

PERANCANGAN DAN IMPLEMENTASI METODE HEURISTIC UNTUK VEHICLE ROUTING PROBLEM PADA PENGANGKUTAN SAMPAH

DESIGN AND IMPLEMENTATION OF HEURISTIC METHOD FOR VEHICLE ROUTING PROBLEM IN WASTE TRANSPORT

Raden Rogers Dwiputra Setiady¹, Agung Nugroho Jati², Fairuz Azmi³

^{1,3}Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Universitas Telkom
¹rogersdwiputra@students.telkomuniversity.ac.id, ²agungnj@telkomuniversity.ac.id,
³worldliner@telkomuniversity.ac.id

Abstrak

Pengangkutan sampah merupakan salah satu masalah logistik kompleks yang ada di setiap kota. Beberapa tahun terakhir kenaikan harga BBM, biaya operasional dan beban regulasi menyebabkan transportasi pengiriman sampah menghasilkan biaya yang cukup besar. Masalah tersebut menjadi potensi besar untuk optimalisasi dan penghematan. Selain biaya pengangkutan yang tinggi, masalah terjadi juga pada pengangkutan kontainer sampah yang belum penuh atau sudah terlalu penuh karena petugas hanya mengangkut berdasarkan jadwal, akibatnya terjadi pemborosan saat truk mengambil kontainer sampah yang belum penuh. Masalah rute pengangkutan tersebut adalah *Vehicle Routing Problem*. Pada penelitian ini dibangun sebuah sistem yang menerima informasi TPS yang penuh untuk selanjutnya dilakukan komputasi menggunakan algoritma yang dirancang untuk mendapatkan urutan kunjungan TPS dengan nilai *cost* (Total waktu tempuh) yang paling rendah. Urutan kunjungan TPS ini kemudian dibagikan ke beberapa truk dengan batasan waktu kerja masing – masing truk 6 jam. Pengujian pada algoritma dilakukan dengan memberikan beberapa masukan jumlah TPS yang penuh. Berdasarkan pengujian yang telah dilakukan, Sistem yang dibuat dapat mengatur proses logistik pengangkutan sampah berdasarkan daftar TPS yang penuh lalu sistem membuat rute kunjungan TPS untuk ditampilkan menjadi panduan rute di aplikasi *mobile*. Algoritma untuk menentukan rute dengan nilai waktu tempuh terendah memiliki nilai total *cyclomatic complexity* 11 dan waktu komputasi tertinggi yang berhasil di uji adalah 3.1 Jam untuk 12 TPS.

Kata Kunci: *vehicle routing problem*, vrp, heuristik, pengangkutan sampah, rute, kota Bandung.

Abstract

Transporting waste is one of the complex logistics problems that exist in every town. The last few years the increase in fuel prices, operating costs and regulatory burdens caused by transporting garbage generate considerable cost. The problem becomes a large potential for optimization and savings. In addition to high transportation costs, problems occur also in the transport of waste containers are not full or too full because the officer only transport based on a schedule, resulting in wastage when the trucks take the garbage containers that have not been filled. The problem is called Vehicle Routing Problem. In this study constructed a system that receives polling information with full condition, the computation is then performed using an algorithm that is designed to get the order of visits with the lowest cost value (total travel time). Sequence of TPS visit is then distributed to several trucks with 6-hours limit time for each truck. Tests on the algorithm is done by giving some *input* number of polling stations were full. Based on the whitebox test results, each stage of the algorithm running well and provide output in accordance with the draft, but the computing time testing showed high computational time. According to the test, the system has been made to organize the logistics process of waste collection, and create the lowest cost route for navigation systems in *mobile* applications. Algorithm to determine the route with the lowest value of travel time has a total cyclomatic complexity value of 11 and the highest computing time succeeded in the test is 3.1 hours for 12 TPS.

Keyword: *vehicle routing problem*, VRP, heuristics, waste transportation, route, Bandung

1. Pendahuluan

Pengumpulan sampah adalah salah satu masalah logistik yang paling kompleks yang dihadapi kota manapun. Dalam beberapa tahun terakhir kenaikan harga BBM, biaya operasional dan beban regulasi yang berkembang telah menyebabkan perusahaan pengumpulan sampah baik negara maupun swasta untuk mengoptimalkan rute pengumpulan sampah mereka [1]. Menurut penelitian VN. Bhat (1996) biaya transportasi merupakan antara 70% dan 80% dari

seluruh biaya operasional dalam pengumpulan sampah [2]. Oleh karena itu bahkan perbaikan kecil di rute pengumpulan sampah dapat menyebabkan penghematan besar [1].

Pola pengangkutan sampah di Kota Bandung dilakukan dengan sistem tidak langsung. Sistem tidak langsung adalah sistem pengangkutan sampah dimana sampah tidak langsung diangkut ke TPA dari sumbernya, melainkan sampah dikumpulkan dulu di TPS sebelum diangkut menuju TPA [3].

Tabel 1 Kendaraan Operasional PD Kebersihan Kota Bandung (per 1 Juli 2008) [2]

No.	Jenis Kendaraan	Kapasitas Angkut (M ³)	Jumlah Truk di Wilayah Operasional (Unit)				Total
			Bandung Barat	Bandung Utara	Bandung Selatan	Bandung Timur	
1.	Dump Truck	10	2	5	5	6	18
2.	Dump Truck	6	8	7	4	2	21
Jumlah			10	12	9	8	39

Tabel 1 Kebutuhan PD Kebersihan Kota Bandung (per Juli 2008) [2]

No.	Uraian	Yang ada	Kebutuhan	Kekurangan
1.	TPS	184	225	41
2.	Kontainer (10 M ³ dan 6 M ²)	198	450	252
3	TPA	1	2	1

Tabel diatas menunjukkan jumlah TPS lebih banyak dari jumlah Truk pengangkut yang tersedia. Saat ini, metode pengangkutan sampah dilakukan dengan cara mendatangi lokasi TPS yang ada sesuai jadwal dan kebiasaan. Tidak peduli kontainer sampah dalam keadaan penuh atau tidak, dan tidak adanya rute tetap hanya berdasarkan kebiasaan operator truk. Optimalisasi dilakukan dengan menggunakan metode heuristik, sistem akan mendapatkan data informasi TPS yang sudah atau hampir penuh dari *embedded sistem* dan membuat rute dengan total jarak tempuh terpendek dan waktu tempuh terpendek berdasarkan kecepatan rata – rata truk untuk diberikan kepada masing – masing truk. Dengan adanya optimalisasi rute ini berdampak pada efisiensi energi, waktu, dan optimalisasi sumber daya yang ada.

2. Tinjauan Pustaka

2.1 Vehicle Routing Problem

Vehicle Routing Problem (VRP) adalah permasalahan optimasi penentuan rute dengan keterbatasan kapasitas kendaraan. Pada permasalahan ini, ada sebuah depot awal dan sejumlah *n* tempat untuk dikunjungi dengan *demand* yang dapat berbeda-beda. Sebuah kendaraan diharapkan untuk memenuhi permintaan setiap tempat tersebut dari depot. Diusulkan oleh Dantzig dan Ramser pada tahun 1959, VRP merupakan masalah penting di bidang transportasi, distribusi dan logistik. Seringkali konteksnya adalah bahwa dari pengiriman barang yang terletak di sebuah depot pusat untuk pelanggan yang telah menempatkan pesanan untuk barang-barang tersebut. Tujuan dari VRP adalah untuk meminimalkan biaya rute keseluruhan [1].

2.2 Metode Heuristik

Heuristik berasal dari Bahasa Yunani yang berarti menemukan. Heuristik digunakan pada bidang optimasi untuk mengkarakterisasi jenis tertentu pemecahan dari sebuah masalah. Pada Heuristik dilakukan prosedur yang efisien untuk menemukan solusi yang baik meskipun solusi tersebut bukan yang paling optimal. Berbeda dengan metode yang tepat, yang menjamin untuk memberikan solusi optimal dari sebuah masalah, metode

heuristik hanya mencoba untuk menghasilkan yang baik, tapi belum tentu optimal. Meskipun demikian, waktu yang dibutuhkan oleh metode yang tepat untuk menemukan solusi optimum untuk sebuah masalah sangat sulit, jika memang metode seperti itu ada, metode tersebut jauh lebih besar urutan besarnya dari yang heuristik (kadang-kadang pada banyak kasus itu tidak dapat diterapkan). Sehingga lebih sering menggunakan metode-metode heuristik untuk memecahkan masalah optimasi nyata [4].

2.3 REST

Representational State Transfer atau disingkat REST adalah suatu arsitektur dalam *web service* untuk melakukan pertukaran data yang berorientasi pada sumber daya *Mobile* atau resource melalui protokol HTTP. Berikut penjelasan Roy Fielding mengenai *Representational State Transfer*: “*Representational State Transfer is intended to evoke an image of how a well-designed Web Application behaves: a network of web pages (a virtual state-machine), where the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use*” [5].

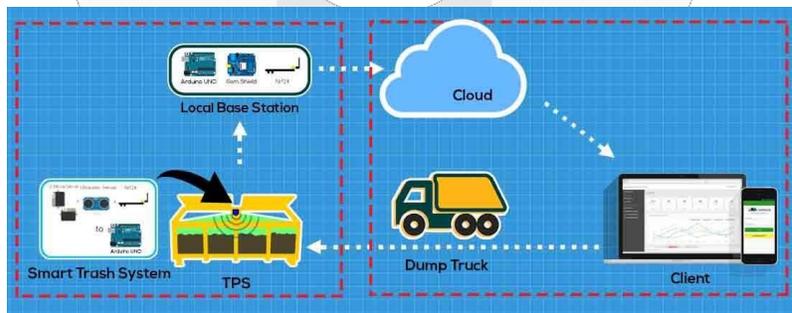
REST secara spesifik merujuk pada suatu prinsip-prinsip arsitektur jaringan yang menggariskan pendefinisian dan pengalamanan sumber daya. Istilah ini sering digunakan untuk mendeskripsikan semua antarmuka sederhana yang menyampaikan data dalam domain spesifik melalui HTTP tanpa tambahan lapisan pesan seperti SOAP atau pelacakan sesi menggunakan cookie HTTP [5].

2.4 Google Direction Web Service

Google Directions Service adalah *Web service* yang menghitung arah antara lokasi menggunakan permintaan REST. Arah dapat menentukan asal-usul, tujuan dan *waypoints* baik sebagai string teks (misalnya "Chicago, IL") atau sebagai koordinat lintang / bujur. Layanan *Google Directions Service* dapat memberikan respon beberapa titik arah [6].

3. Perancangan

3.1 Gambaran Umum Sistem



Gambar 1 Gambaran Umum Sistem

Berdasarkan gambar ilustrasi diatas, pada bagian *server* akan menerima data status Tempat Pembuangan Sampah (TPS) yang dikirim oleh sensor melalui protokol HTTP dengan arsitektur RestFul, masukan data tersebut ditangani oleh aplikasi untuk disimpan ke *database*. Setelah informasi status TPS tersebut disimpan pada *Database*, setiap harinya pada pukul 20.00 waktu setempat akan dilakukan kalkulasi rute dengan nilai *cost* terendah menggunakan Algoritma solusi *Vehicle Routing Problem* berbasis metode Heuristik. *Cost* terendah adalah total waktu tempuh rute. Aplikasi *mobile* akan menerima daftar rute yang dihasilkan dari Algoritma tersebut, dan Supir Truk sebagai pengguna Aplikasi *Mobile* akan memilih rute lalu menggunakan aplikasi *mobile* sebagai panduan pengangkutan sampah.

3.2 Deskripsi Sistem

Sistem yang dibangun merupakan sebuah Algoritma yang di implementasikan pada *server* untuk melakukan komputasi rute dengan nilai *cost* terendah. *Cost* terendah adalah total waktu tempuh rute. Algoritma tersebut akan memilih TPS yang dinyatakan penuh atau layak untuk diangkut lalu dilakukan perhitungan *cost* pada setiap

possible route yang akan terjadi. Setelah mendapatkan rute dengan cost terendah, rute tersebut akan disegmentasi sesuai batas waktu maksimal yaitu 6 jam dan disimpan di database. Setelah daftar rute tersedia, Pada sistem ini juga terdapat aplikasi Android yang berfungsi sebagai panduan navigasi untuk supir truk.

Selain sebagai panduan navigasi bagi supir truk, Aplikasi Android akan mengirim informasi kecepatan rata – rata truk ketika mengunjungi tiap node. Kecepatan rata – rata ini akan digunakan oleh algoritma untuk menentukan cost berdasarkan nilai kecepatan yang sebenarnya, sehingga nilai kecepatan pada tiap node di algoritma tidak statis.

3.3 Rancangan Algoritma

Pada tahap awal algoritma, dilakukan operasi menentukan jumlah kemungkinan kombinasi urutan TPS yang mungkin terjadi, dengan cara permutasi (AB != BA). Pada tahap ini ditentukan juga jumlah nilai cost-nya.

Berikut adalah notasi matematika untuk menentukan peluang permutasi rute yang mungkin terjadi :

$$n! \dots \dots (1)$$

Dan untuk menghitung total cost dari masing – masing rute hasil permutasi :

Tabel 3 Notasi

	Jarak dari node i ke j
	Kecepatan truk dari node i ke j, nilai kecepatan ini akan diperbarui setiap terjadi kunjungan dari node i ke j.
	Waktu tempuh dari node i ke j
cost	Nilai optimal
P	Pool Truk, tempat truk memulai
	TPS ke n
TPA	Tempat Pembuangan Sampah Akhir
L	Waktu bongkar muat bak sampah

$$\dots \dots (2)$$

$$C_{total} = (C_{i,j} + L) + (C_{j,k} + L) + (C_{k,l} + L) + (C_{l,m} + L) + \dots \dots (3)$$

Setelah mendapatkan nilai cost dari masing – masing permutasi, dilakukan pengurutan nilai cost terendah. Pengurutan dilakukan menggunakan algoritma quicksort, lalu rute tersebut disegmentasi menjadi beberapa rute sesuai batasan waktu maksimal setiap rute.

4. Pembahasan

4.1 Pengujian Algoritma dengan metode Whitebox

Pengujian algoritma dilakukan dengan metode whitebox dengan tujuan dapat mengetahui kerja dari setiap tahapan pada implementasi algoritma. Pengujian algoritma dilakukan tiga tahap sesuai langkah – langkah blok kode program pada algoritma tersebut yaitu blok permutasi, blok quicksort dan blok segmentasi rute. Parameter pengujian ini adalah nilai cyclomatic complexity.

Pseudocode	Flow Graph
<pre> if n = 1 then for setiap A cost = cost + waktu tempuh dari berangkat ke tujuan + waktu bongkar muat end for output(cost) else for i := 0; i < n-1; i++ do if n is even then swap(a[i],a[i-1]) else swap(a[0],a[n-1]) end if end for tahap1(n-1,a) end if </pre>	

Nilai *Cyclomatic complexity* untuk algoritma blok pertama adalah :

$$\begin{aligned}
 CC &= N + 2 \\
 &= 11 - 11 + 2
 \end{aligned}$$

Nilai *Cyclomatic complexity* untuk algoritma blok pertama adalah 2 dengan *path* sebagai berikut .:

1. 1-2-3-4-6-7-8-9-10-11
2. 1-2-3-5-6-7-8-9-10-11

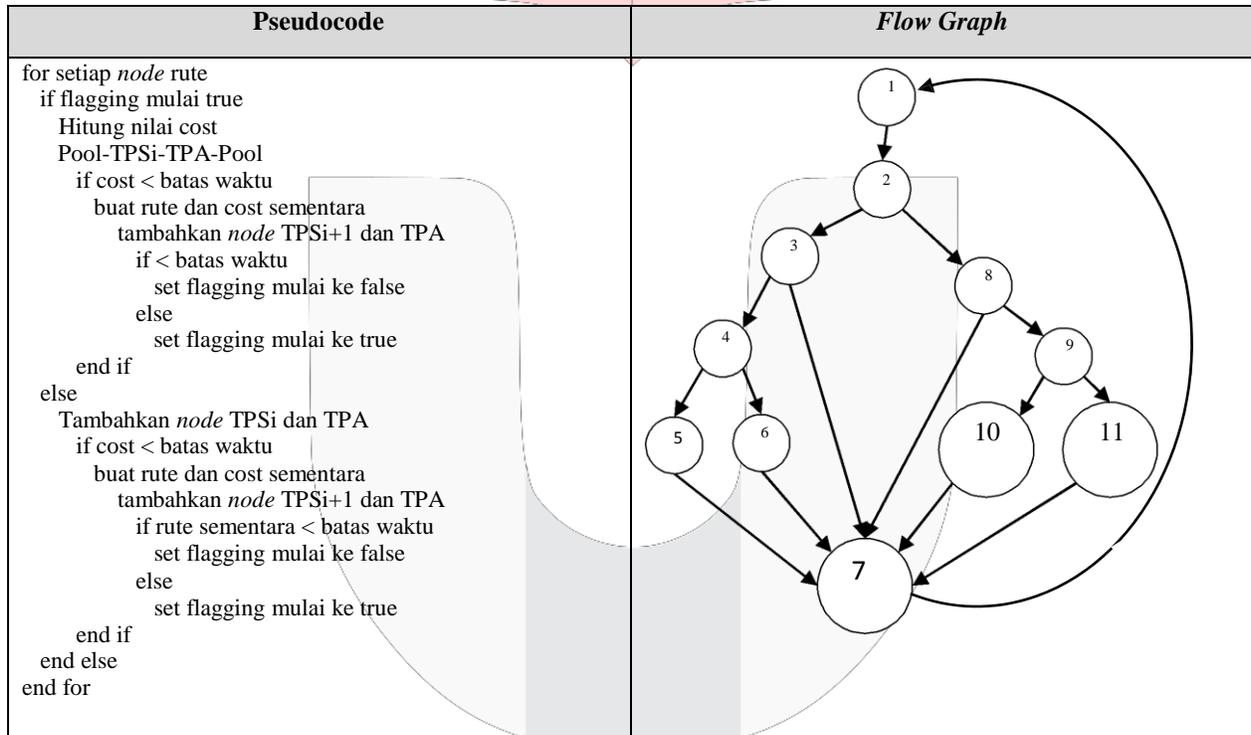
Pseudocode	Flow Graph
<pre> int i : left int j : right separator : (left + right) / 2 while left <= right while A[i] < separator i++ end while while A[i] > separator i-- end while if i<=j swap A[i] A[j] i++ j++ end if end while if left < j A = quicksort(A, left, j) end if if right > i A = quicksort(A,i,right) end if output(A) </pre>	

Nilai *Cyclomatic complexity* untuk algoritma blok kedua adalah :

$$\begin{aligned}
 \text{CC} &= E - N + 2 \\
 &= 14 - 12 + 2 \\
 &= 4
 \end{aligned}$$

Nilai *Cyclomatic complexity* untuk algoritma blok kedua adalah 4 dengan *Path*-nya sebagai berikut :

1. 1-2-3-4-5-6-7-8-9-10-11-12
2. 1-2-3-4-5-6-8-9-10-11-12
3. 1-2-3-4-5-6-8-10-11-12
4. 1-2-3-4-5-6-8-10-12



Nilai *Cyclomatic complexity* untuk algoritma blok ketiga adalah :

$$\begin{aligned}
 \text{CC} &= E - N + 2 \\
 &= 15 - 11 + 2 \\
 &= 6
 \end{aligned}$$

Nilai *Cyclomatic complexity* untuk algoritma blok kedua adalah 6 dengan *Path*-nya sebagai berikut :

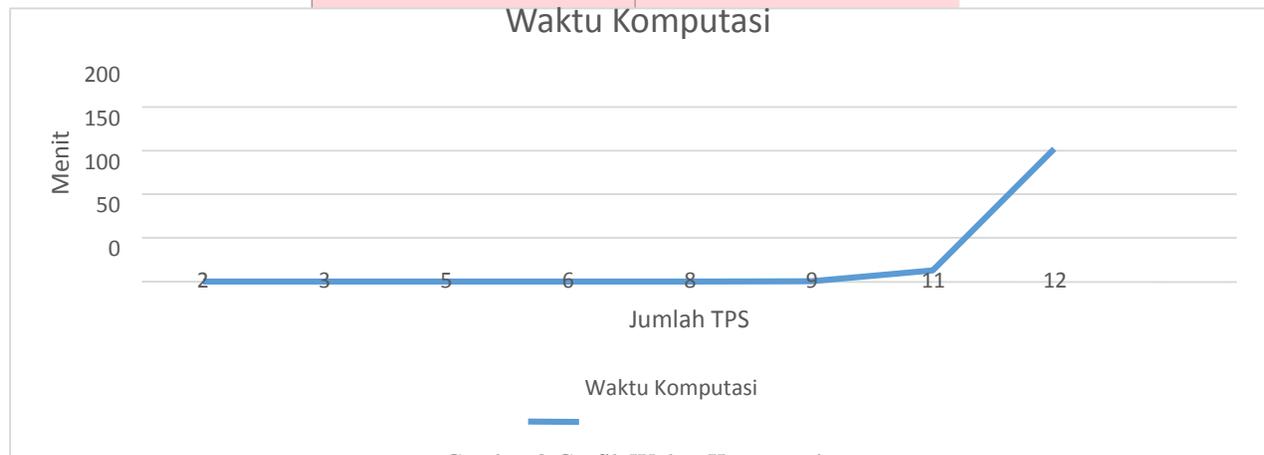
1. 1-2-3-7
2. 1-2-8-7
3. 1-2-3-4-5-7
4. 1-2-3-4-6-7
5. 1-2-8-9-10-7
6. 1-2-8-11-7

Berdasarkan nilai *Cyclomatic complexity* hasil pengujian, untuk melakukan pengujian algoritma tersebut terdapat minimal 12 skenario dengan path seperti di atas.

4.2 Pengujian Waktu Komputasi Algoritma

Pengujian waktu komputasi dilakukan dengan tujuan mengetahui waktu yang dibutuhkan oleh algoritma memberikan output dengan masukan yang berbeda, selain itu tujuan pengujian ini untuk melakukan evaluasi hasil implementasi sebelum diterapkan pada kasus nyata.

Parameter pengujian waktu komputasi dalam satuan menit, diberikan masukan TPS yang penuh dengan jumlah yang berbeda, setiap jumlah TPS penuh dilakukan pengujian sebanyak tiga kali. Spesifikasi *Hardware* yang digunakan adalah Laptop HP Probook 4420s Intel Core i3 M350 @2.27GHz RAM 6 GB dan *Operating System* Windows 7 SP1 64bit.



Gambar 2 Grafik Waktu Komputasi

Berdasarkan gambar 4.1, jumlah TPS yang penuh berpengaruh pada nilai permutasi yang terjadi dan berdampak pada waktu komputasi karena semakin banyak cost yang harus di hitung.

5. Kesimpulan dan Saran

5.1 Kesimpulan

1. Sistem yang dibuat dapat mengatur proses logistik pengangkutan sampah dengan menerima masukan daftar TPS yang penuh untuk dikunjungi lalu sistem membuat rute kunjungan TPS untuk ditampilkan menjadi panduan rute di aplikasi *mobile*.
2. Algoritma untuk menentukan rute dengan nilai waktu tempuh terendah telah dibuat dengan nilai total cyclomatic complexity 12 dan waktu komputasi tertinggi yang berhasil di uji adalah 3.1 Jam untuk 12 TPS.

5.2 Saran

1. Pengembangan algoritma untuk implementasi secara parallel.

6. DAFTAR PUSTAKA

- [1] I. Markov, S. Varone and M. Bierlaire, "Vehicle Routing for a Complex Waste Collection Problem," 2014.
- [2] V. N. Bhat, "A Model for the Optimal Allocation of Trucks for Solid Waste Management," 1996.
- [3] M. T. P. B. B. d. Aguiar, "Optimization Techniques for the Mixed Urban Rural Solid Waste Collection Problem," 2010.
- [4] D. B. Liesje, J. Beliën and J. V. Ackere, *Municipal Solid Waste Collection and Management Problems: A Literature Review*, pp. 1-4, 2013.
- [5] S. D. d. M. Chaerul, "EVALUASI SISTEM PENGANGKUTAN SAMPAH DI WILAYAH BANDUNG UTARA," 2010.

- [6] M. R and R. G, "Exact and Heuristic Methods in Combinatorial Optimization," *The Linear Ordering Problem*, p. 17, 2011.
- [7] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," 2000.
- [8] Google, "Google Maps Directions API," Google, 16 November 2015. [Online]. Available: <https://developers.google.com/maps/documentation/directions/?hl=en>. [Accessed 1 December 2015].
- [9] G. B. D. a. J. H. Ramser, "The Truck Dispatching Problem," 1959.
- [10] S. R. a. P. Norvig, "Artificial Intelligence: A Modern Approach," 2010.
- [11] V. Hemmelmayr, K. F. Doerner, R. F. Hartl and S. Rath, "A heuristic solution method for *node* routing based," 2011.
- [12] V. Pavela and I. Nurhadi P, "PENYELESAIAN VEHICLE ROUTING PROBLEM DENGAN MENGGUNAKAN ALGORITMA NEAREST NEIGHBOR DAN TABU SEARCH," 2011.
- [13] J. C. S., "ANALISIS SISTEM PENGANGKUTAN SAMPAH KOTA MAKASSAR DENGAN METODE PENYELESAIAN VEHICLE ROUTING PROBLEM (VRP)," 2011.

