

# OPTIMASI PENENTUAN RUTE KENDARAAN PADA SISTEM DISTRIBUSI BARANG DENGAN ANT COLONY OPTIMIZATION

**Gunawan<sup>1</sup>, Indra Maryati<sup>2</sup>, Henry Kurniawan Wibowo<sup>2</sup>**

<sup>1</sup>Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya, Surabaya 60111  
E-mail : gunawan@stts.edu

<sup>2</sup>Jurusan Teknik Informatika, Sekolah Tinggi Teknik Surabaya, Surabaya 60284  
E-mail : maryati@stts.edu, henry\_k\_86@yahoo.com

## ABSTRAK

*Masalah penentuan rute kendaraan atau yang lebih dikenal dengan Vehicle Routing Problem (VRP) merupakan model masalah transportasi yang mengundang banyak minat ilmuwan komputer, yang bertujuan untuk mencari rute optimal untuk sejumlah kendaraan dalam melayani sejumlah customer. VRP memiliki banyak varian sehingga dapat memodelkan bermacam-macam masalah transportasi, logistik dan distribusi. Makalah ini membahas tentang masalah distribusi barang dengan menggunakan varian Capacitated VRP (CVRP). Masalah VRP dapat diselesaikan dengan menggunakan beberapa metode, yaitu Brute-force Search yang termasuk exact methods, Nearest Neighbor Insertion (NNI) yang termasuk metode heuristik, dan Ant Colony Optimization (ACO) yang termasuk metode metaheuristik. Aplikasi pendukung dibuat untuk menyelesaikan varian VRP, yaitu CVRP. Makalah ini membuktikan bahwa algoritma ACO dapat menghasilkan solusi yang lebih mendekati optimal dibandingkan dengan Brute-force Search maupun NNI. Penjelasan juga diberikan meliputi contoh kasus sederhana untuk setiap varian, representasi grafnya, dan solusi optimal untuk masalah tersebut.*

**Kata kunci :** Vehicle Routing Problem, CVRP, Ant Colony Optimization, Ant Colony System

## 1. PENDAHULUAN

Vehicle Routing Problem (VRP) adalah suatu model yang memiliki banyak varian, yang menggambarkan masalah transportasi sebagai model graf, yang bertujuan untuk menemukan rute dengan biaya minimum untuk pengiriman suatu produk kepada sejumlah customer di beberapa lokasi yang berbeda, dengan menggunakan beberapa kendaraan. Setelah diperkenalkan pada tahun 1959 oleh Dantzig dan Ramser melalui makalah mereka yang berjudul "The Truck Dispatching Problem", telah banyak metode atau algoritma yang dipakai atau diadaptasi untuk memecahkan VRP dan varian-varianannya. Yang membuat VRP menarik untuk dibahas adalah model masalah yang metodenya dapat diimplementasikan ke aplikasi yang sangat fungsional di dunia nyata.

Transportasi merupakan bagian yang tidak terpisahkan dari semua sektor industri. Hal itu dikarenakan hampir semua sektor industri selalu mencakup proses distribusi dan logistik. Bahan bakar dan waktu yang digunakan untuk pengiriman adalah biaya (cost) yang cukup besar untuk industri yang melakukan distribusi dengan frekuensi yang rutin ke banyak lokasi. Transportasi selalu memakan biaya, dan oleh karena itu mempengaruhi biaya produksi dan distribusi hingga 10-20% dari total biaya suatu produk. Oleh karena itu, efisiensi di bidang transportasi sangat penting dan dapat secara signifikan mengurangi total biaya produksi dan distribusi.

Untuk mencapai pemakaian sarana transportasi yang ideal, diperlukan suatu model, yang dapat menggambarkan berbagai masalah dalam bidang transportasi. Selain itu, diperlukan metode atau algoritma untuk menyelesaikan model masalah tersebut. Dengan permodelan masalah tersebut, akan memudahkan pencarian solusi (karena dapat dikerjakan oleh komputer dengan menggunakan algoritma tertentu) untuk menemukan rute untuk sejumlah kendaraan dengan biaya minimal.

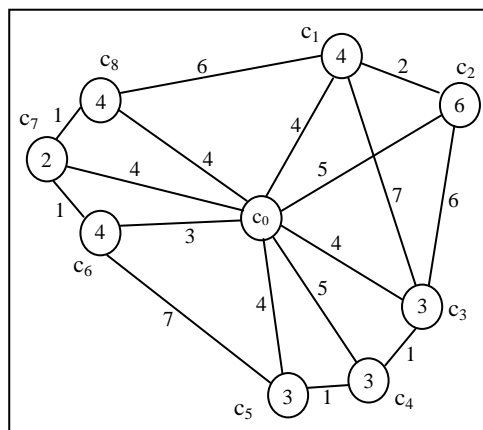
## 2. CAPACITATED VEHICLE ROUTING PROBLEM (CVRP)

Capacitated Vehicle Routing Problem adalah bentuk paling dasar dari Vehicle Routing Problem, yang dapat dilihat sebagai formulasi gabungan dari dua masalah terkenal yang telah ada sebelumnya, yaitu Travelling Salesman Problem (TSP) dan Bin Packing Problem. Pertama kali diperkenalkan melalui makalah yang berjudul The Truck Dispatching Problem, Capacitated Vehicle Routing Problem, sebagai bentuk paling dasar dari VRP, telah menarik minat banyak peneliti dan ilmuwan dari seluruh dunia.

Capacitated Vehicle Routing Problem adalah masalah optimasi untuk menemukan rute dengan biaya minimal (minimum cost) untuk sejumlah kendaraan (vehicles) dengan kapasitas tertentu yang homogen (homogeneous fleet), yang melayani permintaan sejumlah customer yang kuantitas permintaannya telah diketahui sebelum proses pengiriman berlangsung. Kendaraan berangkat dari central depot untuk melakukan pengiriman, dan kembali ke central depot setelah melakukan pengiriman. Diasumsikan jarak atau biaya perjalanan antara semua lokasi telah diketahui. Jarak antara dua lokasi adalah simetris, yang berarti jarak dari lokasi A ke lokasi B sama dengan jarak dari lokasi B ke lokasi A.

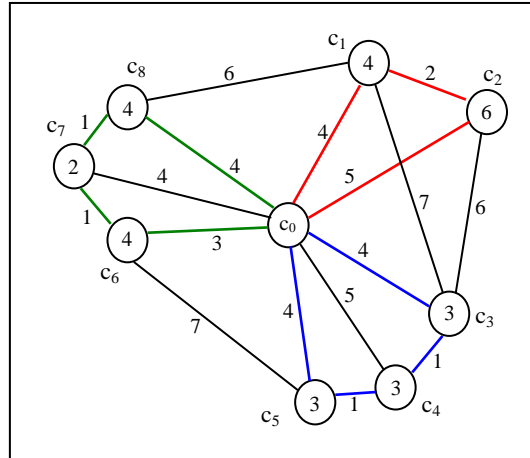
Definisi Capacitated Vehicle Routing Problem hampir sama dengan definisi The Truck Dispatching Problem. Dengan beberapa generalisasi, seperti mengganti truk menjadi kendaraan, bahan bakar menjadi barang, pom bensin menjadi customer/pelanggan, dan terminal utama menjadi central depot (pusat pengiriman), Capacitated Vehicle Routing Problem menjadi model masalah transportasi yang jauh lebih luas.

Agar lebih mengerti tentang konsep Capacitated Vehicle Routing Problem, akan diberikan contoh kasus yang sederhana, disertai representasi graf dan solusi optimalnya. Untuk representasi graf dari contoh kasus ini dapat dilihat pada gambar 2 dibawah ini.



Gambar 1: Representasi Graf untuk Contoh Kasus CVRP

*Vertex/node* ditengah yang dilambangkan  $c_0$  adalah central depot. Sedangkan *vertex-vertex* lain adalah customer. Angka yang berada di dalam *vertex* melambangkan customer's demand atau kuantitas permintaan pelanggan. Sedangkan *edge* (garis yang menghubungkan 2 *vertex*) adalah jalan yang dapat ditempuh ke lokasi lain. Angka yang terdapat di masing-masing *edge* adalah bobot dari *edge* yang merepresentasikan jarak/waktu (cost) yang harus ditempuh untuk sampai pada lokasi yang bersangkutan.



Gambar 2: Representasi Graf untuk Solusi CVRP

Dari permasalahan CVRP diatas, Jumlah kendaraan yang dipakai adalah 3. Kendaraan pertama melewati rute dengan edge berwarna merah. Kendaraan kedua melewati edge warna biru, dan kendaraan ketiga melewati edge warna hijau. Ketiga kendaraan ini akan melayani 8 customer dengan total permintaan 29 satuan. Kendaraan pertama melayani customer dengan total permintaan 10 satuan, dan melewati total jarak 11 satuan. Kendaraan kedua membawa barang 9 satuan, dan melewati total jarak 10 satuan. Kendaraan ketiga membawa barang 10 satuan, dan melewati total jarak 9 satuan. Jadi total jarak yang ditempuh oleh ketiga kendaraan adalah 30 satuan. Rute kendaraan 1 = 0-1-2-0. Rute kendaraan 2 = 0-3-4-5-0. Rute kendaraan 3 = 0-6-7-8-0. Total biaya = 30 satuan.

#### 4. METODE PENYELESAIAN CVRP

Metode penyelesaian CVRP akan dikategorikan menjadi tiga bagian, yaitu metode pasti, metode heuristik, dan metode metaheuristik. Metode pasti yang dibahas adalah Brute Force Search. Sedangkan metode heuristik yang dibahas adalah Nearest Neighbor Insertion (NNI). Untuk metaheuristik, metode yang dibahas adalah Ant Colony Optimization (ACO). Semua metode tersebut akan digunakan untuk melakukan pembuktian pada makalah ini.

##### 4.1. Brute Force Search

Brute Force Search atau Exhaustive Search, yang juga dikenal dengan nama Generate and Test, merupakan metode penyelesaian masalah yang sangat umum dan sederhana, yang secara sistematis membangun (generate) semua kemungkinan kandidat solusi dari suatu masalah dan mengecek (test) apakah tiap kandidat itu valid, dan mengambil satu kandidat valid pertama yang ditemukan atau kandidat terbaik (untuk masalah optimasi) sebagai solusi. Berikut akan diberikan contoh sederhana agar lebih memahami konsep brute force. Untuk mencari faktor dari bilangan  $n$ , maka metode brute force akan menghasilkan kandidat solusi mulai dari 1 sampai  $n$ , dan mengecek apakah tiap kandidat itu merupakan faktor dari  $n$ . Kandidat yang valid adalah kandidat yang merupakan faktor dari bilangan  $n$ .

Pada Vehicle Routing Problem, metode ini akan membangun rute secara sistematis, dan mengecek apakah rute yang dihasilkan valid. Setiap rute yang dihasilkan dianggap valid jika tidak melanggar batasan (constraints) yang telah dijabarkan oleh varian VRP yang diselesaikan. Kemudian metode ini akan mencari kemungkinan rute yang lain. Proses ini berlanjut sampai semua kemungkinan rute telah ditelusuri. Rute dengan biaya minimal akan menjadi output atau solusi.

##### 4.2. Nearest Neighbor Insertion

Metode Nearest Neighbor Insertion (NNI) pertama kali diperkenalkan pada tahun 1983 dan merupakan metode yang sangat sederhana dan tamak (greedy). Cara kerja metode ini adalah sebagai berikut. Pertama-

tama, semua rute kendaraan masih kosong. Dimulai dari rute kendaraan pertama, metode ini memasukkan (insert) satu persatu customer terdekat (nearest neighbor) yang belum dikunjungi ke dalam rute, selama memasukkan customer tersebut ke dalam rute kendaraan tidak melanggar batasan kapasitas maksimum kendaraan tersebut (atau batasan-batasan yang dijabarkan oleh varian VRP yang lain). Kemudian proses yang sama juga dilakukan untuk kendaraan-kendaraan berikutnya, sampai semua kendaraan telah penuh atau semua customer telah dikunjungi.

### 4.3. Ant Colony Optimization

Ant Colony Optimization (ACO) adalah suatu metode penyelesaian masalah optimasi yang berupa kumpulan beberapa algoritma yang menggunakan teknik probabilistik dan prinsip komunikasi koloni semut dalam mencari makanan. Konsep ACO pertama kali diperkenalkan melalui algoritma Ant System (AS) pada tahun 1992 oleh Marco Dorigo dalam disertasinya.

Terinspirasi oleh cara koloni semut dalam mencari rute ke sumber makanan, metode ini meniru sistem komunikasi koloni semut yang meninggalkan zat kimia yang disebut feromon di rute-rute perjalanan mereka. Semut yang menemukan sumber makanan, akan meninggalkan feromon di rute saat kembali ke koloninya. Semut lain yang mencium feromon di suatu rute, akan cenderung untuk mengikuti rute tersebut jika kandungan feromon cukup padat. Semakin padat kandungan feromon pada suatu rute, semakin besar kemungkinan semut lain mengikuti rute tersebut. Feromon akan mengalami penguapan seiring berjalannya waktu. Rute yang pendek akan mengandung feromon yang cukup padat, karena waktu yang digunakan untuk pulang-pergi (tiap kali pulang ke koloninya, semut selalu meninggalkan feromon) dari koloni ke sumber makanan lebih sedikit, yang menyebabkan penguapan feromon menjadi minimal. Selain feromon, yang juga mempengaruhi semut dalam memilih rute adalah *visibility*. *Visibility* merupakan naluri semut untuk menemukan rute terdekat menuju sumber makanan ataupun kembali ke koloninya. Dua komponen penting ini, yaitu feromon dan *visibility*, akan digunakan ACO sebagai komponen utama.

Algoritma ACO yang dipakai untuk menyelesaikan 3 varian VRP dalam aplikasi yang dibuat untuk tugas akhir ini adalah Ant Colony System (ACS). Dalam aplikasinya untuk menyelesaikan masalah VRP, metode ini akan menggunakan satu koloni semut artifisial, yang tiap semut artifisial akan menghasilkan 1 solusi dalam 1 iterasi. Jumlah semut artifisial dalam suatu koloni dapat diatur sesuai keinginan.

#### Algoritma 1: Ant Colony System

```
1. SetPheromoneInitialValue()
2. REPEAT
3.   FOR k ← 1 TO numberOfAnts DO
4.     GenerateTour(k)
5.     LocalPheromoneUpdate()
6.   ENDFOR
7.   GlobalPheromoneUpdate()
8. UNTIL (stopCriterion())
```

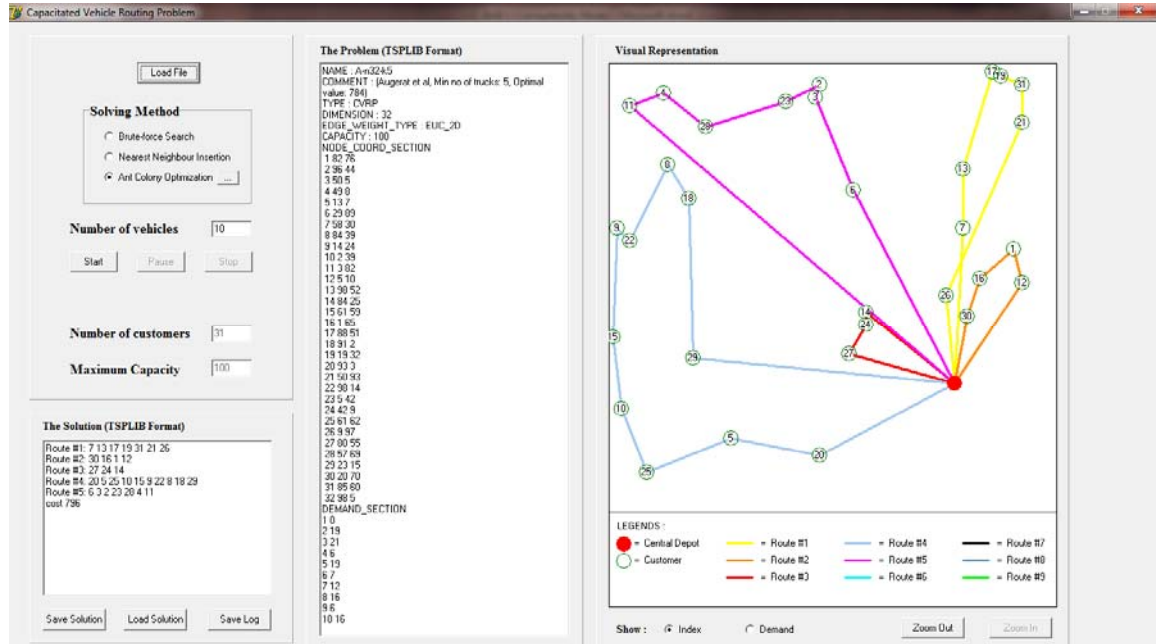
Fungsi *SetPheromoneInitialValue()* berguna untuk memberi nilai awal feromon di semua jaringan jalan. Nilai awal feromon yang umum dipakai adalah  $1/N \cdot J_{\psi}^h$ , dimana N adalah jumlah customer dan  $J_{\psi}^h$  adalah total biaya dari solusi yang dihasilkan menggunakan heuristik Nearest Neighbor. Variabel *numberOfAnts* menampung jumlah semut arfisial dalam koloni. Fungsi *GenerateTour()* akan membentuk rute/solusi. Fungsi *LocalPheromoneUpdate()* untuk mengubah nilai feromon secara lokal, dan *GlobalPheromoneUpdate()* untuk mengubah nilai feromon secara global. Penjelasan tentang perubahan nilai feromon akan diberikan lebih lanjut di bagian berikutnya. Fungsi *stopCriterion()* berguna untuk mengecek apakah kriteria untuk berhenti sudah terpenuhi. Kriteria berhenti ini dapat berupa jumlah iterasi sudah terpenuhi, atau sumber daya waktu yang diberikan sudah habis.

Pembentukan rute oleh semut artifisial dipengaruhi oleh 2 komponen utama ACO, yaitu feromon dan *visibility*. *Visibility* ini tergantung dari jenis masalah yang ingin diselesaikan (*problem-specific*). Dalam kasus VRP, *visibility* berhubungan dengan jarak/waktu (*cost*) ke calon *customer* selanjutnya. Semakin kecil

cost ke calon *customer* tersebut, semakin besar kemungkinan calon *customer* tersebut terpilih untuk dikunjungi selanjutnya.

## 5. HASIL DAN PEMBAHASAN

Hasil dari makalah ini adalah membuktikan bahwa Ant Colony Optimization lebih baik dari pada 2 metode lainnya yaitu Brute Force Search dan Nearest Neighbor Insertion. Dan untuk mendukung pembuktian tersebut peneliti membuat sebuah aplikasi program untuk melakukan pencarian rute kendaraan berdasarkan dataset dan aplikasi ini juga merepresentasikan rute tersebut ke bentuk graf. Untuk tampilan dari aplikasi program dapat dilihat pada gambar 3.



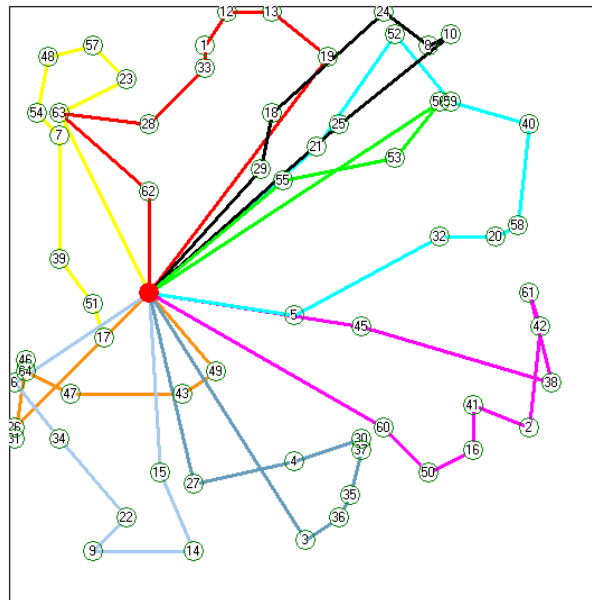
Gambar 3: Tampilan Aplikasi Program

Proses ujicoba akan dilakukan pada dataset yang merupakan open problem. Untuk varian CVRP, akan digunakan *dataset* yang dibuat oleh Augerat et. al. yang mempunyai format TSPLIB. Karena tidak ditemukan *instance* dengan skala kecil, maka beberapa *instance* dibuat sendiri, untuk melakukan *testing* terhadap metode Brute-Force Search yang tidak dapat menyelesaikan masalah dengan skala relatif besar. Tabel 1 menunjukkan hasil *testing* dari 3 metode penyelesaian terhadap *dataset* CVRP.

Tabel 1: Hasil Testing CVRP

Instance	Solution Cost (Number of Vehicles Used)			
	Best Known	Brute Force	NNI	ACS (ACO)
CVRP-n8-k3	375 (3)	375 (3)	375 (3)	375 (3)
CVRP-n10-k3	455 (3)	455 (3)	477 (4)	455 (3)
A-n32-k5	784 (5)	-	961 (5)	787 (5)
A-n33-k5	661 (5)	-	829 (5)	667 (5)
B-n41-k6	829 (6)	-	1184 (7)	834 (6)
B-n44-k7	909 (7)	-	1194 (8)	926 (7)
A-n53-k7	1010 (7)	-	1313 (8)	1054 (7)
A-n55-k9	1073 (9)	-	1320 (10)	1104 (9)
B-n57-k7	1153 (7)	-	1500 (8)	1173 (8)
B-n64-k9	861 (9)	-	1093 (10)	901 (9)
A-n65-k9	1177 (9)	-	1465 (10)	1250 (9)

Dua masalah pertama, CVRP-n8-k3 dan CVRP-n10-k3, merupakan masalah berskala kecil yang dibuat sendiri. Pada dua masalah ini, metode Brute-Force Search dapat menemukan solusi optimal. Dua metode lain, NNI dan ACS, juga dapat menemukan solusi optimal untuk CVRP-n8-k3. Untuk CVRP-n10-k3, NNI gagal memberikan solusi optimal, sedangkan ACS dapat menemukan solusi optimal dari masalah tersebut. Nama masalah mengandung informasi mengenai jumlah *customer* dan kendaraan yang dipakai dalam solusi terbaik yang ditemukan. Contoh, A-n65-k9, berarti jumlah *customer* dan *central depot* adalah 65, sedangkan kendaraan yang dipakai di solusi terbaik adalah 9. Dengan menggunakan dataset A-n65-k9, didapatkan 9 rute kendaraan sesuai dengan jumlah kendaraan yang digunakan. Rute tersebut yaitu rute kendaraan 1: 17-51-39-7-54-48-57-23-11, rute kendaraan 2: 49-43-47-64-46-31-26, rute kendaraan 3: 62-63-28-33-1-12-13-19, rute kendaraan 4: 6-34-22-9-14-15, rute kendaraan 5: 60-50-16-41-2-42-61-38-45, rute kendaraan 6: 5-32-20-58-40-59-52-21, rute kendaraan 7: 29-18-24-8-10-25, rute kendaraan 8: 3-36-35-37-30-4-27, dan rute kendaraan 9: 55-53-56-44. Total biaya yang dibutuhkan adalah 1323. Untuk representasi graf dapat dilihat pada gambar



Gambar 4: Representasi Graf untuk Solusi CVRP A-n65-k9

Untuk metode ACS, *setting* parameter yang digunakan adalah 1000 iterasi, 20 semut artifisial,  $\beta = 2$ ,  $p$  (*evaporation rate*) = 0,1 (10%), dan  $q_0$  (probabilitas *exploitation*) = 0,9 (90%). Untuk tiap masalah, metode ACS dijalankan 10 kali, dan diambil satu solusi dengan *cost* minimum. *Setting* parameter ini juga berlaku untuk varian SDVRP dan VRPTW.

Dari hasil yang ditampilkan di tabel 1, dapat dilihat bahwa ACS yang termasuk dalam metode metaheuristik, merupakan metode yang terbaik dalam memberikan solusi untuk varian CVRP jika dibandingkan Brute-Force Search dan NNI. Metode ACS mampu memberikan solusi yang optimal untuk 2 *instance* yang dibuat sendiri, dan solusi yang mendekati optimal untuk 9 *instance* dari *dataset* yang dibuat Augerat et. al. dengan format TSPLIB.

## 7. PENUTUP

Dari hasil pengamatan yang dilakukan untuk setiap tahap yang dilewati dalam pembuatan makalah ini, maka dapat ditarik beberapa kesimpulan seperti berikut:

- VRP merupakan masalah yang lebih kompleks dan sulit jika dibandingkan TSP dan BPP. Metode pasti dapat menyelesaikan masalah TSP dan BPP dengan skala yang relatif besar, sedangkan masalah VRP skala menengah ada yang belum dapat diselesaikan oleh metode pasti.
- Metode Brute-Force Search hanya dapat menyelesaikan masalah CVRP dengan skala kecil. Hal ini dikarenakan metode ini menelusuri setiap kandidat solusi yang jumlahnya menjadi sangat besar

dengan bertambahnya skala masalah.

- Metode NNI dapat membutuhkan waktu komputasi yang sangat sedikit, tetapi tidak mampu memberikan solusi yang mendekati optimal untuk masalah berskala besar. Penggunaan heuristik yang sangat sederhana menyebabkan kualitas solusi yang dihasilkan metode NNI menjadi semakin jauh dari optimal seiring bertambahnya skala masalah CVRP yang diselesaikan.
- Metode ACO merupakan metode penyelesaian terbaik jika dibandingkan dengan Brute-Force Search dan NNI. Metode ACO mampu menyelesaikan masalah CVRP berskala besar dan mampu menghasilkan solusi yang mendekati optimal. Selain itu waktu komputasi atau jumlah iterasi ACO dapat diatur sesuai dengan sumber daya waktu yang dimiliki.
- Hanya metode heuristik dan metaheuristik yang dapat diimplementasikan di sistem riil. Hal ini dikarenakan metode pasti (*exact methods*) tidak dapat menyelesaikan masalah berskala besar yang merupakan ciri khas sistem riil.

## DAFTAR PUSTAKA

- [1] Applegate and Bixby and Chvatal and Cook. 2007. The Traveling Salesman Problem. Princeton University Press
- [2] Clausen, Jens. 1999. Branch and Bound Algorithm – Principles and Examples. University of Copenhagen
- [3] Dantzig and Ramser. 1959. The Truck Dispatching Problem. INFORMS
- [4] Diaz, Bernabé. 2006. CVRPTW Instances. ([http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem\\_Instances/CVRPTWInstances.html](http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem_Instances/CVRPTWInstances.html))
- [5] Diaz, Bernabé. 2006. Description for Files of Cordeau's Instances. ([http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html?/Problem\\_Instances/CordeauFilesDesc.html](http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html?/Problem_Instances/CordeauFilesDesc.html))
- [6] Diaz, Bernabé. 2006. MDVRP Instances. ([http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem\\_Instances/MDVRPInstances.html](http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem_Instances/MDVRPInstances.html))
- [7] Diaz, Bernabé. 2006. SDVRP Instances. ([http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem\\_Instances/SDVRPInstances.html](http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem_Instances/SDVRPInstances.html))
- [8] Diaz, Bernabé. 2006. VRPPD Instances. ([http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html?/Problem\\_Instances/VRPPDInstances.html](http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html?/Problem_Instances/VRPPDInstances.html))
- [9] Dorigo, Marco and Birattari, Mauro and Stutzle, Thomas. 2006. Ant Colony Optimization. IRIDIA
- [10] Dorigo, Marco and Gambardella, Luca. 1996. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. IRIDIA
- [11] Lysgaard, Jens. 1997. Clarke & Wright's Savings Algorithm. The Aarhus School of Business
- [12] Malkevitch, Joseph. \_\_\_\_\_. Bin Packing. (<http://www.ams.org/samplings/feature-column/fcarc-bins1>)
- [13] Reinelt, Gerhard. 1997. TSPLIB. (<http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html>)
- [14] Scholl and Klein. 2003. Bin Packing. (<http://www.wiwi.uni-jena.de/Entscheidung/binpp/>)
- [15] Skorobohatyj, Georg. 1995. MP-TESTDATA - The TSPLIB Capacitated Vehicle Routing Problem Instances. (<http://elib.zib.de/pub/mp-testdata/tsp/tsplib/vrp/index.html>)
- [16] Takes, Frank. 2010. Applying Monte Carlo Techniques to the Capacitated Vehicle Routing Problem. Leiden University
- [17] Toth, Paolo and Vigo, Daniele. 2002. The Vehicle Routing Problem. SIAM
- [18] \_\_\_\_\_. Ant Colony Optimization. ([http://en.wikipedia.org/wiki/Ant\\_colony\\_optimization](http://en.wikipedia.org/wiki/Ant_colony_optimization))
- [19] \_\_\_\_\_. Bin Packing Problem. ([http://en.wikipedia.org/wiki/Bin\\_packing\\_problem](http://en.wikipedia.org/wiki/Bin_packing_problem))
- [20] \_\_\_\_\_. Branch and bound. ([http://en.wikipedia.org/wiki/Branch\\_and\\_bound](http://en.wikipedia.org/wiki/Branch_and_bound))
- [21] \_\_\_\_\_. Brute-force search. ([http://en.wikipedia.org/wiki/Brute-force\\_search](http://en.wikipedia.org/wiki/Brute-force_search))
- [22] \_\_\_\_\_. NP-hard. (<http://en.wikipedia.org/wiki/NP-hard>)
- [23] \_\_\_\_\_. Travelling Salesman Problem. ([http://en.wikipedia.org/wiki/Travelling\\_salesman\\_problem](http://en.wikipedia.org/wiki/Travelling_salesman_problem))
- [24] \_\_\_\_\_. Vehicle Routing Problem. ([http://en.wikipedia.org/wiki/Vehicle\\_routing\\_problem](http://en.wikipedia.org/wiki/Vehicle_routing_problem))