



## Penyelesaian *Capacitated Vehicle Routing Problem* Menggunakan Gabungan Algoritma Genetika dan *Simulated Annealing*

Yusup Syarif Firmansyah\*, Khusnul Novianingsih, Husty Serviana Husain

Program Studi Matematika, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam,  
Universitas Pendidikan Indonesia, Indonesia

\*Correspondence: E-mail: [y.s.firman@upi.edu](mailto:y.s.firman@upi.edu)

### ABSTRAK

Capacitated Vehicle Routing Problem (CVRP) adalah permasalahan pendistribusian sejumlah barang oleh kendaraan yang tersedia dengan kapasitas tertentu dari suatu depot ke sejumlah pelanggan lalu kembali ke depot. Tujuan penyelesaian CVRP adalah untuk menentukan rute pendistribusian dengan total jarak terpendek. Pada penelitian ini, digunakan gabungan algoritma genetika dan simulated annealing (GASA) untuk menyelesaikan permasalahan CVRP. Algoritma GASA bekerja dengan cara melakukan tahapan-tahapan pada Algoritma GA yaitu merepresentasikan kromosom, menghitung nilai fitness, seleksi, crossover, dan mutasi, kemudian dilanjutkan ke tahapan-tahapan Algoritma SA yaitu memodifikasi solusi terbaik yang diperoleh dari Algoritma GA sebelumnya, membandingkan nilai fitness solusi hasil modifikasi dengan solusi terbaik pada Algoritma GA, setelah itu melakukan kembali tahapan-tahapan Algoritma GA sampai iterasi maksimum tercapai. Dengan demikian gabungan GA dan SA mempunyai peluang besar untuk memberikan solusi optimal global. Hasil implementasi model CVRP dan Algoritma GASA pada masalah pendistribusian es krim suatu perusahaan di Kota Bandung diperoleh kesimpulan bahwa Algoritma GASA dapat menyelesaikan masalah tersebut dengan baik.

© 2021 Kantor Jurnal dan Publikasi UPI

### ABSTRACT

The capacitated vehicle routing problem (CVRP) is a problem of distributing a number of goods using vehicles with limited carrying capacity that needed to pick up or deliver items at various locations, such as round trips from a store to customers. The goal of CVRP is to obtain a route with shortest travel distance. This research proposed the combination of genetics algorithm with simulated annealing (GASA) to solve CVRP. The first step in GASA is to represent customer as chromosomes, calculate fitness values, selection, crossover, and mutation. After that we continue to optimize the problem using SA algorithm by modifying the best solution produced by GA algorithm, comparing the fitness of modified best solution with the best solution of GA, and return to GA algorithm until maximum iteration achieved. Thus, GASA has greater chance to obtain global optimal solution. To simulate the algorithm, GASA was used for CVRP of an ice cream company in Bandung City and was able to solve it well.

© 2021 Kantor Jurnal dan Publikasi UPI

### INFORMASI ARTIKEL

#### Sejarah Artikel:

Diterima 22 Agustus 2021

Direvisi 10 September 2021

Disetujui 24 Oktober 2021

Tersedia online 1 November 2021

Dipublikasikan 1 Desember 2021

#### Kata Kunci:

Algoritma Genetika (GA),  
Capacitated Vehicle Routing  
Problem (CVRP),  
Optimisasi,  
Simulated Annealing (SA).

#### Keywords:

Capacitated Vehicle Routing  
Problem (CVRP),  
Genetic Algorithm (GA),  
Optimization,  
Simulated Annealing (SA).

## 1. PENDAHULUAN

*Vehicle Routing Problem* (VRP) adalah masalah penentuan rute perjalanan sejumlah kendaraan dari suatu lokasi yang disebut depot ke sejumlah lokasi pelanggan dan kembali lagi ke depot semula. Tujuan dari penyelesaian VRP adalah untuk menentukan rute kendaraan yang mendatangi lokasi dari setiap pelanggan sehingga jarak tempuh dengan rute yang dibentuk menjadi seminimal mungkin (Wirdianto, 2007).

Terdapat beberapa jenis permasalahan dari VRP ini, salah satunya adalah yang ditangani pada penelitian ini, yaitu *Capacitated Vehicle Routing Problem* (CVRP). CVRP adalah perluasan dari VRP dengan faktor tambahan yaitu setiap kendaraan punya kapasitas tersendiri untuk satu komoditas (Shahab & Irawan, 2016). CVRP memiliki tujuan untuk menentukan rute dengan total jarak terpendek dengan permintaan barang untuk setiap rute tidak boleh melebihi kapasitas kendaraan yang melewati rute tersebut. CVRP termasuk ke dalam masalah *NP-Hard Problem*, artinya usaha komputasi yang digunakan semakin meningkat seiring dengan meningkatnya ruang lingkup masalah. Akibatnya, jika solusi optimal dihitung dengan menggunakan metode eksak, maka diperlukan waktu komputasi yang lama. Oleh karena itu untuk menyelesaikan CVRP diperlukan metode metaheuristik untuk mengaproksimasi solusi optimum agar solusinya dapat diperoleh dengan cepat dan cukup baik (Permana, et. al., 2020).

Terdapat beberapa metode yang telah terbukti handal dapat menyelesaikan CVRP diantaranya adalah Algoritma Genetika dan *Simulated annealing*. Algoritma Genetika (GA) merupakan algoritma pencarian yang diadaptasi dari proses genetika dan evolusi pada makhluk hidup (Li & Wei, 2008). Kelemahan pada GA adalah terindikasi mengalami konvergensi prematur karena sangat cepat dalam mencapai fase konvergen sehingga solusi dari GA terjebak pada solusi dari optimum lokal (Josi, 2017; Rahmi et. al., 2017). *Simulated Annealing* (SA) merupakan algoritma pencarian yang diadaptasi dari bidang metalurgi saat pembentukan kristal (Samana, et. al., 2015). Kelemahan dari SA adalah hanya dapat menyimpan satu solusi terbaik dan mengabaikan solusi terdahulu yang masih memungkinkan menghasilkan nilai yang lebih baik (Anggasari, 2017).

Untuk mengatasi kelemahan algoritma GA dan SA, kedua algoritma tersebut digabungkan menjadi algoritma GASA. Algoritma GASA merupakan gabungan antara GA dan SA yang akan digunakan dalam penelitian ini, SA memiliki kelebihan yaitu solusinya mampu keluar dari keadaan optimum lokal (Permana, et. al., 2020). Akibatnya solusi yang dihasilkan diharapkan bersifat optimum global. Selain itu algoritma GA juga menutupi kelemahan dari SA pada GA mampu menyimpan solusi terbaik pada setiap generasi.

*Artikel* ini membahas cara kerja algoritma GASA dalam menyelesaikan CVRP, serta implementasi algoritma GASA pada kasus pendistribusian es krim di salah satu perusahaan di Kota Bandung. Selain itu, pada artikel ini membahas mengenai perbandingan jarak pada rute kasus pendistribusian es krim dengan menggunakan algoritma GA, SA, dan GASA.

## 2. METODE

Pada bagian ini membahas mengenai Model CVRP dan juga teknik penyelesaian CVRP dengan menggunakan Algoritma GASA.

## 2.1 Model CVRP

Berdasarkan Permana, *et. al.*, (2020), asumsi-asumsi yang harus dipenuhi pada pemodelan CVRP adalah sebagai berikut:

- 1) Setiap pelanggan hanya dikunjungi tepat satu kali oleh satu kendaraan.
- 2) Setiap kendaraan mempunyai batasan kapasitas yang sama.
- 3) Setiap pelanggan terhubung satu sama lain dan jarak antara pelanggan  $i$  ke  $j$  sama dengan jarak pelanggan  $j$  ke  $i$ .
- 4) Kendaraan yang tersedia cukup untuk mengirim semua permintaan pelanggan.

Untuk menurunkan model CVRP, terlebih dahulu didefinisikan himpunan-himpunan yang digunakan oleh model sebagai berikut:

$S$  : himpunan pelanggan

$D$  : himpunan depot

$K$  : himpunan kendaraan

$Q$  : kapasitas kendaraan

$d_i$  : permintaan pelanggan ke  $i$

$c_{ij}$  : jarak dari  $i$  ke  $j$ , dimana  $i, j \in S \cup D$

Adapun variabel keputusan pada model didefinisikan sebagai berikut:

$$x_{ij}^k = \begin{cases} 1, & \text{jika terdapat perjalanan dari } i \text{ ke } j \text{ dengan kendaraan } k \\ 0, & \text{yang lainnya} \end{cases}$$

$$y_i^k = \begin{cases} 1, & \text{jika kendaraan } k \text{ mengunjungi pelanggan } j \\ 0, & \text{yang lainnya} \end{cases}$$

Pada permasalahan CVRP akan dicari rute dengan total jarak terpendek untuk melakukan pendistribusian barang dari depot ke pelanggan-pelanggan lalu kembali ke depot dengan menggunakan sejumlah  $k$  kendaraan sehingga permintaan setiap pelanggan terpenuhi. Oleh karena itu fungsi tujuan dari CVRP dapat direpresentasikan sebagai berikut:

$$\min Z = \sum_{k \in K} \sum_{j \in S} \sum_{i \in S} c_{ij} x_{ij}^k$$

Adapun kendala-kendala dari model CVRP adalah sebagai berikut:

- 1) Setiap pelanggan hanya dikunjungi oleh 1 kendaraan, akibatnya kendala ini direpresentasikan oleh persamaan:

$$\sum_{k \in K} y_i^k = 1, \forall i \in S$$

- 2) Jumlah kendaraan yang keluar dan masuk ke depot ada sebanyak  $K$ . Akibatnya kendala ini direpresentasikan oleh persamaan:

$$\sum_{j \in S} x_{0j} = \sum_{j \in S} x_{j0} = K, 0 \in D$$

- 3) Setiap kendaraan mengunjungi pelanggan tepat satu kali. Setelah itu kendaraan meninggalkan pelanggan tersebut dan melanjutkan perjalanan menuju pelanggan

selanjutnya atau kembali menuju depot jika pelanggan tersebut adalah pelanggan terakhir dari sebuah rute. Ini berarti, kendaraan yang menuju pelanggan  $j$  akan meninggalkan pelanggan  $j$ . Akibatnya kendala ini direpresentasikan sebagai berikut:

$$\sum_{j \in S} x_{ij}^k = \sum_{j \in S} x_{jg}^k = y_i^k, \forall i, g \in S \cup D, \forall k \in K$$

- 4) Setiap kendaraan memiliki batasan kapasitas  $Q$ , sehingga jumlah permintaan pelanggan yang harus dipenuhi dalam satu rute tidak melebihi kapasitas maksimum kendaraan. Akibatnya kendala ini direpresentasikan sebagai berikut:

$$\sum_{i \in S} d_i y_i^k \leq Q, \forall k \in K$$

## 2.2 Teknik Penyelesaian

Pada penelitian ini, CVRP akan diselesaikan dengan Algoritma GASA, yaitu gabungan antara Algoritma Genetika (GA) dan *Simulated Annealing* (SA). Algoritma GASA dirancang untuk menutupi kekurangan algoritma genetika dan *simulated annealing* sehingga diharapkan solusi yang dihasilkan adalah optimum global (Permana, et. al., 2020). GASA menerapkan operator-operator evolusi seperti *crossover*, mutasi, dan seleksi individu, serta variabel Temperatur digunakan untuk menentukan jumlah iterasi pada algoritma ini. Model CVRP pada subbab sebelumnya akan diselesaikan dengan algoritma GASA (*Genetic Algorithm Simulated Annealing*).

Sebelum memulai algoritma GASA terlebih dahulu ditentukan nilai dari parameter-parameter yang diperlukan yaitu: temperatur ( $T$ ), penurunan temperatur setiap iterasi ( $A$ ), kriteria penghentian ( $\varepsilon$ ), banyak generasi, *population rate* ( $p$ ), *crossover rate* ( $cr$ ), dan *mutation rate* ( $mr$ ). Secara garis besar proses dalam mencari solusi yang optimum pada algoritma GASA dimulai dengan melakukan proses pencarian solusi oleh algoritma GA pada solusi awal sehingga diperoleh solusi terbaik dari GA. Setelah solusi terbaik GA didapatkan dilanjutkan ke proses pencarian oleh SA dengan memodifikasi solusi terbaik pada GA. Proses ini akan diulang sampai iterasi maksimum tercapai.

Tahapan Algoritma GASA yang digunakan pada penelitian ini adalah sebagai berikut (Tamiliarasi, 2010):

1. Inisialisasi nilai dari  $T$ ,  $A$ , dan  $\varepsilon$  untuk proses SA.
2. Inisialisasi nilai dari banyak generasi yang dipilih, jumlah kromosom dalam satu populasi, *crossover rate*, dan *mutation rate*.
3. Merepresentasikan kromosom dalam bentuk permutasi encoding. Kromosom dari permutasi encoding adalah kromosom yang dibentuk dari bilangan integer yang merepresentasikan suatu urutan dalam satu rute.
4. Inisialisasi populasi awal sebanyak  $N$  kromosom.
5. Hitung nilai fitness dari masing-masing kromosom. Nilai fitness adalah nilai yang menyatakan baik atau tidaknya suatu individu untuk menjadi solusi. Fungsi fitness yang digunakan adalah

$$f = \frac{1}{z}$$

dengan  $z$  adalah total jarak suatu rute.

6. Pilih sebanyak  $m$  kromosom dari  $N$  jumlah kromosom dimana  $m$  adalah bilangan positif yang lebih kecil dari  $N$ . Kromosom tersebut dipilih melalui seleksi dengan metode rank.
7. Lakukan Crossover dengan probabilitas sebesar crossover rate. Proses crossover dilakukan pada kromosom yang telah lolos dalam seleksi dengan metode order crossover.
8. Lakukan proses mutasi pada offspring (kromosom anak) hasil crossover dan  $m$  kromosom terbaik pada generasi sekarang dengan probabilitas  $mr$ . Proses mutasi yang digunakan adalah swapping mutation.
9. Hitung nilai fitness dari masing-masing kromosom baru.
10. Urutkan fitness dari kromosom baru dan kromosom pada populasi generasi sekarang, lalu pilih  $N$  kromosom terbaik pada generasi sekarang untuk menuju generasi selanjutnya.
11. Ulangi langkah 5 sampai 10 dilanjutkan hingga mencapai generasi maksimum.
12. Kemudian jalankan Simulated Annealing dengan cara sebagai berikut:
13. Jika  $T > \varepsilon$ , maka
14. Modifikasi kromosom terbaik pada algoritma GA dengan cara dengan cara memilih dua nilai integer secara acak, lalu posisi gen pada dua bilangan integer tersebut ditukar. Sehingga diperoleh kromosom baru disebut sebagai kromosomx.
15. Hitung fitness kromosom baru hasil modifikasi kromosom terbaik pada GA.
16. Bandingkan kromosom terbaik pada algoritma GA dengan Kromosomx.
  - a. Jika selisih *fitness* kromosom baru dikurangi kromosom terbaik dari GA lebih dari nol maka gantikan kromom terbaik GA dengan kromom baru sebagai solusi.
  - b. Jika selisih *fitness* kromom baru dengan kromosom kurang dari nol maka gantikan kromosom terbaik dari populasi dengan kromosom baru jika peluang  $\exp\left(-\frac{|\Delta E|}{T}\right) > \delta$ . Dimana  $\Delta E$  adalah selisih nilai *fitness* kromosom baru dikurangi kromosom terbaik,  $\delta$  adalah sebuah bilangan acak antara 0 sampai 1.
17. Kurangi nilai  $T$ .
18. Jika nilai  $T < \varepsilon$ , stop. Jika  $T \geq \varepsilon$  ulangi langkah 6 dengan populasi yang digunakan adalah populasi terakhir pada GA dengan memperhatikan hasil perbandingan dari kromosom baru hasil modifikasi pada proses SA dengan kromosom terbaik dari GA.

### 3. HASIL DAN PEMBAHASAN

Pada bagian ini membahas mengenai komputasi yang dilakukan untuk menentukan rute pendistribusian sebuah perusahaan es krim di Kota Bandung. Terdapat satu buah depot sebagai tempat awal dan berakhirnya kendaraan mendistribusikan es krim ke agen distributor. Kendaraan yang digunakan adalah mobil Hino Dutro Box yang dapat mengangkut es krim tidak lebih dari 300 paket es krim. Agen distributor berjumlah 50, dimana setiap agen tersebut memiliki permintaan es krim yang berbeda-beda seperti data pada Tabel 1.

**Tabel 1.** Data Koordinat Depot dan Pelanggan serta permintaan setiap Pelanggan

No	Tempat	X	Y	Permintaan (Paket)
0	Depot	-6.878	107.587	11
1	Alfamart Bojong Soang	-6.986	107.627	11
2	Alfamart Batununggal	-6.953	107.627	19
3	Alfamart Ciroyom	-6.913	107.634	13
4	Borma Sk-Dago	-6.877	107.618	19
5	Griya Pasteur	-6.895	107.588	18
6	Griya Setrasari	-6.881	107.583	14
7	Hero Setrasari	-6.881	107.581	14
8	Lion Superindo Dago	-6.901	107.612	45
9	Lion Superindo Kopo	-6.967	107.575	19
10	Yogya Sukajadi	-6.894	107.597	12
11	Yomart Garuda	-6.908	107.578	11
12	Giant Suci	-6.897	107.635	20
13	Alfamart Cigugur Girang II	-6.817	107.587	19
14	Alfamart Kolmas	-6.805	107.594	14
15	Alfamart Kolmas Cihideung	-6.804	107.588	19
16	Indomaret Kolonel Masturi	-6.860	107.544	28
17	Yogya Cimahi	-6.880	107.551	26
18	Indomaret Lengkong Besar	-6.926	107.614	22
19	Waserba Martanegara	-6.934	107.633	14
20	Yogya Sunda	-6.918	107.618	16
21	Lion Superindo Moch Ramdan	-6.937	107.610	13
22	Transmart TSM	-6.927	107.635	43
23	Alfamart Lengkong Besar 101 B	-6.926	107.613	18
24	Yomart Tamansari	-6.902	107.607	17
25	Alfamart N115 Gegerkalong Girang	-6.864	107.592	18
26	Alfamart Ciwaruga	-6.861	107.576	12
27	Yomart Ciumbuleuit	-6.878	107.603	41
28	Alfamart Gegerkalong	-6.862	107.587	15
29	Yomart Kopo Imanuel	-6.933	107.596	22
30	Indomaret Martadinata Bandung	-6.906	107.613	14
31	Yomart Minimarket	-6.947	107.592	19
32	Indomaret Leuwi Panjang	-6.994	107.595	27
33	Indomaret Gegerkalong Hilir 172	-6.867	107.580	21
34	Indomaret Point	-6.917	107.609	19
35	Yomart Pagarsih	-6.921	107.588	36
36	Indomaret Sekeloa	-6.888	107.618	22

No	Tempat	X	Y	Permintaan (Paket)
37	Giant Istana Plaza	-6.905	107.596	35
38	Hypermart BIP	-6.908	107.611	37
39	Griya Setiabudhi	-6.871	107.594	31
40	Borma Cipadung	-6.926	107.710	30
41	Carrefour Cipadung	-6.925	107.712	36
42	Hypermart Metro Indah Mall	-6.942	107.658	39
43	Super INDO Antapani	-6.913	107.660	32
44	Alfamidi Cibaduyut	-6.948	107.594	20
45	Hero Supermarket Paskal	-6.915	107.593	29
46	Borma Toserba Dakota	-6.890	107.571	27
47	Indomaret Rajawali	-6.910	107.570	25
48	Alfamart Naripan	-6.920	107.614	23
49	Toko Sidodadi Roti & Kue	-6.923	107.604	12
50	Borma Toserba Cijerah	-6.919	107.569	42

### 3.1 Tahapan Implementasi

Pada bagian ini ini membahas mengenai tahapan implementasi dari algoritma GASA untuk masalah pendistribusian sebuah perusahaan es krim di Kota Bandung. Sebelum memulai algoritma GASA terlebih dahulu ditentukan nilai dari parameter-parameter yang diperlukan yaitu: temperatur ( $T$ ), penurunan temperatur setiap iterasi ( $A$ ), kriteria penghentian ( $\epsilon$ ), banyak generasi ( $G$ ), *population rate* ( $p$ ), *crossover rate* ( $cr$ ), dan *mutation rate* ( $mr$ ). Nilai parameter-parameter dari algoritma GASA yang digunakan terdapat pada Tabel 2.

**Tabel 2.** Setting Parameter pada GASA

Parameter	Nilai
T	1000
A	0.8
$\epsilon$	0.001
G	200
P	100
Cr	0.6
Mr	0.4

### 3.2 Hasil Implementasi

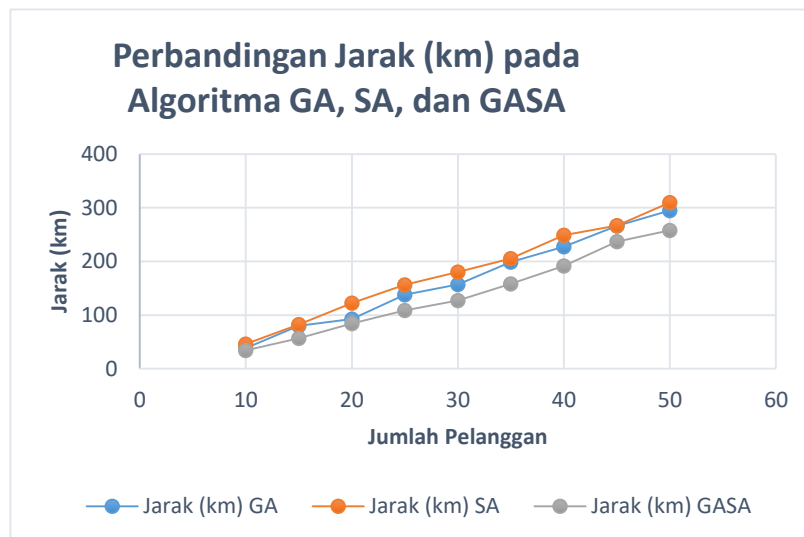
Dengan menggunakan *setting* parameter pada Tabel 2, setelah iterasi maksimum pada algoritma GASA tercapai, diperoleh sebuah kromosom dengan jarak terpendek. Kromosom tersebut menjadi solusi yang optimal untuk pendistribusian es krim perusahaan X di Kota Bandung. Hasil yang diperoleh untuk pendistribusian es krim perusahaan X di Kota Bandung

menggunakan algoritma GASA adalah sebuah rute dengan jarak minimum sebesar 257.716 km dengan rute sebagai berikut.

rute total: '0', '17', '13', '38', '25', '21', '47', '44', '9', '32', '50', '39', '23', '48', '12', '6', '30', '27', '1', '43', '33', '26', '16', '24', '15', '49', '18', '41', '7', '37', '42', '2', '3', '34', '19', '31', '45', '11', '28', '46', '20', '5', '14', '4', '35', '40', '22', '29', '8', '10', '36', '0'.

### 3.3 Perbandingan GA, SA, dan GASA

Pada bagian ini membahas mengenai perbandingan jarak dari rute yang dihasilkan oleh Algoritma Genetika (GA), *Simulated Annealing* (SA), dan gabungan Algoritma Genetika dan *Simulated Annealing* (GASA). *Setting* parameter yang digunakan dalam GA dituliskan pada Tabel 3.3 dengan G adalah jumlah generasi; p adalah jumlah kromosom dalam satu populasi; cr adalah *crossover rate*, dan mr adalah *mutation rate*. *Setting* parameter yang digunakan dalam algoritma SA juga dituliskan pada Tabel 3 dengan T adalah nilai temperatur, A adalah penurunan temperatur setiap iterasi, dan  $\epsilon$ , adalah kriteria penghentian. Setelah masing-masing algoritma dijalankan didapat hasil pada Gambar 1.



**Gambar 1** Perbandingan Jarak (km) pada Algoritma GA, SA, dan GASA

**Tabel 3.** Setting Parameter pada GA dan SA

Parameter GA	Nilai	Parameter SA	Nilai
G	200	T	1000
P	100	A	0.8
cr	0.621	$\epsilon$	0.001
Mr	0.38		

Berdasarkan Gambar 1 dapat diperoleh kesimpulan bahwa Algoritma GASA menghasilkan solusi rute yang lebih baik dibandingkan dengan rute yang dihasilkan oleh GA maupun SA berdasarkan jarak rute yang dihasilkan. Hasil ini selaras dengan hasil penelitian Li & Wei (2008) yang mengungkapkan bahwa Algoritma GASA memberikan solusi yang lebih baik dibandingkan dengan Algoritma GA dan Algoritma SA pada masalah optimasi pada reservoir



berganda. Algoritma GASA memberikan rute dengan jarak lebih kecil dibandingkan dengan GA maupun SA dikarenakan pada algoritma GASA mengalami total iterasi yang lebih banyak meskipun *setting* pada parameter GASA disamakan seperti pada GA dan SA. Sehingga rute yang diperoleh pada GASA lebih beragam dibandingkan GA maupun SA, akibatnya memungkinkan untuk mendapatkan rute dengan jarak paling minimum.

#### 4. KESIMPULAN

Berdasarkan model CVRP dan implementasi Algoritma GASA untuk kasus pendistribusian es krim salah satu perusahaan di Kota Bandung dapat diambil kesimpulan bahwa dalam menyelesaikan CVRP Algoritma GASA bekerja dengan cara melakukan tahapan-tahapan pada Algoritma GA yaitu merepresentasikan kromosom, menghitung nilai *fitness*, seleksi, *crossover*, dan mutasi, kemudian dilanjutkan ke tahapan-tahapan Algoritma SA yaitu memodifikasi solusi terbaik yang diperoleh dari Algoritma GA sebelumnya, membandingkan nilai *fitness* solusi hasil modifikasi dengan solusi terbaik pada Algoritma GA, setelah itu melakukan kembali tahapan-tahapan Algoritma GA sampai iterasi maksimum tercapai.

Masalah Pendistribusian salah satu perusahaan es krim di Kota Bandung dimodelkan dengan model CVRP dan Algoritma GASA berhasil diimplementasikan untuk menyelesaikan masalah tersebut. Berdasarkan hasil implementasi diperoleh kesimpulan bahwa rute dengan jarak minimum diperoleh dengan parameter *population rate* sebesar 100, banyak generasi sebesar 200, *crossover rate* sebesar 0.6, *mutation rate* sebesar 0.4, temperatur sebesar 1000, kriteria penghentian sebesar 0.001, dan penurunan temperatur 0.8 dimana diperoleh rute jarak 257.716 km.

#### 5. DAFTAR PUSTAKA

- Anggasari F, M. F. (2017). Optimasi kebutuhan gizi untuk balita menggunakan hybrid algoritma genetika dan simulated annealing. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1(12), 1668-1677.
- Josi, A. (2017). Implementasi algoritma genetika pada aplikasi penjadwalan perkuliahan berbasis web dengan mengadopsi model waterfall. *Jurnal Informatika: Jurnal Pengembangan IT*, 2(2), 77-83.
- Li, X. G., & Wei, X. (2008). An improved genetic algorithm-simulated annealing hybrid algorithm for the optimization of multiple reservoirs. *Water Resources Management*, 22, 1031-1049.
- Permana, E. R., Midyanti, D. M., & Hidayati, R. (2020). Optimasi pencarian rute terpendek distribusi barang menggunakan metode simulated annealing (Studi kasus: PD Bumi Jaya Indah Kota Pontianak). *Coding Jurnal Komputer dan Aplikasi*, 8(3), 9-18.
- Rahmi, A., Mahmudy, W. F., & Anam, S. (2017). Hibridisasi algoritma genetika dengan Variable Neighborhood Search (VNS) pada optimasi biaya distribusi. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 4(2), 87-96.
- Samana, E., Prihandono, B., & Noviani, E. (2015). Aplikasi simulated annealing untuk menyelesaikan travelling salesman problem. *Bimaster: Buletin Ilmiah Matematika, Statistika dan Terapannya*, 4(1), 25-32.

- Shahab, M. L., & Irawan, M. I. (2016). Algoritma genetika ganda untuk capacitated vehicle routing problem. *Jurnal Sains dan Seni ITS*, 4(2), A19-A24.
- Tamiliarasi, A., & Kumar, T. A. (2010). An enhanced genetic algorithm with simulated annealing for job-shop scheduling. *International Journal of Engineering, Science and Technology*, 2(1), 144-151.
- Wirdianto, E. et al. (2007). Penerapan algoritma simulated annealing pada penjadwalan distribusi produk. *Optimasi Sistem Industri*, 7(1), 7–20.