *customer* hanya dikunjungi satu kali dan hanya oleh satu kendaraan

$$\text{Batasan 2 : } \sum_{i=1}^n d_i \leq Q$$

Total permintaan dari setiap *customer* dalam satu rute tidak boleh melebihi kapasitas kendaraan

$$\text{Batasan 3 : } \sum_{i \in N} x_{ij}^k \sum_{h \in N} x_{hj}^k = 0, \forall h \in N \setminus \{0\}, \forall k \in V$$

Setiap kendaraan harus meninggalkan *customer* yang telah dikunjungi

$$\begin{aligned} \text{Batasan 4 : } \sum_{k=1}^K x_{0j}^k &= 1, \forall j \in N \setminus \{0\} \\ \sum_{k=1}^K x_{i0}^k &= 1, \forall i \in N \setminus \{0\} \end{aligned}$$

Setiap kendaraan yang meninggalkan *depot* harus kembali ke *depot*.

$$\text{Batasan 5 : } \sum_{j=1}^n q_j \leq Q$$

Jumlah volume yang diambil dalam satu rute tidak boleh melebihi kapasitas kendaraan.

$$\text{Batasan 6 : } \sum_{i=0}^n d_i (\sum_{j=0}^n x_{ij}^k) \leq Q_k, k = 1, 2, \dots, K$$

Permintaan masing-masing *customer* tidak boleh melebihi kapasitas kendaraan yang dipakai untuk melayani *customer* tersebut.

$$\text{Batasan 7 : } \sum_{i=0}^n d_i (\sum_{j=0}^n x_{ij}^k) \leq \sum_{k=1}^K Q_k, \forall i \in N, k = 1, 2, \dots, K$$

Jumlah permintaan seluruh *customer* tidak boleh melebihi jumlah seluruh kapasitas kendaraan yang dipakai.

Keterangan :

c_{ij} = jarak dari pelanggan *i* ke pelanggan *j*

x_{ij} = status kendaraan dari melewati *i* ke *j*

d_i = Jumlah barang yang harus dikirim pada tiap *customer* *i* dengan $i \in N$

q_j = jumlah / kuantitas volume pelanggan ke *j*

Q = kapasitas kendaraan

Q_k = Kapasitas kendaraan ke- *k*

- **Artificial Bee Colony**

Menurut Karaboga (2005), terdapat dua konsep dasar pelaksanaan bersama dari suatu kawanan yaitu *self-organization* dan pembagian tenaga kerja. Pelaksanaan bersama merupakan kebutuhan dan properti untuk memperoleh perilaku kecerdasan kawanan seperti sistem penyelesaian masalah pendistribusian yang diatur sendiri dan menyesuaikan dengan

kondisi lingkungan. *Self-organization* merupakan suatu mekanisme atau sistem yang memiliki aturan dimana terdapat interaksi antar individu dalam kawanan.

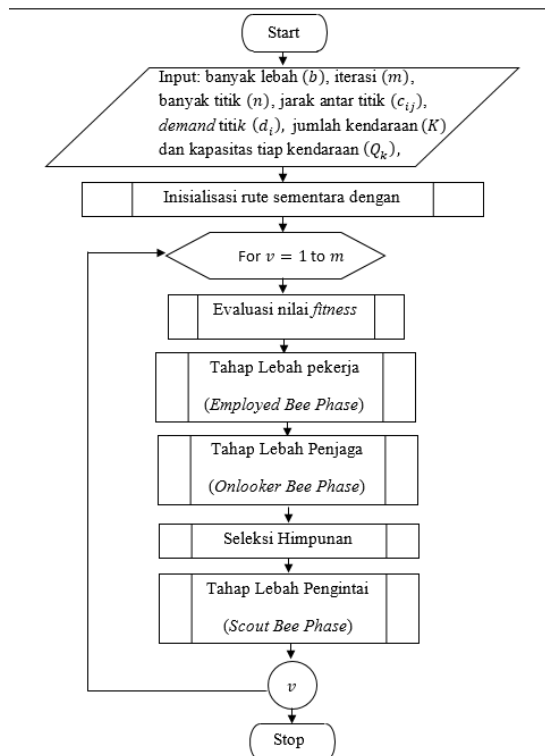
Dari perilaku kecerdasan kawanan tersebut, ide dasar Algoritma *Artificial Bee Colony* diperoleh dari perilaku kawanan lebah madu dimana Algoritma ABC digunakan untuk menyelesaikan berbagai permasalahan distribusi. Menurut Karaboga (2005), contoh seleksi mencari makan yang menjadi dasar munculnya kecerdasan bersama dari kawanan lebah madu terdiri dari tiga komponen utama yaitu sumber makanan, lebah pekerja (*employed foragers/employed bee*) dan selain lebah pekerja (*unemployed foragers/unemployed bee*).

- ***Artificial Bee Colony* untuk MFVRP**

Berikut langkah-langkah *Artificial Bee Colony* yang digunakan untuk menyelesaikan permasalahan MFVRP menurut Szeto (2011) :

1. Proses Inisialisasi, dibangkitkan suatu himpunan yang terdiri dari semua *customer*. Jumlah himpunan yang dibentuk sesuai dengan kebutuhan yang kemudian menentukan calon-calon rute untuk himpunan *customer* sesuai kapasitas kendaraan.
2. Evaluasi nilai *fitness*.
3. Tahap Lebah Pekerja (*Employed Bee Phase*) dengan Neighborhood Operator dalam hal ini digunakan Metode *Swaps*. Dilanjutkan dengan menghitung nilai *fitness* untuk kemungkinan himpunan rute baru dan *Greedy Selection* yaitu seleksi himpunan rute.
4. Tahap Lebah Penjaga (*Onlooker Bee Phase*), yaitu menghitung probabilitas dan membangkitkan nilai random [0,1] serta perluasan kemungkinan himpunan rute baru dengan Metode *Swaps*.
5. Tahap Seleksi Himpunan - Himpunan Rute, dengan membandingkan nilai *fitness* himpunan rute lama dengan nilai *fitness* himpunan rute baru.
6. Tahap Lebah Pengintai (*Scout Bee Phase*)

Berikut *flowchart* program Algoritma *Artificial Bee Colony*



Gambar 1. Flowchart Program Algoritma *Artificial Bee Colony*

Setelah membuat rancangan *flowchart* program selanjutnya adalah membuat rancangan tampilan program. Implementasi Algoritma ABC untuk pembuatan program ABC menggunakan software Borland Delphi 7. Pada Borland Delphi 7, tampilan program disusun dalam area yang disebut form yang terdiri dari *OpenDialog*, *SaveDialog*, *Button*, *PageControl*, dan *Label*. Karena luas form yang terbatas dan menggunakan banyak komponen, maka digunakan komponen *PageControl*. Satu *PageControl* akan berisi banyak page (*TabSheet*) yang saling bertumpuk. Pada program ABC untuk MFVRP ini digunakan 4 *TabSheet*. Struktur yang banyak digunakan dalam pembuatan program ini adalah *array* dan *record*. *Array* adalah kumpulan nilai yang sejenis. Bila suatu variabel dideklarasikan sebagai *array* maka akan ada sejumlah elemen didalamnya dan masing-masing elemen dapat dikenali dari nomor indeksnya (Satyananda: 2012).

Untuk *TabSheet* pertama menggunakan komponen *Panel* dan *Image*. *TabSheet* kedua menggunakan komponen *StringGrid* sebagai input data jarak, permintaan barang, banyak kendaraan dan kapasitas kendaraan. *TabSheet* ketiga menggunakan komponen *Label*, *Edit* dan *Memo* untuk menampilkan rute-rute dan total jarak tempuh hasil perhitungan program ABC. *TabSheet* keempat menggunakan komponen *Panel* dan *Image* untuk menampilkan visualisasi hasil perhitungan program ABC.

- **Contoh Penerapan Algoritma *Artificial Bee Colony* untuk MFVRP**

Diberikan data banyak permintaan dari 8 *customer* (8 titik), data jarak antar depot dan antar *customer* serta jumlah kendaraan dan kapasitas tiap-tiap kendaraan sebagai berikut :

Tabel 1 Banyak Permintaan Barang Tiap Customer

<i>Customer (i)</i>	1	2	3	4	5	6	7
d_i	40	32	56	22	19	48	30

Tabel 2 Jarak antar Depot dan Jarak antar Customer dalam (km)

$i \backslash j$	0	1	2	3	4	5	6	7
0	0	24	41	32	11	10	6	21
1	24	0	14	24	17	30	28	27
2	41	14	0	12	26	34	38	25
3	32	24	12	0	24	16	15	9
4	11	17	26	24	0	5	4	7
5	10	30	34	16	5	0	3	8
6	6	28	38	15	4	3	0	13
7	21	27	25	9	7	8	13	0

Tabel 3 Kapasitas Kendaraan yang Dimiliki Perusahaan

Kendaraan(<i>k</i>)	1	2	3
Q_k	72	75	150

Berikut perhitungan dengan menggunakan Algoritma *Artificial Bee Colony*.

1. Proses Inisialisasi, dibangkitkan suatu himpunan yang terdiri dari semua *customer*.

x_p	Urutan <i>customer</i> tiap solusi						
x_1	2	5	1	4	7	3	6
x_2	5	7	2	4	6	1	3
x_3	6	1	3	5	7	2	4
x_4	1	5	2	6	3	7	4

Jumlah himpunan yang dibentuk sesuai dengan kebutuhan yang kemudian menentukan calon-calon rute untuk himpunan *customer* sesuai kapasitas kendaraan.

Dari *customer* urutan pertama pada x_1 yaitu *customer* 2 dengan jumlah permintaan barang 32, cek kendala kapasitas ($d_2 = 32 \leq Q_1$). Karena penambahan *customer* 2 dalam grup memenuhi kendala kapasitas maka *customer* 2 masuk dalam grup kendaraan 1. Grup kendaraan 1:

x_1	0	2
-------	---	---

Dari *customer* urutan pertama berlanjut ke *customer* kedua pada x_1 yaitu *customer* 5 dengan jumlah permintaan barang 19, cek kendala kapasitas ($d_2 + d_5 = 32 + 19 = 51 \leq Q_1$). Karena penambahan *customer* 5 dalam grup memenuhi kendala kapasitas maka *customer* 5 masuk dalam grup kendaraan 1. Grup kendaraan 1:

x_1	0	2	5
-------	---	---	---

Lakukan langkah yang sama untuk masing-masing kelompok *customer* $x_p, p = 1, 2, \dots, 4$. diperoleh

x_p	Calon Rute Tiap Kendaraan										
x_1	0	2	5	0	1	4	0	7	3	6	0
x_2	0	5	7	0	2	4	0	6	1	3	0
x_3	0	6	0	1	0	3	5	7	2	4	0
x_4	0	1	5	0	2	0	6	3	7	4	0

2. Evaluasi nilai *fitness*.

x_p	Rute V_1	Jarak V_1 ($\sum c_{ij}^1$)	Rute V_2	Jarak V_2 ($\sum c_{ij}^2$)	Rute V_3	Jarak V_3 ($\sum c_{ij}^3$)	Total Jarak $z(x_p)$
x_1	[0,2,5,0]	85	[0,1,4,0]	52	[0,7,3,6,0]	51	188
x_2	[0,5,7,0]	39	[0,2,4,0]	78	[0,6,1,3,0]	90	207
x_3	[0,6,0]	12	[0,1,0]	48	[0,3,5,7,2,4,0]	118	178
x_4	[0,1,5,0]	64	[0,2,0]	82	[0,6,3,7,4,0]	48	194

Untuk nilai *fitness* masing-masing himpunan rute (x_p) yaitu :

x_p	Calon Rute Tiap Kendaraan	$\frac{1}{z(x_p)}$	$f(x_p)$
x_1	0 2 5 0 1 4 0 7 3 6 0	$\frac{1}{188}$	0,005319
x_2	0 5 7 0 2 4 0 6 1 3 0	$\frac{1}{207}$	0,00483
x_3	0 6 0 1 0 3 5 7 2 4 0	$\frac{1}{178}$	0,005617
x_4	0 1 5 0 2 0 6 3 7 4 0	$\frac{1}{194}$	0,005154

3. Tahap Lebah Pekerja (*Employed Bee Phase*) dengan Neighborhood Operator dalam hal ini digunakan Metode *Swaps*.

x_1	0	2	5	0	1	4	0	7	3	6	0
-------	---	---	---	---	---	---	---	---	---	---	---

\tilde{x}_1	0	2	7	0	1	4	0	5	3	6	0
---------------	---	---	---	---	---	---	---	---	---	---	---

\tilde{x}_1 merupakan himpunan rute baru dimana selanjutnya akan dihitung nilai *fitness* \tilde{x}_1 atau ($f(\tilde{x}_1)$). Dilanjutkan dengan menghitung nilai *fitness* untuk kemungkinan himpunan rute baru dan *Greedy Selection* yaitu seleksi himpunan rute.

Karena total permintaan masing-masing rute x_1 dan \tilde{x}_1 memenuhi kendala kapasitas masing-masing kendaraan, maka dapat dilanjutkan dengan seleksi nilai *fitness*. $f(x_1) = 0,005319 < f(\tilde{x}_1) = 0,005376$ maka pilih $f(\tilde{x}_4) = 0,005376$ karena nilai *fitness* terbesar. Sehingga kedudukan x_p (himpunan rute lama) digantikan dengan \tilde{x}_p (himpunan rute baru *Swaps* ke I) dan limit $l_1 = 0$.

x_p	Permintaan (d_i)	$f(x_p)$	\tilde{x}_p	Permintaan (d_i)	$f(\tilde{x}_p)$
-------	----------------------	----------	---------------	----------------------	------------------

	$d_i(V_1)$	$d_i(V_2)$	$d_i(V_3)$			$d_i(V_1)$	$d_i(V_2)$	$d_i(V_3)$	
x_1	51*	62*	134*	0,005319	\tilde{x}_1	62*	62*	123*	0,005376
x_2	49*	54*	144*	0,00483	\tilde{x}_2	49*	88#	110*	0,005376
x_3	48*	40*	159#	0,005617	\tilde{x}_3	48*	56*	143*	0,005
x_4	59*	32*	156#	0,005154	\tilde{x}_4	51*	40*	156#	0,005524

4. Tahap Lebah Penjaga (*Onlooker Bee Phase*), yaitu menghitung probabilitas dan membangkitkan nilai random [0,1]

p	Himpunan Rute Tahap <i>Employed</i> (x_p)	$f(x_p)$	$\sum_{p=1}^4 f(x_p)$
1	0 2 7 0 1 4 0 5 3 6 0	0,005376	0,02073
2	0 5 7 0 2 4 0 6 1 3 0	0,00483	
3	0 6 0 3 0 1 5 7 2 4 0	0,005	
4	0 2 5 0 1 0 6 3 7 4 0	0,005524	

- Probabilitas x_1

$$P(x_1) = \frac{f(x_1)}{\sum_{p=1}^4 f(x_p)}$$

$$= \frac{0,005376}{0,02073}$$

$$= 0,2593$$

- Probabilitas x_2

$$P(x_2) = \frac{f(x_2)}{\sum_{p=1}^4 f(x_p)}$$

$$= \frac{0,00483}{0,02073}$$

$$= 0,2329$$

- Probabilitas x_3

$$P(x_3) = \frac{f(x_3)}{\sum_{p=1}^4 f(x_p)}$$

$$= \frac{0,005}{0,02073}$$

$$= 0,2411$$

- Probabilitas x_4

$$P(x_4) = \frac{f(x_4)}{\sum_{p=1}^4 f(x_p)}$$

$$= \frac{0,005524}{0,02073}$$

$$= 0,2664$$

x_p	$I(x_p) = \% P(x_p)$	Interval Garis Bilangan $I(x_p)$
x_1	$0,260 \times 100\% = 26\%$	0 – 0,2593
x_2	$0,231 \times 100\% = 23,1\%$	0,2594 – 0,4923
x_3	$0,241 \times 100\% = 24,1\%$	0,4924 – 0,7335
x_4	$0,265 \times 100\% = 26,5\%$	0,7336 – 1
Probabilitas max		1

x_p	Nilai Random	Interval	Kedudukan pada x_p
x_1	0,573	(0,493 – 0,734)	x_3
x_2	0,255	(0 – 0,260)	x_1
x_3	0,221	(0 – 0,260)	x_1
x_4	0,312	(0,261 – 0,492)	x_2

p	Himpunan Rute Baru Hasi Random (x_p)	Limit (l_p)
1	0 6 0 3 0 1 5 7 2 4 0	0
2	0 2 7 0 1 4 0 5 3 6 0	0
3	0 2 7 0 1 4 0 5 3 6 0	0
4	0 5 7 0 2 4 0 6 1 3 0	0

Perluasan kemungkinan himpunan rute baru dengan Metode *Swaps*.

Himpunan Rute Baru Hasil Swaps II

p	Himpunan Rute Baru (\tilde{x}_p)										
1	0	6	0	2	0	1	5	7	3	4	0
2	0	2	5	0	1	4	0	7	3	6	0
3	0	2	7	0	5	4	0	1	3	6	0
4	0	5	7	0	2	3	0	6	1	4	0

5. Tahap Seleksi Himpunan - Himpunan Rute, dengan membandingkan nilai *fitness* himpunan rute lama dengan nilai *fitness* himpunan rute baru.

Dipilih himpunan rute \tilde{x}_3 karena memiliki nilai *fitness* terbesar yaitu

$$f(\tilde{x}_3) = 0,005494, \text{ dan } \tilde{x}_3 \text{ sebagai } \hat{x}$$

$\tilde{x}_3 = x$	0	2	7	0	5	4	0	1	3	6	0
-------------------	---	---	---	---	---	---	---	---	---	---	---

p	Himpunan Rute Tahap Random (x_p)	$f(x_p)$	$f(\hat{x})$	Keterangan
1	0 6 0 3 0 1 5 7 2 4 0	0,0050	0,005494	$f(x_1) < f(\hat{x})$
2	0 2 7 0 1 4 0 5 3 6 0	0,005376	0,005494	$f(x_2) < f(\hat{x})$
3	0 2 7 0 1 4 0 5 3 6 0	0,005376	0,005494	$f(x_3) < f(\hat{x})$
4	0 5 7 0 2 4 0 6 1 3 0	0,00483	0,005494	$f(x_4) < f(\hat{x})$

Karena tidak ada nilai limit yang paling besar, maka pilih x_r yang memiliki nilai *fitness* paling kecil yaitu pilih $x_r = x_2$. Kedudukan x_2 yaitu himpunan rute hasil random [0,1], diganti dengan himpunan rute terpilih \hat{x} .

Sehingga diperoleh data akhir himpunan-himpunan rute tahap 5

p	Himpunan Rute Baru Hasi Random (x_p)	Limit (l_p)	Fitness ($f(x_p)$)
1	0 6 0 3 0 1 5 7 2 4 0	1	0,005
2	0 2 7 0 1 4 0 5 3 6 0	1	0,005376
3	0 2 7 0 1 4 0 5 3 6 0	1	0,005376
4	0 2 7 0 5 4 0 1 3 6 0	0	0,005494

6. Tahap Lebah Pengintai (*Scout Bee Phase*). Kemungkinan himpunan rute yang lain dengan Metode Swaps diperoleh

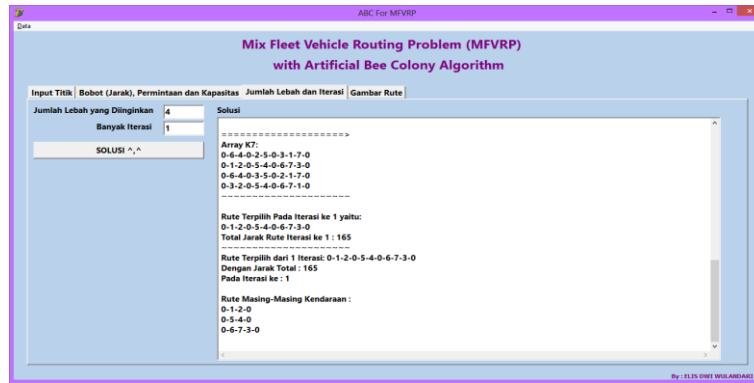
p	Himpunan Rute Lain Tahap Scout (\tilde{x}_p)
1	0 4 0 3 0 1 5 7 2 6 0
2	0 2 7 0 1 5 0 4 3 6 0
3	0 6 7 0 1 4 0 5 3 2 0
4	0 2 3 0 5 4 0 1 7 6 0

Diantara ke empat calon solusi pada tahap 5, perhatikan bahwa hanya terdapat tiga himpunan rute yaitu x_1, x_2 dan x_4 , karena $x_2 = x_3$. Pilih nilai limit terbesar yaitu nilai limit x_1 dan x_2 . Karena nilai limit x_1 sama dengan nilai limit x_2 sehingga dapat dipilih untuk x_i yang memiliki nilai *fitness* terbesar yaitu pilih x_2 dengan $f(x_2) = 0,005376$ sebagai himpunan rute terpilih pada iterasi pertama ($v = 1$). Jadi rute-rute terpilih pada iterasi pertama yaitu menunjukkan:

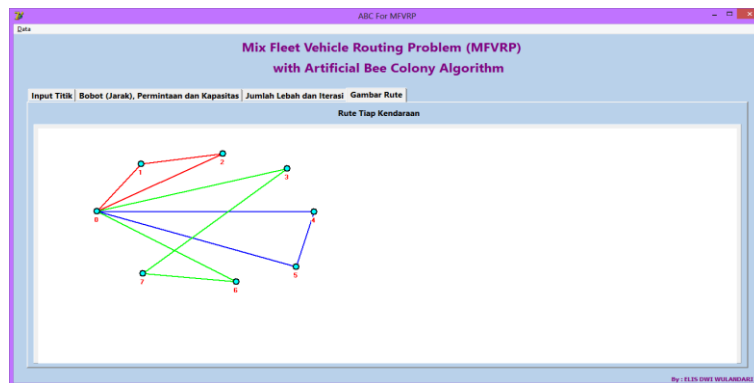
- Rute untuk kendaraan 1 : [0,2,7,0] dengan jarak tempuh 87 km
- Rute untuk kendaraan 2 : [0,1,4,0] dengan jarak tempuh 52 km
- Rute untuk kendaraan 3 : [0,5,3,6,0] dengan jarak tempuh 47 km

Dan total jarak tempuh keseluruhan adalah $87 + 52 + 47 = 186$ km. Proses kembali ke tahap lebah pekerja (*Employed Bee Phase*) dengan tetap melibatkan nilai limit terakhir pada tahap 5 dan terus berulang sampai kriteria pembatas terpenuhi.

Penyelesaian menggunakan program



Gambar 2. Hasil perhitungan



Gambar 3. Visualisasi Hasil

Pada uji coba 8 titik (4 lebah dan 1 iterasi) dengan program diperoleh rute kendaraan 1 yaitu 0-1-2-0, rute kendaraan 2 yaitu 0-5-4-0 dan untuk rute kendaraan 3 yaitu 0-6-7-3-0 dengan total jarak tempuh 165 km

PENUTUP

Kesimpulan

- Langkah-langkah Algoritma *Artificial Bee Colony* (ABC) mampu menyelesaikan permasalahan *Mix Fleet Vehicle Routing Problem* (MFVRP) untuk sejumlah kota yang harus dikunjungi. Penentuan rute pendistribusian barang pada MFVRP diperoleh dari perhitungan – perhitungan yang terdapat dalam langkah-langkah Algoritma *Artificial Bee Colony* diantaranya adalah pembentukan himpunan-himpunan rute awal dengan menentukan bilangan awal dan interval antar titik (*customer*) secara acak setiap himpunan. Pembentukan perluasan rute sebanyak lebah yang diinginkan. Perluasan kemungkinan himpunan rute dalam kasus ini melibatkan rute yang telah terbentuk, perhitungan *fitness* tiap kemungkinan himpunan rute, perhitungan *probability* tiap kemungkinan himpunan rute, dan pencarian nilai maksimum untuk nilai

limit atau nilai *fitness* terbesar dari himpunan rute. Dari langkah-langkah tersebut, terdapat iterasi untuk perbaikan sehingga kemungkinan himpunan rute yang berisi jarak tempuh total rute terpendek bisa ditemukan.

2. Implementasi langkah-langkah Algoritma *Artificial Bee Colony* dalam bahasa pemrograman untuk MFVRP telah berhasil, yaitu terbentuknya program ABC untuk MFVRP. Dimulai dengan input data *customer*, jarak antar *customer*, permintaan tiap *customer*, banyak kendaraan serta kapasitas masing-masing kendaraan yang kemudian data diproses dengan menggunakan Algoritma *Artificial Bee Colony* dalam bahasa pemrograman Borland Delphi 7 dan output yang dihasilkan berupa rute-rute yang optimum untuk masing-masing kendaraan. Rute terpilih bergantung pada banyak titik, banyak lebah dan banyak iterasi yang digunakan. Pada beberapa iterasi yang digunakan dengan program ABC dapat diperoleh rute minimal untuk setiap kendaraan dalam mendistribusikan produk tanpa melanggar kapasitas kendaraan. Pembuatan program tersebut sebagai alat bantu perhitungan sangat membantu bagi instansi-instansi untuk menentukan rute pendistribusian yang optimal dalam setiap permasalahan *Mix Fleet Vehicle Routing Problem* yang ditemui, serta mampu menyelesaikan masalah *Mix Fleet Vehicle Routing Problem* dengan jumlah *customer* yang besar.

Saran

Pada skripsi ini diberikan batasan bahwa graph yang digunakan adalah graph komplit berbobot, sehingga disarankan untuk mengkaji permasalahan *Mix Fleet Vehicle Routing Problem* dengan kapasitas kendaraan yang ada kurang dari permintaan customer, MFVRP yang dimodelkan dengan graph tidak komplit, serta menggunakan jarak yang *assymetric*. Karena adanya fakta dalam kehidupan nyata bahwa terdapat beberapa *customer* yang tidak terhubung langsung oleh jalan yang dilewati dari *customer* A ke B tidak sama dengan jalur yang dilewati dari *customer* B ke A.

DAFTAR RUJUKAN

- Brajevic, Ivona. 2011. Artificial Bee Colony Algorithm for The Capacitated Vehicle Routing Problem. *Proceedings of the European Computing Conference ISBN: 978-960-474-297-4*. (Online), (<http://www.wseas.us/e-library/conferences/2011/Paris/ECC/ECC-39.pdf>), diakses 20/01/2015 : 10:31 AM.
- Fachrurrozi, M. 2005. *Konsep dan Aplikasi Pemrograman Menggunakan Borland c++ Builder 6*. Palembang. Tidak diterbitkan.
- Iskandar. 2010. Model Optimasi Vehicle Routing Problem dan Implementasinya. Tesis. Institut Pertanian Bogor.
- Johnsonbaugh, R. 2001. *Discrete Mathematics. Fifth Editions*. New Jersey: Prentice-Hall, Inc.
- Karaboga, D. 2005. An Idea Based on Honey Bee Swarm for Numerical Optimization. *Erciyes University, Engineering Faculty Computer*

- Engineering Department Kayseri/Türkiye*. (Online), (http://mf.erciyes.edu.tr/abc/pub/tr06_2005.pdf) diakses pada 13 Januari 2015.
- Karaboga, D, Basturk, B. 2007. Artificial Bee Colony (Abc) Optimization Algorithm For Solving Constrained Optimization Problems. *In: Proceedings Of The 12th International Fuzzy Systems Association World Congress On Foundations Of Fuzzy Logic And Soft Computing*. Springer, Berlin, IFSA 2007, Pp 789–798.
- Karaboga, D, Ozturk, C. 2011. A Novel Clustering Approach: Artificial Bee Colony (ABC) algorithm. *Applied Soft Computing* 11(1):652–657. (Online), (http://web.info.uvt.ro/~dzaharie/cne2012/proiecte/tehnici/ABC/ABC_clustering.pdf), diakses 9 Januari 2015.
- Nirwan, Sembiring. 1991. *Ekonomi Koperasi*. Jakarta: Erlangga.
- Prive, Julie. 2005. Solving a Vehicle Routing Problem Arising in Soft Drink Distribution. (Online), (<http://www.fsa.ulaval.ca/personnel/renaudj/pdf/Recherche/Soft%20Drink%20-%20JORS%20May%202005.pdf>) diakses 18 Januari 2015: 03.37 PM
- Sarwadi & Krismi, A. 2004. Algoritma Genetika untuk Penyelesaian Masalah Vehicle Routing. *Jurnal matematika dan Komputer*, (Online), 7(2) : 1-10, (http://eprints.undip.ac.id/2226/1/1_Sarwadi_-_Anjar_Krismi.pdf), diakses 3 Januari 2013.
- Satyananda, Darmawan. 2012. *Panduan Praktikum Struktur Data*. Malang: Fakultas Matematika dan Ilmu Alam Universitas Negeri Malang.
- Staton. 1996. *Fundamental Of Marketing*. Sydney: McGraw-Hill.
- Suthikarnnarunai, N. 2008. A Sweep Algorithm for the Mix Fleet Vehicle Routing Problem. *Proceedings of the International Multi Conference of Engineers and Computer Scientists 2008 Vol II IMECS 2008, 19-21 March, 2008, Hong Kong*. (Online), (http://www.iaeng.org/publication/IMECS2008/IMECS2008_pp1914-1919.pdf), diakses pada 20 Januari 2015.
- Szeto, W. Y., Wu, Y., Ho, S. C. 2011. *An Artificial Bee Colony Algorithm for The Capacitated Vehicle Routing Problem*. *European Journal Of Operational Research*, 2011, v.215 n.1, p. 126-135. (Online), (<http://hub.hku.hk/bitstream/10722/135063/1/Content.pdf?accept=1>), diakses 10 Februari 2015.
- Yeun, Choong, L, and Zirour, M. 2008. *Vehicle Routing Problem : Models and Solution*, (Online), (http://www.ukm.my/ppsmfst/jqma/Vol4_Is1/abstract/JQMA-4-1-19-abstractrefs.pdf), diakses 21 Oktober 2014.
- Wilson, J.R. 2000. *Graphs and Applications*. Jerman: Springer.