

# OPTIMASI DISTRIBUSI BARANG BERDASARKAN RUTE DAN DAYA TAMPUNG MENGGUNAKAN METODE SIMULATED ANNEALING

Susilo Dwi Juniarto, Entin Martiana K., S.Kom., M.Kom. , Arna Fariza, S.Kom., M.Kom. ,  
Ira Prasetyaningrum S.Si., M.T.

Jurusan Teknik Informatika, PENS - ITS Surabaya

Jl. Raya ITS, Surabaya

+62(31) 594 7280; Fax: +62(31) 594 6114

E-mail : [susilo@student.eepis-its.edu](mailto:susilo@student.eepis-its.edu), [entin@eepis-its.edu](mailto:entin@eepis-its.edu), [arna@eepis-its.edu](mailto:arna@eepis-its.edu), [ira@eepis-its.edu](mailto:ira@eepis-its.edu)

## ABSTRAK

*Vehicle Routing Problem* (VRP) adalah sebuah permasalahan dimana terdapat beberapa rute yang harus dilalui oleh sejumlah kendaraan yang berangkat dari suatu depot menuju beberapa tempat tujuan yang telah ditentukan dan berakhir pada depot yang sama. Permasalahan VRP dapat diselesaikan menggunakan algoritma *Simulated Annealing*. Algoritma *Simulated Annealing* merupakan suatu metode yang mengadopsi proses peleburan dan pendinginan besi secara perlahan-lahan dengan mengurangi *temperature*. Prinsip penggunaan algoritma *Simulated Annealing* adalah untuk mendapatkan hasil nilai *global optimum* pada penyelesaian suatu permasalahan yang ada.

PT Tirta Bahagia merupakan perusahaan penghasil produk Air Minum Dalam Kemasan (AMDK) merek Club yang memiliki permasalahan VRP. Permasalahan VRP yang dimaksud adalah permasalahan optimasi pendistribusian barang. Optimasi pendistribusian barang menyebabkan adanya pencarian rute terbaik dengan mempertimbangkan berat barang terhadap daya tampung kendaraan. Data-data pendistribusian barang yang ada pada PT Tirta Bahagia diolah menggunakan algoritma *Simulated Annealing* dengan menentukan teknik *cooling schedule* dan nilai jumlah iterasi yang tepat sehingga didapatkan hasil penyelesaian masalah yang sesuai dengan prinsip algoritma *Simulated Annealing*.

Hasil penyelesaian masalah pada permasalahan pendistribusian barang PT Tirta Bahagia yang sudah didapatkan adalah rute terpendek, informasi penggunaan kendaraan dan jumlah biaya yang dibutuhkan dalam pendistribusian barang. Rute terpendek divisualisasikan dengan menggunakan teknologi *Google Maps*.

**Kata kunci** : *Vehicle Routing Problem*, *Simulated Annealing*, *global optimum*, rute, berat barang, daya tampung kendaraan, optimasi, teknologi *Google Maps*.

## 1. PENDAHULUAN

Agar kehidupan usahanya dapat tetap berjalan, sebuah perusahaan selalu menjalankan proses produksi sehingga menghasilkan produk yang siap jual. Setelah menghasilkan produk yang siap jual, maka proses selanjutnya adalah melakukan penjualan produk. Dalam melaksanakan penjualan produk yang telah dihasilkan, suatu perusahaan tidak langsung menjual produk mereka kepada konsumen akhir. Produk-produk tersebut terlebih dahulu didistribusikan menuju distributor-distributor yang kemudian memasarkan produk-produk tersebut kepada konsumen akhir. Distributor tersebar di beberapa wilayah yang berbeda dalam suatu kota. Selanjutnya, proses pendistribusian barang dilakukan oleh pihak distributor menuju konsumen akhir yang telah melakukan pembelian produk-produk yang dibutuhkan.

Pada teknis pelaksanaan pendistribusian barang, pihak distributor menggunakan kendaraan-kendaraan pengangkut barang dengan ukuran kendaraan yang berbeda-beda. Dalam melakukan pendistribusian barang, sebuah kendaraan pendistribusi barang tidak hanya melayani satu konsumen akhir saja. Namun harus melayani

beberapa konsumen akhir sekaligus dalam melakukan sebuah perjalanan pendistribusian barang. Wilayah-wilayah konsumen akhir yang berbeda menyebabkan suatu kendaraan pendistribusi barang harus menentukan rute perjalanan yang akan dilaluinya sebelum melakukan perjalanan pendistribusian barang. Penentuan rute yang akan diambil harus sesuai dengan jarak terbaik antar distributor satu dengan distributor yang lainnya. Selain rute, daya tampung kendaraan terhadap berat barang yang akan dibawa juga harus diperhatikan. Dua hal tersebut menjadi sangat penting, karena dapat mempengaruhi kualitas proses pendistribusian barang. Semakin baik kualitas proses pendistribusian barang, semakin besar pula keuntungan yang akan didapatkan. Namun sebaliknya, kualitas proses pendistribusian barang yang buruk akan mengurangi keuntungan yang di dapat dari penjualan produk. Hal ini disebabkan karena kualitas proses pendistribusian barang yang buruk akan meningkatkan harga jual produk. Naiknya harga juala dapat menurunkan minat daya beli konsumen terhadap produk tersebut. Kemudian, dampak terburuk yang dapat ditimbulkan adalah menurunnya tingkat penjualan produk yang pada

akhirnya dapat mengancam kelangsungan hidup usaha dari sebuah perusahaan.

Pada proyek akhir ini dibangun sebuah aplikasi optimasi pendistribusian barang dengan mempertimbangkan rute perjalanan dan daya tampung kendaraan. Aplikasi ini bertujuan untuk mendapatkan hasil pendistribusian barang yang optimal. Permasalahan pencarian rute terbaik dengan mempertimbangkan daya tampung merupakan permasalahan Vehicle Routing Problem (VRP) yang diselesaikan dengan menggunakan metode Simulated Annealing. Sedangkan visualisasi hasil optimasi pendistribusian barang yang telah dihasilkan akan diimplementasikan dengan menggunakan Teknologi Google Maps.

Hasil yang dicapai dari pengerjaan proyek akhir ini memiliki tujuan sebagai berikut :

1. Mendapatkan nilai *global optimum* dari penggunaan algoritma *Simulated Annealing*.
2. Menentukan teknik *cooling schedule* dan nilai jumlah iterasi yang tepat dalam penggunaan algoritma *Simulated Annealing*.
3. Memproses data-data yang terkait dengan proses pendistribusian barang PT Tirta Bahagia menggunakan VRP dan metode *Simulated Annealing* dalam upaya pengoptimasian hasil proses pendistribusian barang PT Tirta Bahagia.
4. Melakukan visualisasi hasil optimasi yang didapatkan menggunakan teknologi *Google Maps*.

Pada penyelenggaraan proyek akhir ini, batasan permasalahan yang digunakan adalah :

1. Wilayah objek permasalahan adalah wilayah kota Surabaya.
2. Data diperoleh dari PT. Tirta Bahagia, dimana data tersebut meliputi data perusahaan berupa :
  - Letak Distributor barang di kota Surabaya.
  - Wilayah-wilayah konsumen di kota Surabaya.
  - Jenis barang dan berat barang yang dikirim oleh distributor.
  - Jenis Kendaraan dan daya tampung kendaraan yang digunakan.
  - Jadwal pendistribusian barang.
3. Parameter penentu optimasi adalah rute, beban rute dan daya tampung kendaraan.

Data mengenai peta dan jarak antar wilayah di kota Surabaya yang digunakan berasal dari Google Maps.

## 2. PENELITIAN YANG TERKAIT

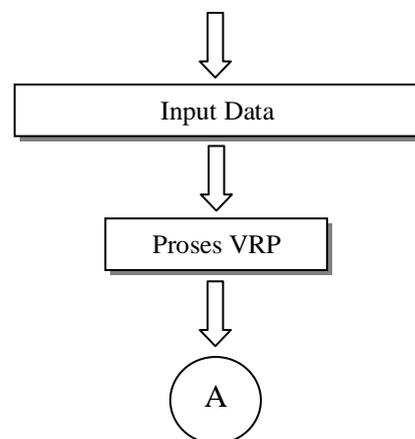
Stefanus Eko Wiratno [1] Permasalahan pembentukan rute dan jadwal kendaraan dapat diselesaikan dengan dua jenis algoritma, yaitu algoritma eksak dan algoritma heuristik. Algoritma heuristik memiliki keunggulan dalam hal waktu komputasi dibandingkan dengan algoritma eksak, akan tetapi dengan trade off solusi yang dihasilkan

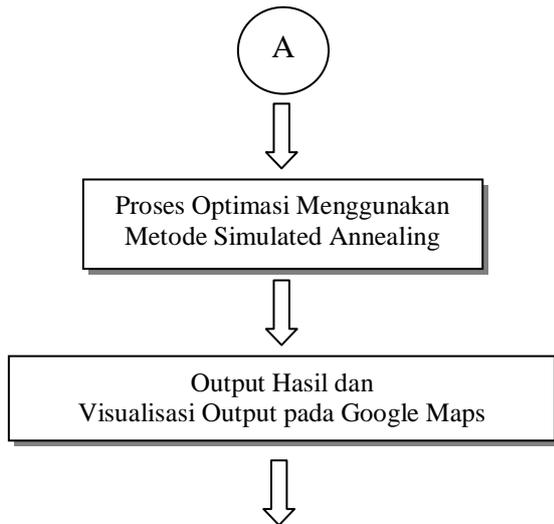
tidak sebaik yang dihasilkan oleh algoritma eksak. Simulated Annealing dikembangkan sebagai suatu pendekatan heuristik yang mampu melakukan perbaikan tingkat kualitas solusi dari suatu solusi awal yang diberikan. Selain mudah dalam implementasinya, kemampuannya untuk menghindari local optima yang buruk memberikan suatu harapan untuk memperoleh hasil yang lebih baik secara signifikan. Pembentukan rute dan jadwal kendaraan yang dibahas dalam penelitian ini diselesaikan melalui dua tahap dengan tetap memperhatikan batasan-batasan yang berlaku. Tahap pertama adalah tahap konstruksi tour dengan menggunakan

Algoritma Sweep untuk menghasilkan suatu initial solution yang digunakan sebagai input bagi Algoritma Simulated Annealing untuk diperbaiki. Algoritma Simulated Annealing melakukan perbaikan terhadap solusi awal tersebut dengan melakukan perpindahan node pelanggan antara dua rute melalui prosedur string relocation method. Hasil pembentukan rute dan jadwal kendaraan dengan menggunakan kedua algoritma tersebut menghasilkan suatu rute dan jadwal kendaraan yang tidak melanggar batasan sistem yang telah ditetapkan sebelumnya. Pembentukan dan jadwal awal dengan Algoritma Sweep menghasilkan rute dengan total jarak tempuh sebesar 249.149 km. Rute dan jadwal awal yang dihasilkan ini diperbaiki dengan Algoritma Simulated Annealing yang menghasilkan rute dengan total jarak tempuh sebesar 225.262 km yang berarti terjadi kenaikan sebesar 9.6%.

## 3. PERANCANGAN SISTEM

Aplikasi Optimasi distribusi barang ini dibuat untuk menyelesaikan permasalahan VRP dengan menggunakan metode Simulated Annealing. Dalam pelaksanaannya terjadi interaksi antara user dengan sistem maupun sistem dengan sistem sehingga aplikasi dapat berjalan sebagai mana mestinya. Gambaran rancangan global sistem secara umum adalah sebagai berikut :





**Gambar 1.** Gambaran Umum Sistem

Penjelasan proses pada perancangan global sistem :

1. Proses dimulai ketika admin melakukan input data mengenai jadwal pengiriman barang. Dimana di dalam jadwal pengiriman barang terdapat data-data terkait pengiriman barang seperti lokasi tujuan pengiriman barang, beban barang di setiap lokasi, jumlah kendaraan dan daya tampung kendaraan.
2. Kemudian admin menjalankan action yang menangani proses optimasi.
3. Kemudian data akan diproses menggunakan cara penyelesaian masalah dengan metode yang ada.
4. Setelah proses dijalankan, maka akan dihasilkan jarak rute terpendek beserta urutan rute terhadap kendaraan yang ada.
5. Output yang terakhir kali diberikan adalah berupa visualisasi rute menggunakan Teknologi Google Maps.

#### 4. ALGORITMA SIMULATED ANNEALING

Simulated Annealing adalah teknik pencarian random yang menggunakan analogi bagaimana pendinginan besi dan membekukannya ke dalam struktur energi kristalisasi minimum (proses annealing) dan mencari nilai minimum pada sistem secara keseluruhan, membentuk basis teknik optimasi untuk permasalahan kombinatorial dan permasalahan lainnya.

Simulated annealing dikembangkan oleh Kirkpatrick (1983) yang digunakan untuk optimasi kombinatorial yang merupakan varian dari algoritma Metropolis [4]. Simulated annealing digunakan untuk menyelesaikan persoalan local minimum.

Simulated annealing adalah teknik optimasi numerik dengan prinsip thermo-dynamic. Annealing adalah proses dimana material solid

dilebur dan didinginkan secara perlahan-lahan dengan mengurangi temperature. Partikel dari material berusaha menyusun dirinya sendiri selama proses pendinginan. Kumpulan status energi dari partikel yang dibentuk disebut “konfigurasi” dari material. Probabilitas partikel pada semua level energi dapat dihitung dengan menggunakan distribusi Boltzman. Dengan turunnya temperature, distribusi Boltzman mempertahankan konfigurasi partikel yang mempunyai energi terendah. Metropolis menemukan bahwa proses equilibrium dapat disimulasikan untuk temperature tetap dengan menggunakan Metode Carlo untuk membangkitkan deretan state energi. Sistem mengalami perubahan state dalam membentuk konfigurasi partikel yang baru. Level energi sebelum perubahan state ( $E_s$ ) dan level energi setelah perubahan state ( $E_i$ ) dibandingkan. Jika  $E_s$  lebih besar dari  $E_i$  (berarti  $\Delta E > 0$ ), sistem baru diterima sebagai konfigurasi partikel yang baru. Jika  $\Delta E > 0$ , probabilitas menerima sistem yang mengalami perubahan state menggunakan kriteria Metropolis yang disebut faktor probabilitas Boltzman

$$p' = e^{(\Delta E/KT)}$$

dimana  $k$  = konstanta Boltzman

$T$  = temperature

$\Delta E$  = perbedaan level energi sebelum dan sesudah perubahan energi

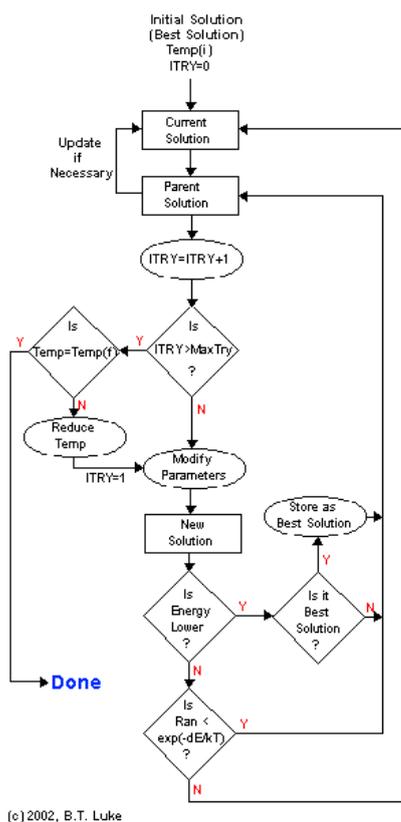
Angka random,  $p$ , dituliskan sebagai distribusi random uniform pada interval  $[0,1]$ . Jika  $p > p'$ , maka perubahan state ditolak dan step baru digunakan untuk posisi saat ini. Jika  $p' > p$ , step perubahan state diterima dan konfigurasi baru menggantikan yang lama. Step baru ditempatkan relatif untuk konfigurasi saat ini. Kriteria memungkinkan variable baru diterima sebagai konfigurasi baru meskipun mempunyai nilai fungsi respon yang jelek dibandingkan konfigurasi saat ini. Memindahkan yang sangat jelek masing kemungkinan diterima daripada memindahkan yang tidak jelek. Gambaran ini merupakan algoritma untuk meninggalkan local optima. Step baru diambil sampai kriteria terminasi tercapai.

Alur algoritma Simulated Annealing dimulai dengan solusi inisial suatu permasalahan yang juga dianggap sebagai solusi terbaik sementara, dan temperature diatur pada nilai inisial temperature tinggi  $Temp(i)$ . Solusi ini menjadi solusi saat ini (current) dan parent atau solusi aktif. Jumlah Monte Carlo (ITRY) diset 0. ITRY dinaikkan dan dites apakah mencapai nilai maksimum saat itu. Jika iya, maka temperature saat itu dilihat.

Apabila sama dengan temperature aktif  $Temp(f)$ , simulasi berakhir dan didapatkan solusi akhir dan solusi terbaik selama simulasi. Jika temperature saat ini diatas temperature akhir, maka temperature diturunkan menggunakan penjadwalan

pendinginan Jumlah Monte Carlo (ITRY) di-reset menjadi 1.

Jika jumlah temperature ini tidak tercapai, atau temperature sudah diturunkan solusi parent dimodifikasi untuk membangkitkan solusi baru. Jika energi pada solusi baru lebih kecil dibandingkan dengan parent, maka di-cek untuk melihat apakah solusi terbaik ditemukan. Jika ditemukan, dikirm secara terpisah. Apakah solusi terbaik atau tidak, solusi baru menjadi solusi parent baru untuk tahap Monte Carlo berikutnya. Meskipun solusi parent diubah, akan menjadi solusi current.



Gambar 2. Flowchart Algoritma Simulated Annealing

Jika energi dari solusi baru lebih tinggi dari parent dengan menghitung dE, probabilitas Boltzman dihitung,. Jika probabilitas lebih besar dari angka random (Ran) antara 0.0. dan 1.0, solusi baru diterima dan menjadi solusi parent untuk iterasi berikutnya, dan solusi current. Sebaliknya jika, probabilitas lebih kecil dari Ran, solusi baru ditolak dan solusi current atau parent tetap sama dan digunakan untuk iterasi berikutnya.

## 5. PENGGUNAAN TEKNOLOGI GOOGLE MAPS

Google Maps adalah layanan gratis Google yang cukup populer. Kita dapat menambahkan fitur Google Maps dalam web Kita

sendiri dengan Google Maps API. Google Maps API adalah library JavaScript. Yang dibutuhkan dalam penggunaan Google Maps API adalah pengetahuan tentang HTML dan JavaScript, serta koneksi Internet. Dengan menggunakan Google Maps API Kita dapat menghemat waktu dan biaya Kita untuk membangun aplikasi peta digital yang handal, sehingga Kita dapat focus hanya pada data-data. Biarkan data peta dunia menjadi urusan Google saja.

Saat ini versi terakhir Google Map API adalah versi 3. Versi ini, dikabarkan akan tampil lebih cepat dari versi sebelumnya khususnya untuk browser ponsel. Kita bisa membangun situs web yang dilengkapi peta untuk iPhone dan ponsel dengan sistem operasi Android.

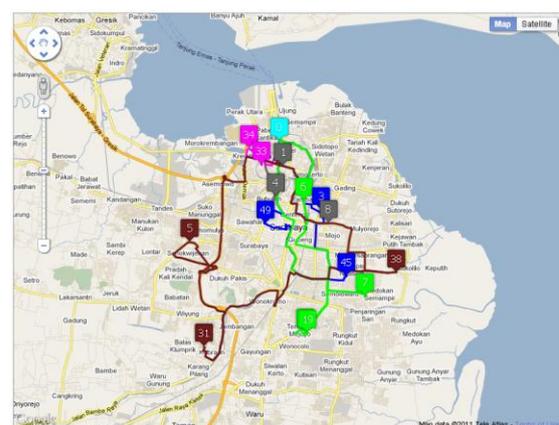
### 5.1 Menggunakan Google Maps API

Kita bisa mulai menulis program Google Map API dengan urutan sebagai berikut:

1. Memasukkan Maps API JavaScript ke dalam HTML kita.
2. Membuat element div dengan nama map\_canvas untuk menampilkan peta.
3. Membuat beberapa objek literal untuk menyimpan properti-properti pada peta.
4. Menuliskan fungsi JavaScript untuk membuat objek peta.
5. Meng-inisiasi peta dalam tag body HTML dengan event onload.

### 5.2 Contoh Penggunaan Google Maps API

Disini diberikan sebuah contoh penggunaan Google Maps API untuk menampilkan 10 penanda lokasi secara random seperti screenshot dibawah ini:



Gambar 3. Contoh Google Maps API

## 6. HASIL UJI COBA

Untuk menguji sistem yang telah dibuat, dilakukan 12 kali uji coba data menggunakan 2

parameter yang memiliki nilai berbeda-beda. Parameter tersebut terdiri dari 4 nilai jumlah iterasi (N) dan 3 teknik *cooling schedule*. Nilai jumlah iterasi yang digunakan adalah 100, 300 500 dan 700 iterasi. Kemudian terdapat 3 teknik *cooling schedule* berbeda yang digunakan.

Pada uji coba proses optimasi ini digunakan 15 data dimana data tersebut merupakan 15 data tujuan pengiriman barang yang dilakukan oleh pihak distributor PT Tirta Bahagia menuju agen yang menjadi konsumen AMDK (Air Minum Dalam Kemasan) merek Club. Data yang digunakan adalah sebagai berikut :

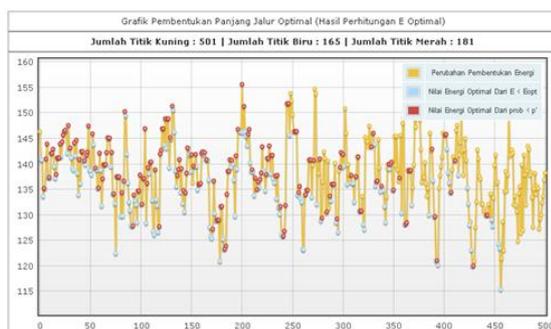
**Tabel 1.** Tabel Data Uji Coba

ID Pengiriman	ID Agen Tujuan	Total Beban (kg)	Total Volume (m3)	Tanggal Pengiriman
KK00001	AG00001	166.0	0.25102375	12-July-2011
KK00002	AG00003	214.0	0.361518	12-July-2011
KK00003	AG00004	120.5	0.20730775	12-July-2011
KK00004	AG00005	108.4	0.16490239999999997	12-July-2011
KK00005	AG00006	137.2	0.2173824	12-July-2011
KK00006	AG00007	137.0	0.1984612	12-July-2011
KK00007	AG00019	255.5	0.3969624000000001	12-July-2011
KK00008	AG00033	359.9	0.5573464	12-July-2011
KK00009	AG00008	142.3	0.2473646	12-July-2011
KK00010	AG00031	61.60000000000001	0.10002125	12-July-2011
KK00011	AG00038	91.8	0.1630368	12-July-2011
KK00012	AG00045	205.8	0.3260736	12-July-2011
KK00013	AG00049	122.4	0.2173824	12-July-2011
KK00014	AG00044	30.6	0.0543456	12-July-2011
KK00015	AG00034	122.0	0.17958	12-July-2011

Berikut ini merupakan uji coba data terhadap system yang dikembangkan :

**Tabel 2.** Tabel Hasil Uji Coba

Percobaan ke-	1	2	3	4	5
Waktu Proses(s)	872.97	889.271	849.23	879.965	702.055
Jarak Awal	137.39	146.329	136.28	147.102	137.056
Jarak Optimal	129.51	115.35	139.26	134.046	123.276
Jumlah E < Eopt	161	165	167	175	160
Jumlah prob < p'	198	181	199	195	191



**Gambar 4.**

Grafik pembentukan panjang jalur optimal

Setelah melakukan serangkaian uji coba data terhadap sistem, terdapat beberapa hal yang dapat di jadikan kesimpulan, diantaranya adalah :

1. Teknik penjadwalan *temperature (cooling schedule)* mempengaruhi hasil pembentukan solusi-solusi yang ada. Karena teknik ini

merupakan penentu pembentukan nilai T pada setiap tahap proses algoritma *Simulated Annealing*. Sedangkan nilai T sendiri mempengaruhi perhitungan nilai konstanta *bolzman* yang memiliki rumus  $p' = e^{-AE/T}$ . Perubahan nilai T yang lebih stabil menghasilkan kecenderungan nilai p' yang lebih stabil pula.

2. Pada uji coba data terhadap sistem terbukti bahwa nilai teknik *cooling schedule* pertama dengan rumus  $T_i = T_0 (T_N / T_0)^{i/N}$  dan jumlah iterasi 500 menghasilkan nilai optimasi yang lebih baik dibandingkan dengan teknik *cooling schedule* kedua dan ketiga serta jumlah iterasi lainnya (dapat dilihat di bagian uji coba). Nilai jumlah iterasi kurang dari 500 menghasilkan nilai pencapaian optimasi yang kurang maksimal, namun nilai jumlah iterasi lebih dari 500 memungkinkan nilai hasil optimasi yang tidak optimal dikarenakan terlalu banyak kemungkinan yang muncul. Terlalu banyak kemungkinan yang muncul dapat menghilangkan kemungkinan terbaik yang telah terpilih. Sehingga dapat disimpulkan bahwa penggunaan teknik *cooling schedule* pertama dan jumlah iterasi 500 adalah teknik *cooling schedule* dan jumlah iterasi yang paling baik.
3. Pembentukan rute awal bergantung pada nilai *random* yang dihasilkan sehingga nilai *temperature* maupun jumlah iterasi tidak mempengaruhi hasil pembentukan rute awal yang dihasilkan. Begitu juga dengan jumlah kendaraan yang digunakan pada satu rute yang dihasilkan, baik jumlah iterasi maupun *temperature* akhir juga tidak mempengaruhi nilai dari jumlah kendaraan yang dihasilkan tersebut.
4. Pada uji coba data tidak ditemukan kondisi *local optimum*, hal ini menunjukkan bahwa algoritma pada proses sistem yang telah dibangun sesuai dengan prinsip dari algoritma *Simulated Annealing* yaitu untuk menghasilkan nilai *global optimum*.
5. Setelah dilakukan validasi terhadap aplikasi yang telah dibangun kepada PT Tirta Bahagia, didapatkan hasil validasi yang menyebutkan bahwa aplikasi yang dibangun :

“Belum sesuai karena distribusi kami masih berdasarkan kapasitas muatan / volume dan rute berdasarkan kondisi rute kendaraan di lapangan mana yang lebih dahulu atau urgent”.

Hal tersebut disebabkan karena pada implementasi pendistribusian barang, PT Tirta Bahagia tidak memperhatikan daya tampung kendaraan terhadap berat barang yang sesuai dengan spesifikasi daya tampung kendaraan yang digunakan. Namun, pada kenyataannya

aplikasi yang telah dibuat pada proyek akhir ini mampu mengatasi permasalahan tersebut. Dikarenakan aplikasi ini dibangun dengan sistem yang memungkinkan data-data yang digunakan dapat bersifat dinamis dan dapat diubah-ubah sesuai kebutuhan. Sehingga aplikasi tetap dapat digunakan walaupun data-data yang digunakan tidak sesuai dengan data-data dalam keadaan normal.

Dari percobaan diatas, diketahui bahwa hasil terbaik terjadi pada percobaan kedua, dimana didapatkan jarak sebesar 115.35 km. Sedangkan untuk contoh pembagian armada rutanya, bisa dilihat pada gambar berikut:

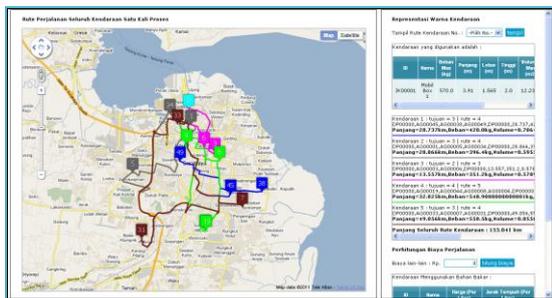
ID	Nama	Beban Max (kg)	Panjang (m)	Lebar (m)	Tinggi (m)	Volume (m <sup>3</sup> )
JK00001	Mobil Box 1	570.0	3.91	1.565	2.0	12.23

Kendaraan yang digunakan adalah :						
Kendaraan 1 : tujuan = 3   rute = 4 DP00000,AG00045,AG00038,AG00049,DP00000,28.737,44 <b>Panjang=28.737km,Beban=420.0kg,Volume=0.706</b>						
Kendaraan 2 : tujuan = 3   rute = 4 DP00000,AG00001,AG00005,AG00034,DP00000,28.866,39 <b>Panjang=28.866km,Beban=396.4kg,Volume=0.595</b>						
Kendaraan 3 : tujuan = 2   rute = 3 DP00000,AG00003,AG00006,DP00000,13.557,351.2,0.578 <b>Panjang=13.557km,Beban=351.2kg,Volume=0.578</b>						
Kendaraan 4 : tujuan = 4   rute = 5 DP00000,AG00019,AG00044,AG00008,AG00004,DP00000 <b>Panjang=32.825km,Beban=548.9000000000001kg,</b>						
Kendaraan 5 : tujuan = 3   rute = 4 DP00000,AG00033,AG00007,AG00031,DP00000,49.056,55 <b>Panjang=49.056km,Beban=558.5kg,Volume=0.855</b>						
<b>Panjang Seluruh Rute Kendaraan : 153.041 km</b>						

Gambar 5. Rute Hasil Optimasi

Setelah itu, kita tampilkan hasil terbaik tersebut kedalam Google Maps seperti pada gambar di bawah ini :



Gambar 6. Visualisasi Rute Hasil Optimasi Menggunakan Google Maps.

Hasil output aplikasi yang dikerjakan pada proyek akhir ini juga menghasilkan informasi biaya perkiraan dalam melakukan perjalanan berdasarkan rute optimasi yang telah dihasilkan. Contoh informasi biaya terdapat pada gambar 7.

ID	Nama	Harga (Per Liter)	Jarak Tempuh (Per Liter)
BR00001	Premium	4500	6.0

Biaya kendaraan 1 : Rp. 21552.749999999996   Panjang : 28.737 km			
Biaya kendaraan 2 : Rp. 21649.5   Panjang : 28.866 km			
Biaya kendaraan 3 : Rp. 10167.75   Panjang : 13.557 km			
Biaya kendaraan 4 : Rp. 24618.75   Panjang : 32.825 km			
Biaya kendaraan 5 : Rp. 36792   Panjang : 49.056 km			
Biaya Seluruh Rute Kendaraan : Rp. 114780.75			
Panjang Seluruh Rute Kendaraan : 153.041 km			

Gambar 7. Informasi biaya perkiraan perjalanan

## 7. KESIMPULAN

Dari hasil uji coba perangkat lunak yang telah dikembangkan, dapat ditarik beberapa kesimpulan yaitu :

1. Penerapan Algoritma *Simulated Annealing* pada sistem yang dibuat telah sesuai dengan proses algoritma *Simulated Annealing* yang ada. Hal ini dibuktikan dengan tercapainya nilai *global optimum* pada setiap uji coba data. Tercapainya nilai *global optimum* merupakan tujuan dari penggunaan algoritma *Simulated Annealing*.
2. Teknik *cooling schedule* dengan rumus  $T_i = T_0 (T_N / T_0)^{i/N}$  dan nilai jumlah iterasi 500 adalah teknik *cooling schedule* dan nilai jumlah iterasi yang paling tepat untuk digunakan pada penyelesaian permasalahan yang terdapat pada proyek akhir ini. Hal ini dibuktikan dengan diperolehnya nilai terbaik pada uji coba data saat menggunakan kombinasi kedua nilai tersebut.
3. Pengolahan data-data pendistribusian barang PT Tirta Bahagia menggunakan VRP dan metode *Simulated Annealing* dapat menghasilkan proses pendistribusian barang yang optimal.
4. Sistem yang dibangun menghasilkan nilai *output* yang dapat divisualisasikan dengan menggunakan teknologi *Google Maps*.

## 8. DAFTAR PUSTAKA

- [1] Wiratno, Stefanus Eko, *Perancangan Rute dan Penjadwalan Kendaraan dengan Menggunakan Algoritma Perbaikan Simulated Annealing*, ITS. 2008
- [2] Hendriawan, Bambang Eko, *Implementasi Algoritma Pralel Genetic Algorithm untuk Penyelesaian Heterogeneous Fleet Vehicle Routing Problem*, ITS. 2007
- [3] Kusumadewi, S. & Purnomo, H. *Penyelesaian Masalah Optimasi dengan Teknik-teknik Heuristik*. Yogyakarta: Graha Ilmu. 2005
- [4] Dianta, Ashafidz Fauzan. 2010. *Prediksi Luasan Lumpur Dengan Time Series Menggunakan Simulated Annealing*. Surabaya : PENS-ITS
- [5] Google Maps API Tutorial Econym. <http://econym.org.uk/gmap/index.htm>. 14 Juli 2011