

# Implementasi metode *simulated annealing* (SA) pada *vehicle routing problem with simultaneous deliveries and pick-ups* (VRPSDP)

Amelia Wahyu Noor Aida<sup>1</sup>, Mohamad Yasin<sup>2</sup>, Mimiep Setyowati Madja<sup>3</sup>

JURUSAN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS NEGERI MALANG

Email: [ameliawahyunoor@yahoo.com](mailto:ameliawahyunoor@yahoo.com)

**Abstrak:** *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups* (VRPSDP) adalah pengembangan dari VRP dengan penambahan kendala, di mana pada saat pengiriman barang ke konsumen disertai pula oleh pengambilan barang dari konsumen. Metode SA untuk menyelesaikan permasalahan VRPSDP diimplementasikan ke dalam program komputer yang dibuat dengan *Borland Delphi*. Dua dari lima uji coba dibandingkan dengan pengerjaan metode dan algoritma lain. Berdasarkan perbandingan tersebut, program SA mampu menyelesaikan permasalahan VRPSDP dengan hasil paling optimum. Hal ini disebabkan oleh proses random dalam pencarian rute dan iterasi yang diinputkan.

**Kata Kunci:** Metode *Simulated Annealing* (SA), *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups* (VRPSDP), *Vehicle Routing Problem* (VRP).

**Abstract:** *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups* (VRPSDP) is improvement of VRP with additional obstacles, in which at the time delivery goods to customers accompanied by the pick-up of customer goods. SA method to solve problems VRPSD implemented into a computer program made by *Borland Delphi*. The two of five tests compared to other implementation methods and algorithms. Based on this comparison, the SA program is able to solve VRPSDP problems with the most optimum results. It is caused by a random process in the search for routes and the iteration is entered.

**Keywords:** *Simulated Annealing* (SA) Method, *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups* (VRPSDP), *Vehicle Routing Problem* (VRP).

Masalah optimalisasi sering muncul dalam kehidupan sehari-hari, misalkan optimalisasi dalam pencarian rute terpendek. Matematika adalah salah satu cabang ilmu pengetahuan yang dapat digunakan untuk menyelesaikan masalah pencarian rute terpendek dengan memodelkan permasalahan menjadi suatu grafik. Konsep grafik pada matematika dipelajari dalam bahasan teori *graph*. Pada penerapannya, pencarian rute dengan jarak terpendek dapat diselesaikan dengan *Vehicle Routing Problem* (VRP). VRP adalah permasalahan optimalisasi dalam menentukan sejumlah rute untuk sekumpulan kendaraan yang harus melayani sejumlah konsumen dari depot pusat. Setiap rute berawal dan berakhir di depot. Dalam rute-rute yang dibentuk, setiap konsumen hanya dikunjungi satu kali oleh tepat satu kendaraan dengan total permintaan dari semua konsumen dalam satu kali rute tidak boleh melebihi kapasitas kendaraan. *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups* (VRPSDP) adalah pengembangan dari VRP dengan penambahan kendala, di mana pada saat pengiriman barang ke konsumen disertai pula oleh pengambilan barang dari konsumen.

Beberapa algoritma dan metode yang pernah dibahas sebelumnya yaitu *Metode Insertion Heuristic* (Diaz Vinancy Ayuandari, 2009), *Algoritma Scatter Search* (Mohammad Zakaria, 2014), dan *Algoritma Ant Colony System* (Cornelius Bahtiar Cristy, 2014). Berdasarkan contoh yang telah diselesaikan menggunakan algoritma dan metode yang pernah dibahas sebelumnya tersebut, akan dibandingkan hasilnya dalam penulisan ini.

- 
1. Amelia Wahyu Noor Aida adalah mahasiswa jurusan Matematika FMIPA Universitas Negeri Malang
  2. Mohamad Yasin adalah dosen jurusan Matematika FMIPA Universitas Negeri Malang
  3. Mimiep Setyowati Madja adalah dosen jurusan Matematika FMIPA Universitas Negeri Malang

Sehingga dapat disimpulkan metode yang paling optimum untuk menyelesaikan permasalahan VRPSDP.

*Travelling Salesman Problem (TSP)* adalah suatu permasalahan di mana seorang *salesman* mengunjungi seluruh kota dan kembali ke titik awal dengan aturan bahwa tidak boleh ada kota yang dikunjungi lebih dari satu kali, rute yang dipilih adalah rute dengan panjang minimum. Dari pengertian tersebut, permasalahan VRPSDP merupakan TSP yang berulang. Suatu metode yang dapat digunakan untuk menyelesaikan TSP, juga dapat digunakan untuk menyelesaikan permasalahan VRPSDP, salah satunya adalah *Simulated Annealing (SA)*. Metode SA adalah salah satu metode yang digunakan untuk mencari pendekatan terhadap solusi optimum global dari suatu permasalahan yang berbasis probabilitas. Pengerjaan metode SA yang diperoleh melalui proses random dan pengulangan, memungkinkan diperoleh solusi yang optimum. Permasalahan TSP yang diselesaikan menggunakan metode SA mempunyai syarat yaitu kota asal dan tujuannya sama. Dalam VRPSDP, kota awal dan tujuan menunjukkan depot.

Berdasarkan uraian di atas, akan dibahas mengenai metode *Simulated Annealing (SA)* pada permasalahan *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups (VRPSDP)*.

## HASIL YANG DIHARAPKAN

1. Mengetahui langkah-langkah metode SA dalam menyelesaikan permasalahan VRPSDP.
2. Mengimplementasikan metode SA ke dalam program komputer yang dibuat dengan *Borland Delphi 7.0*.
3. Menguji coba program yang dibuat dengan beberapa titik, membandingkan hasil pengerjaan program yang dibuat dengan algoritma dan metode lain.

## HASIL DAN PEMBAHASAN

### ➤ *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups (VRPSDP)*

*Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups (VRPSDP)* merupakan perluasan dari *Vehicle Routing Problem (VRP)*. VRPSDP adalah VRP dengan penambahan kendala, di mana pada saat pengiriman barang disertai pula oleh pengambilan barang secara simultan atau bersama-sama oleh kendaraan pengangkut yang nantinya akan dikembalikan lagi ke depot. Pada permasalahan VRPSDP, hanya ada satu depot dan kendaraan yang sama atau kapasitas sama yang jumlahnya lebih dari satu. Setiap konsumen hanya dikunjungi sekali oleh satu kendaraan. Contoh permasalahan VRPSDP adalah distribusi botol minuman ringan, gas LPG, dan galon minuman mineral. Tujuan dari VRPSDP yaitu untuk meminimumkan total jarak tempuh dari rute.

Formulasi matematika VRPSDP adalah sebagai berikut:

Meminimumkan total jarak.

$$\text{minimize } \sum_{k=1}^k \sum_{i=0}^n \sum_{j=0}^n c_{ij} x_{ijk}$$

Kendala: Setiap konsumen dikunjungi satu kali dan hanya oleh satu kendaraan.

$$\sum_{i=0}^n \sum_{k=1}^k x_{ijk} = 1, j = 1, 2, \dots, n;$$

Kendala: Menjamin bahwa kendaraan yang sama, datang dan berangkat dari konsumen yang telah dikunjungi.

$$\sum_{i=0}^n x_{ijk} - \sum_{i=0}^n x_{jik} = 0, j = 0, 1, \dots, n; k = 0, 1, \dots, k;$$

Batasan: Mendefinisikan bahwa paling banyak  $k$  kendaraan yang digunakan.

$$\sum_{j=0}^n x_{0jk} \leq 1, k = 0, 1, \dots, k;$$

Batasan: Persamaan untuk kapasitas permintaan pengambilan.

$$\sum_{i=0}^n y_{ji} - \sum_{i=0}^n y_{ij} = p_j, \forall j \neq 0;$$

Batasan: Persamaan untuk kapasitas permintaan pengiriman.

$$\sum_{i=0}^n z_{ij} - \sum_{i=0}^n z_{ji} = d_j, \forall j \neq 0;$$

Kendala: Menetapkan bahwa batas permintaan pengambilan dan pengiriman pada sisi yang ada pada solusi, paling banyak sama dengan kapasitas kendaraan.

$$y_{ij} + z_{ji} \leq Q \sum_{k=1}^k x_{ijk}, i, j = 0, 1, \dots, n;$$

Batasan: Kendala jarak maksimum.  $L$  adalah batas atas total beban yang diangkut oleh kendaraan dari setiap rute.

$$\sum_{i=0}^n \sum_{j=0}^n d_{ij} x_{ijk} \leq L, k = 0, 1, \dots, k;$$

Kendala: Sifat dari variabel keputusan.

$$x_{ijk} \in \{0, 1\}, y_{ij} \geq 0, z_{ij} \geq 0, i, j = 0, 1, \dots, n; k = 0, 1, \dots, k$$

**Keterangan:**

$k$  = kendaraan di depot

$V$  = himpunan konsumen yang harus dikunjungi

$n = |V|$  adalah jumlah konsumen

$d_j$  = permintaan pengiriman

$p_j$  = permintaan pengambilan

$c_{ij}$  = jarak antara  $i$  dan  $j$

$Q$  = kapasitas setiap kendaraan

$x_{ij} = 1$  jika sisi  $(i, j)$  mengikuti rute kendaraan  $k$ , 0 jika tidak

$y_{ij}$  = kapasitas pengambilan pada konsumen di titik  $i$  dan pada sisi  $(i, j)$

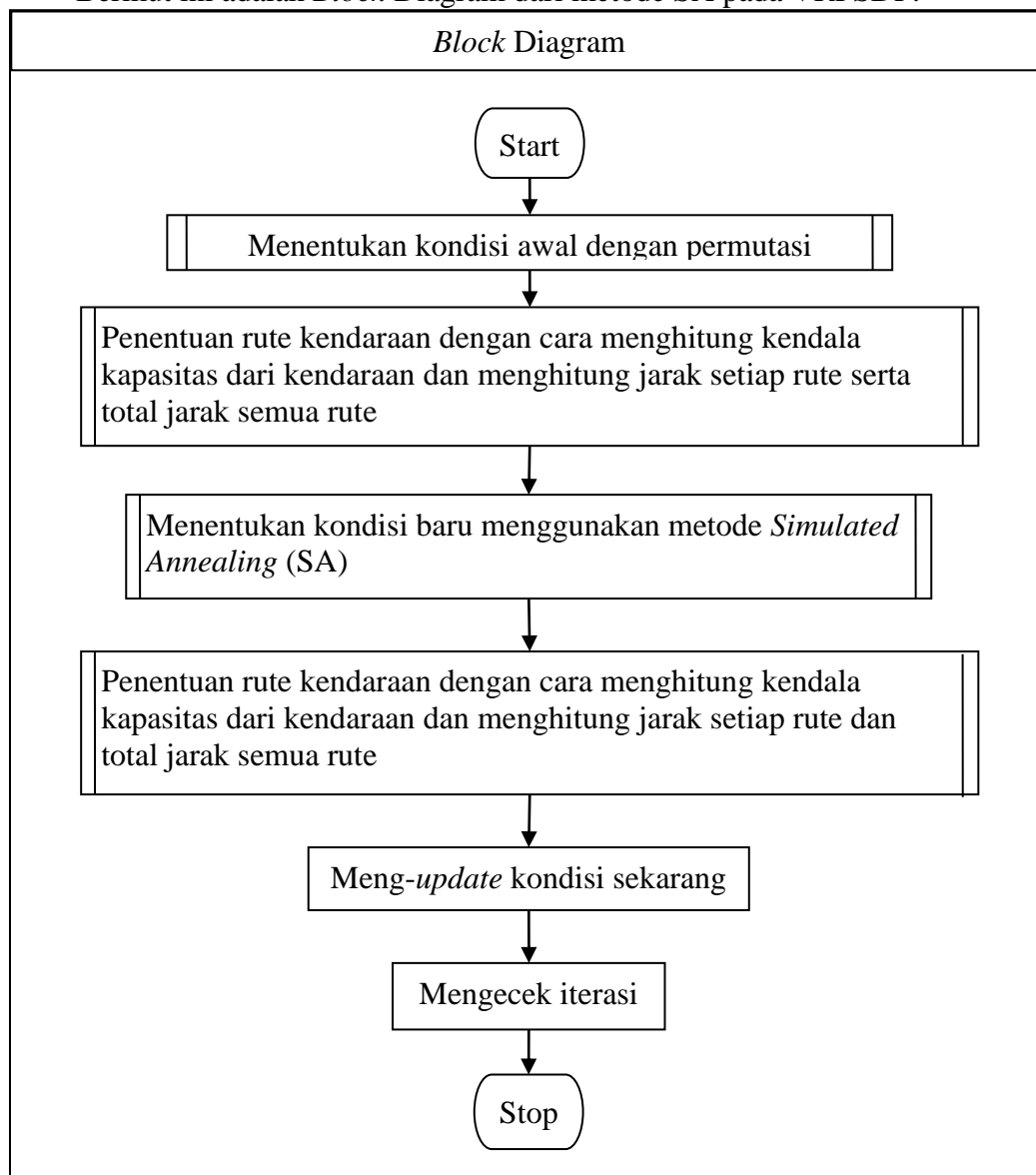
$z_{ij}$  = kapasitas pengiriman kepada konsumen setelah titik  $i$  dan pada sisi  $(i, j)$

➤ **Tahapan Metode Simulated Annealing (SA) pada Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups (VRPSDP)**

1. Tahap pertama adalah menentukan kondisi atau jalur awal menggunakan permutasi *Josephus*, dengan aturan sebagai berikut:

- Membangkitkan bilangan random  $r1$  dan  $r2$  dengan  $r1, r2 \leq$  jumlah konsumen.  $r1$  menyatakan konsumen pertama yang akan dituju,  $r2$  menyatakan selang. Bilangan-bilangan tersebut disimpan dalam larik  $L$ .
2. Tahap kedua adalah penentuan rute kendaraan dengan cara menghitung kendala kapasitas dari kendaraan. Lalu menghitung jarak setiap rute dan total jarak semua rute.
  3. Tahap ketiga adalah menentukan kondisi baru menggunakan metode *Simulated Annealing* (SA) dengan aturan sebagai berikut:  
Ada beberapa operator yang bisa digunakan. Berikut ini adalah salah satu contoh operator untuk menentukan jalur. Misalkan jumlah konsumen yang harus dikunjungi adalah  $NC$ .
    - ❖ Konsumen-konsumen disimpan pada larik  $L$ . Kondisi awal ini diperoleh dari permutasi *Josephus* (langkah 1).
    - ❖ Membangkitkan 2 bilangan random antara 1 sampai  $NC$ , misalkan kedua bilangan itu adalah  $N1$  dan  $N2$  dengan  $N1 < N2$ .
    - ❖ Depan =  $L(1)$  sampai  $L(N1 - 1)$ .
    - ❖ Tengah =  $L(N1)$  sampai  $L(N2)$ .
    - ❖ Belakang =  $L(N2 + 1)$  sampai  $L(NC)$ .
    - ❖ Membangkitkan bilangan random  $r$ , apabila  $r < 0,5$ ; maka:
      - DepanBaru = Depan.
      - TengahBaru = Tengah dengan urutan dibalik.
      - BelakangBaru = Belakang.
      - Lbaru = [DepanBaru TengahBaru BelakangBaru]
    - ❖ Jika  $r \geq 0,5$ ; maka kerjakan:
      - Sementara = [Depan Belakang], misalkan memiliki  $M$  elemen.
      - Bangkitkan bilangan random  $r$  dengan nilai antara 1 sampai  $M$ .
      - DepanBaru = Sementara( $1: r$ ).
      - TengahBaru = Tengah.
      - BelakangBaru = Sementara( $r + 1: M$ ).
      - Lbaru = [DepanBaru TengahBaru BelakangBaru]
  4. Dari kondisi baru yang diperoleh, melakukan tahap keempat dengan cara yang sama seperti pada tahap kedua.
  5. Tahap kelima adalah meng-*update* kondisi sekarang dengan cara membandingkan total jarak kondisi awal dengan kondisi baru, dengan catatan sebagai berikut:
    - ✓ Jika total jarak Lbaru  $\leq$  Lawal maka kondisi sekarang adalah Lbaru,
    - ✓ Jika total jarak Lbaru  $\geq$  Lawal maka kondisi sekarang adalah Lawal.

Berikut ini adalah *Block Diagram* dari metode SA pada VRPSDP.



**Gambar 1: *Block Diagram***

### ➤ Hasil Uji Coba

#### ▪ Uji Coba Manual

Contoh berikut ini merupakan permasalahan VRPSDP yang dirujuk dari skripsi tahun 2009 oleh Diaz Vinancya Ayuandari yang diselesaikan menggunakan metode *Insertion Heuristic*.

Suatu perusahaan air minum yang terletak di kota A, mempunyai 6 *customer* yang tersebar di beberapa wilayah dengan permintaan air minum kemasan galon dan juga pengembalian kemasan isi ulangnya pada bulan Juli ditunjukkan pada Tabel 1 berikut:

**Tabel 1: Jumlah Permintaan dan Pengembalian Tiap Customer Contoh 1**

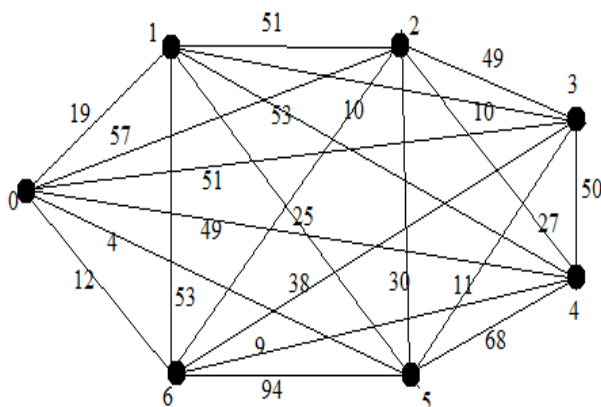
Customer	1	2	3	4	5	6
$d_i$	485	540	325	290	300	245
$p_i$	450	500	300	300	315	250

**Keterangan:**

Baris pertama menyatakan *customer*, baris kedua menyatakan permintaan tiap *customer* dan baris ketiga menyatakan pengembalian dari tiap *customer*. *Customer* 1 dengan jumlah permintaan sebesar 485 dan pengembalian sebesar 450, *customer* 2 dengan jumlah permintaan sebesar 540 dan pengembalian sebesar 500, dan seterusnya.

Perusahaan ini akan melakukan pengiriman serta pengambilan barang dengan menggunakan kendaraan yang sama yaitu truk pengangkut galon dengan kapasitas masing-masing kendaraan sebesar 800 galon.

Gambar 2 adalah graph yang menjelaskan kemungkinan jalur yang dapat dilewati di mana titik 0 mewakili depot dan titik 1-6 mewakili *customer*, setiap sisi mewakili jalan yang menghubungkan antar titik dan setiap bobot pada sisi mewakili jarak yang dinyatakan dalam satuan km.

**Gambar 2: Graph untuk Contoh 1**

Permasalahannya adalah bagaimana menemukan sejumlah rute *minimum* dalam melayani 6 *customer* di mana tiap *customer* hanya dilayani oleh satu kendaraan dengan tidak melanggar kendala kapasitas dan setiap rute berawal dan berakhir di depot.

**Penyelesaian:**

Berdasarkan gambar 2, dapat disusun tabel jarak antara *customer* dengan depot dan jarak antar *customer* pada Tabel 2 berikut.

**Tabel 2: Jarak antara Depot dengan Customer dan antar Customer Contoh 1**

	0	1	2	3	4	5	6
0	0	19	57	51	49	4	12
1	19	0	51	10	53	25	53
2	57	51	0	49	27	30	10
3	51	10	49	0	50	11	38
4	49	53	27	50	0	68	9
5	4	25	30	11	68	0	94
6	12	53	10	38	9	94	0

**Keterangan:**

untuk kolom pertama dan baris pertama, 0 menyatakan depot dan 1 hingga 6 merupakan *customer*. Jarak dari 0 (depot) ke *customer* 1 adalah 19 km, jarak dari *customer* 1 ke *customer* 2 adalah 51 km dan seterusnya.

Berdasar penyelesaian melalui tahapan pada metode SA, diperoleh hasil rute yaitu sebagai berikut:

1. Rute 1: 0 – 1 – 0
2. Rute 2: 0 – 3 – 6 – 0
3. Rute 3: 0 – 5 – 4 – 0
4. Rute 4: 0 – 2 – 0

Total jarak tempuh: 374 km

- **Uji Coba Program**

Metode SA diimplementasikan ke dalam program komputer. Untuk menjalankan program dari implementasi metode SA harus melalui beberapa tahap. Tahap pertama menginputkan data meliputi banyaknya titik dengan titik 0 sebagai depot dan titik lainnya sebagai konsumen, bobot atau jarak antar titik, permintaan pengiriman dan pengembalian tiap konsumen, kapasitas kendaraan, serta jumlah iterasi. Data tersebut akan diproses sesuai dengan tahapan modifikasi metode SA. Output dari proses tersebut yaitu menampilkan solusi rute dan total jarak tempuh berupa teks, serta solusi rute yang berupa visualisasi gambar.

Program ini telah diuji coba dengan 6, 7, 9, 20, 50, dan 100 titik. Berdasar uji coba 6 titik, dengan menginputkan data konsumen, kapasitas kendaraan sebanyak 75 dan iterasi sebanyak 20, menghasilkan 3 rute dengan total jarak tempuh 70 km. Berdasar uji coba 7 titik, dengan menginputkan data konsumen, kapasitas kendaraan sebanyak 800 dan iterasi sebanyak 20, menghasilkan 3 rute dengan total jarak tempuh 266 km. Berdasar uji coba 9 titik, dengan menginputkan data konsumen, kapasitas kendaraan sebanyak 75 dan iterasi sebanyak 50, menghasilkan 3 rute dengan total jarak tempuh 899 km. Berdasar uji coba 20 titik, dengan menginputkan data konsumen, kapasitas kendaraan sebanyak 300 dan iterasi sebanyak 200, menghasilkan 3 rute dengan total jarak tempuh 544 km. Berdasar uji coba 50 titik, dengan menginputkan data konsumen, kapasitas kendaraan sebanyak 400 dan iterasi sebanyak 20000, menghasilkan 5 rute dengan total jarak tempuh 953 km. Berdasar uji coba 100 titik, dengan menginputkan

data konsumen, kapasitas kendaraan sebanyak 100 dan iterasi sebanyak 200000, menghasilkan 6 rute dengan total jarak tempuh 228 km.

Dua dari enam uji coba program dibandingkan dengan metode dan algoritma lain. Berikut adalah perbandingan solusi penyelesaian VRPSDP antara metode SA dengan metode dan algoritma lain, seperti pada Tabel 3 dan 4 berikut:

**Tabel 3: Hasil Perbandingan Penyelesaian Uji Coba 7 Titik**

	Metode <i>Simulated Annealing</i>	Metode <i>Insertion Heuristic</i>
Jumlah Kendaraan	3	3
Rute	[0 – 5 – 3 – 0]	[0 – 2 – 6 – 0]
	[0 – 2 – 6 – 0]	[0 – 1 – 5 – 0]
	[0 – 1 – 4 – 0]	[0 – 3 – 4 – 0]
Total Jarak	266 km	277 km

Keterangan:

untuk baris pertama menyatakan metode yang digunakan untuk menyelesaikan permasalahan VRPSDP. Baris kedua menyatakan jumlah kendaraan yang digunakan, menggunakan metode SA dan metode *Insertion Heuristic* dengan jumlah kendaraan 3. Baris ke 3 sampai 6 menyatakan rute, misal pada baris 3 kolom 2 menyatakan rute pertama dengan penyelesaian metode SA diperoleh rute yang berawal di depot, menuju *customer* 5, menuju *customer* 3, dan kembali lagi ke depot, dan seterusnya. Baris ketujuh menyatakan total jarak, menggunakan metode SA dengan total jarak 266 km dan metode *Insertion Heuristic* dengan total jarak 277 km.

**Tabel 4: Hasil Perbandingan Penyelesaian Uji Coba 9 Titik**

	Metode <i>Simulated Annealing</i>	Algoritma <i>Ant Colony System</i>
Jumlah Kendaraan	3	3
Rute	[0 – 3 – 4 – 5 – 0]	[0 – 5 – 4 – 3 – 0]
	[0 – 6 – 7 – 8 – 0]	[0 – 6 – 7 – 8 – 0]
	[0 – 2 – 1 – 0]	[0 – 2 – 1 – 0]
Total Jarak	899 km	899 km
	Algoritma <i>Scatter Search</i>	Metode <i>Insertion Heuristic</i>
Jumlah Kendaraan	3	3
Rute	[0 – 1 – 2 – 3 – 0]	[0 – 8 – 5 – 4 – 0]
	[0 – 4 – 5 – 6 – 0]	[0 – 7 – 2 – 1 – 0]
	[0 – 8 – 7 – 0]	[0 – 3 – 6 – 0]
Total Jarak	1014 km	1184 km

Berdasarkan hasil perbandingan pada Tabel 3 dan 4, menunjukkan bahwa program SA mampu menghasilkan solusi rute dengan total jarak tempuh yang lebih optimum dibandingkan metode dan algoritma lain. Dengan demikian dapat disimpulkan bahwa program SA untuk menyelesaikan permasalahan VRPSDP berhasil dirancang. Namun, program ini tidak selalu menghasilkan solusi rute yang lebih optimum. Hal ini dikarenakan solusi rute yang dihasilkan oleh program SA diperoleh melalui proses random.

## KESIMPULAN

Berikut ini adalah kesimpulan berdasar hasil dan pembahasan:

1. Metode *Simulated Annealing* (SA) yang telah dimodifikasi untuk menyelesaikan permasalahan *Vehicle Routing Problem With Simultaneous Deliveries and Pick-ups* (VRPSDP) memiliki beberapa tahap. Tahap pertama adalah menentukan kondisi atau jalur awal menggunakan permutasi *Josephus*. Tahap kedua adalah penentuan rute kendaraan dengan cara menghitung kendala kapasitas dari kendaraan. Lalu menghitung jarak setiap rute dan total jarak semua rute. Tahap ketiga adalah menentukan kondisi baru menggunakan metode *Simulated Annealing* (SA). Dari kondisi baru yang diperoleh, melakukan tahap keempat dengan cara yang sama seperti pada tahap kedua. Tahap kelima adalah meng-*update* kondisi sekarang dengan cara membandingkan total jarak kondisi awal dengan kondisi baru. Jika total jarak kondisi baru kurang dari atau sama dengan kondisi awal, maka kondisi sekarang adalah kondisi baru dan jika total jarak kondisi baru lebih dari atau sama dengan kondisi awal, maka kondisi sekarang adalah kondisi awal. Selanjutnya mengulangi tahap ketiga sampai tahap kelima sebanyak iterasi yang diinginkan.
2. Untuk menjalankan program dari implementasi metode SA harus melalui beberapa tahap. Tahap pertama menginputkan data meliputi banyaknya titik dengan titik 0 sebagai depot dan titik lainnya sebagai konsumen, bobot atau jarak antar titik, permintaan pengiriman dan pengembalian tiap konsumen, kapasitas kendaraan, serta jumlah iterasi. Data tersebut akan diproses sesuai dengan tahapan modifikasi metode SA. Output dari proses tersebut yaitu menampilkan solusi rute dan total jarak tempuh berupa teks, serta solusi rute yang berupa visualisasi gambar.
3. Program yang telah dibuat diuji dengan beberapa titik yaitu 6, 7, 9, 20, 50, dan 100 titik. Dua dari lima uji coba program SA dibandingkan dengan metode dan algoritma lain. Berdasar uji coba 7 titik, solusi rute yang diperoleh dari program SA menghasilkan total jarak tempuh yang lebih optimum dibanding dengan solusi rute yang diperoleh dari metode *Insertion Heuristic*. Berdasar uji coba 9 titik, solusi rute yang diperoleh dari program SA dan algoritma *Ant Colony System* menghasilkan total jarak tempuh yang sama dan lebih optimum dibanding dengan solusi rute yang diperoleh dari algoritma *Scatter Search* dan metode *Insertion Heuristic*. Berdasar perbandingan tersebut, menunjukkan bahwa program SA yang telah dirancang mampu menghasilkan rute dengan total jarak tempuh paling optimum dibandingkan dengan rute yang dihasilkan dari pengerjaan manual metode dan algoritma lain.

## SARAN

Bagi pengkaji atau peneliti selanjutnya diharapkan dapat:

1. Menggunakan program SA sebagai alat bantu pencarian solusi rute pada permasalahan VRPSDP. Agar diperoleh rute dengan total jarak yang optimum, disarankan untuk menginputkan jumlah iterasi yang besar.
2. Mengembangkan metode SA pada VRPSDP dengan menambahkan kendala waktu, waktu pelayanan per item, kecepatan kendaraan, dan *Multiple Trip*.
3. Memodifikasi metode-metode lain yang memperoleh solusi rute dengan cara random untuk menyelesaikan permasalahan VRPSDP sehingga diharapkan dapat menemukan solusi rute yang lebih optimum.

**DAFTAR PUSTAKA**

- Ayuandari, Diaz Vinancya. 2009. *Vehicle Routing Problem With Simultaneous Deliveries and Pick-Ups (VRPSDP) dengan Metode Insertion Heuristic dan Penerapannya*. Skripsi tidak diterbitkan. Malang: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Malang.
- Cristy, Cornelius Bahtiar. 2014. *Algoritma Ant Colony System (ACS) pada Vehicle Routing Problem With Simultaneous Delivery and Pick-Up (VRPSDP)*. Skripsi tidak diterbitkan. Malang: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Malang.
- Cao, E. & Lai, M. *An Improved Genetic Algorithm for the Vehicle Routing Problem With Simultaneous Delivery and Pick-up Service*. Tidak ada tahun. (Online), (<http://www.scribd.com/doc/54102392/An-Improved-Genetic-Algorithm-for-the-Vehicle-Routing-Problem-With>), diakses 21 Maret 2013.
- Kusumadewi, Sri. 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.
- Toth, Paolo & Vigo, Daniele. 2014. *Vehicle Routing: Problems, Methods, and Applications, Second Edition*. USA: Society for Industrial and Applied Mathematics and the Mathematical Optimization Society.
- Weisstein, Eric. 2015. *Wolfram Web Resource*, (Online), (<http://mathworld.wolfram.com/JosephusProblem.html>), diakses 23 April 2015.
- Zakaria, Mohammad. 2014. *Algoritma Scatter Search pada Vehicle Routing Problem With Simultaneous Delivery and Pick-Up (VRPSDP)*. Skripsi tidak diterbitkan. Malang: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Malang.