

OPTIMISASI JARINGAN DISTRIBUSI MIGAS MENGGUNAKAN *MULTIPLE VEHICLE ROUTING PROBLEM* TERINTEGRASI *GEOGRAPHICAL INFORMATION SYSTEM* (STUDI KASUS DI PT. X SURABAYA)

Rima Tri Wahyuningrum
rima_tw@yahoo.co.id
Jurusan Teknik Informatika
Universitas Trunojoyo

ABSTRAK

Permasalahan distribusi barang merupakan faktor yang sangat penting dalam peningkatan kualitas pelayanan dan pendapatan dari suatu perusahaan yang memiliki jaringan distribusi luas. Perusahaan harus menyediakan lebih dari satu kendaraan dalam armada pengiriman yang dimilikinya untuk melayani konsumen. Permintaan yang berubah setiap hari menimbulkan permasalahan baru yakni bagaimana menentukan rute untuk keseluruhan armada untuk dapat mengunjungi konsumen, dengan biaya pengiriman minimal setiap harinya. Metode penyelesaian yang digunakan adalah *ant colony algorithm*, suatu algoritma yang diilhami oleh perilaku kumpulan semut di dunia nyata. Hasil optimisasi, berupa urutan konsumen yang dilayani serta jalan-jalan yang harus dilalui, ditampilkan dalam suatu sistem informasi geografis.

Kata Kunci : optimisasi, jaringan distribusi, VRP, *ant colony algorithm*

1. PENDAHULUAN

Dalam sistem jaringan distribusi, dimungkinkan terdapat satu gudang induk produk atau bahan baku dan beberapa unit produksi yang terpisah satu sama lain. Masalah distribusi adalah bagian dari permasalahan logistik yang didefinisikan sebagai penyediaan barang dan jasa dari titik persediaan (*resource*) ke titik permintaan (*destination*). Kemampuan dalam jaringan distribusi untuk mengirimkan produk dari titik persediaan kepada titik permintaan atau pelanggan secara tepat waktu, jumlah yang sesuai dan ongkos yang murah akan menentukan keunggulan kompetitif di pasar. Permasalahan optimisasi biaya, rute, dan waktu distribusi produk dari titik persediaan (unit produksi, gudang induk, depot) ke titik permintaan (pelanggan, retailer) termasuk dalam kelas permasalahan yang di sebut *Vehicle Routing Problem (VRP)*. *VRP* terkait dengan permasalahan bagaimana mendatangi pelanggan dengan kendaraan dengan mempertimbangkan urutan pelanggan yang akan didatangi, dimana kendaraan berawal dan berakhir ke titik persediaan. *VRP* menjadi *Travelling Salesman Problem (TSP)* pada saat hanya terdapat satu alat angkut yang kapasitasnya tidak terhingga (Yudhistira, et. al., 2005).

Terkait dengan pemecahan masalah *VRP* telah banyak algoritma heuristik yang dikembangkan untuk menyelesaikan permasalahan *VRP* tradisional dan varian-varianannya. Yudhistira, et. al. (2005) mengembangkan algoritma heuristik penjadwalan alat angkut untuk pendistribusian produk majemuk dengan sumber tunggal dan tujuan majemuk. Suprayogi dan Imawati (2005) mengembangkan algoritma *sequential insertion* dengan *forward* dan *backward pass* untuk memecahkan masalah *VRP* dengan *multiple trips* dan *time windows*. Polacek et.al., (2004), mengembangkan algoritma berdasarkan *variable neighborhood* untuk permasalahan distribusi multi depot *VRP* dengan *time windows*. Sementara itu Buriol et.al., (2004) telah mengembangkan algoritma *memetic* untuk pemecahan masalah *TSP* dengan matrik jarak yang tidak simetris. Alvarenga, et.al., (2003) mengusulkan pendekatan *robust heuristic* untuk permasalahan *Vehicle Routing Problem* dengan *Time Window (VPRTW)* menggunakan *genetic algorithm (GA)* yang efisien dengan formulasi *mix integer programming (MIP)*, dengan hasil yang selalu kurang dari 1% terhadap solusi optimum (dibandingkan metode *exact*) (Sutapa, et al., 2003). Algoritma yang dikembangkan untuk mendapatkan solusi optimal dalam penelitian-penelitian tersebut belum dintegrasikan dengan perangkat lunak *Geographical Information System (GIS)* yang membantu memberikan informasi geografis rute pengiriman barang.

Pada kasus di PT. X Surabaya, jumlah barang yang dipesan dan frekuensi pemesanan barang oleh pelanggan selalu berubah setiap harinya (bersifat stokastik). Untuk efisiensi pelayanan, harus disediakan armada pengirim dengan jumlah kendaraan lebih dari satu. Permasalahannya adalah menentukan *route* yang harus ditempuh oleh tiap-tiap armada pengirim agar didapatkan jarak terpendek secara keseluruhan untuk melayani semua permintaan setiap hari. Penentuan *route* terpendek ini akan menghasilkan waktu pelayanan akan menjadi lebih singkat, dengan anggapan bahwa jarak tempuh sebanding dengan waktu yang diperlukan. Sehingga pengiriman dapat dilakukan lebih cepat, dan akan meminimalkan biaya distribusi secara signifikan. Tujuan penelitian ini adalah untuk merancang rute pengiriman barang masing-masing armada yang dapat meminimalkan jarak total pengiriman secara keseluruhan dalam frekuensi harian. Penentuan rute tidak hanya menentukan urutan konsumen yang harus dilayani tetapi menentukan urutan jalan yang harus dilalui. Kemudian dibuat perangkat lunak yang dapat memberikan informasi geografis rute pengiriman barang.

VRP termasuk kategori permasalahan *Non-Polynomial Hard (NP Hard)* karena merupakan gabungan masalah *Travelling Salesmen Problem* dan *Assignment Problem* yang keduanya juga merupakan *NP Hard*. Sehingga metode optimal hanya bisa berlaku untuk kasus-kasus yang sederhana, untuk kasus-kasus besar dan kompleks banyak dikembangkan metode heuristik. Pemecahan menggunakan metode heuristik ini lebih efektif jika menggunakan bantuan alat bantu komputasi (*computation software*) karena waktu perhitungannya membutuhkan waktu yang lama dan pemecahannya sulit. Dengan memperhatikan permasalahan *VRP* yang ada di PT X Surabaya, maka untuk mendapatkan variabel keputusan yang mendekati optimal digunakan algoritma meta heuristik *ant colony algorithm* yang diimplementasikan dalam *software Borland Delphi 7* yang diintegrasikan dengan *Geographical Information Systems (GIS): MapInfo Profesional 7* untuk mendapatkan *database* jaringan jalan dan menampilkan rute yang harus ditempuh pada peta serta *MapInfo X* sebagai komponen yang digunakan untuk koneksi *MapInfo* dengan *Borland Delphi*. Hasil kolaborasinya untuk *software* aplikasi optimisasi jaringan distribusi barang berbasis *GIS*. Data yang merepresentasikan peta dunia nyata disimpan dalam bentuk grafis dan tabel yang terhubung satu sama lainnya. Tiap obyek pada peta dan atribut dari masing-masing obyek tersebut disimpan dalam *database*. Penggunaan Sistem Informasi Geografis (*GIS*) dalam permasalahan perencanaan rute kendaraan dapat memberikan penyelesaian yang lebih baik dan mudah dipahami.

2. MODEL

2.1 Model Optimisasi *VRP*

Penyelesaian permasalahan *route* dan penjadwalan sangat bergantung pada jumlah permintaan tiap *node*, lokasi dan jarak antar *node*, serta kapasitas armada. Untuk permasalahan ini model yang digunakan adalah *Vehicle Routing Problem (VRP)*. *VRP* dapat didefinisikan sebagai penentuan sejumlah *tour* yang masing-masing berisi sejumlah *node*. *VRP* dengan jumlah kendaraan lebih dari satu disebut *Multiple-VRP*. Model dasar *VRP* dapat digambarkan sebagai berikut [Kallehauge (2001) dalam Sutapa, et al., (2003)] :

$$\text{Minimize } \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^K d_{ij} \cdot X_{ijk} \quad (1)$$

$$\text{Subject to: } \sum_{i=0}^N X_{ijk} - \sum_{i=0}^N X_{jik} = 0 \quad (2)$$

Setiap *node* *i* harus dikunjungi satu kali oleh salah satu kendaraan *k* yang berangkat dari *node* 0 (*depot*) serta kembali ke *node* 0.

$$\sum_{i=1}^N q_i \cdot Y_{ik} \leq Q_k \quad (3)$$

Jumlah demand tiap *node* *i* dalam setiap *tour* tidak boleh melebihi kapasitas kendaraan *k* yang melalui *tour* tersebut.

Dengan :

d_{ij} = jarak perpindahan dari *node i* ke *node j*
 $X_{ijk} = \begin{cases} 1, & \text{Jika kendaraan } k \text{ mengunjungi } j \text{ setelah } i \\ 0, & \text{Sebaliknya} \end{cases}$

$Y_{ik} = \begin{cases} 1, & \text{Jika } node\ j \text{ dikunjungi oleh kendaraan } k \\ 0, & \text{Sebaliknya} \end{cases}$

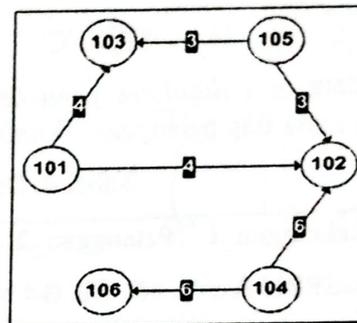
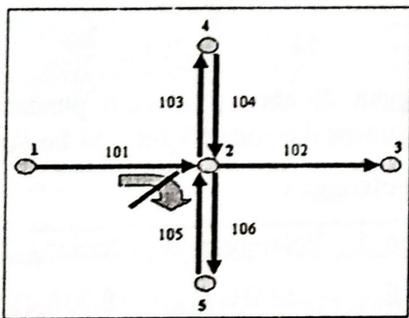
N = jumlah *node* yang harus dikunjungi

q_i = demand pada *node i*

Q_k = kapasitas kendaraan k

2.2. Pemodelan Jaringan Jalan

Penghitungan jarak terpendek antara satu *node* ke semua *node* lainya dapat dimodelkan sebagai suatu permasalahan lintasan terpendek. Jalan yang terbentuk di dunia nyata dimodelkan dalam bentuk *graph*, di mana tiap-tiap ruas jalan direpresentasikan dalam bentuk *arc*. Sedangkan pertemuan dua ruas jalan direpresentasikan oleh sebuah *node*. [Jiang, J., Han, G., Chen, J., (2002)]



Gambar 1. Jaringan jalan dengan aturan belokan. Gambar 2. Representasi jaringan jalan dalam

Setelah didapatkan pemodelan jaringan jalan dalam bentuk *graph*, maka penyelesaian masalah lintasan terpendek dapat diselesaikan dengan algoritma *Dijkstra*. Permasalahan yang dihadapi dalam menerapkan algoritma yang ada pada jaringan jalan adalah adanya aturan belokan, *turn restriction*, pada *node* persimpangan. Sehingga gambar 4 dan tabel 1 diformulasikan sebagai tabel 2.

Tabel 1. Atribut ruas jalan

Kode_ruas	node_asal	node_tujuan	Panjang
101	1	2	4
102	2	3	5
103	2	4	6
104	4	2	6
105	5	2	3
106	2	2	3

Tabel 2. Atribut aturan ruas jalan

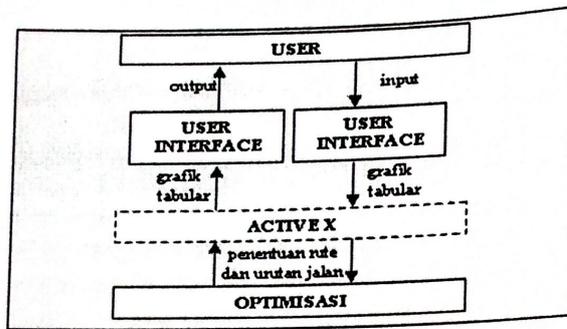
Kode_ruas	next_ruas
101	102
101	103
104	102
104	106
105	102
105	103

3. PERANCANGAN PERANGKAT LUNAK

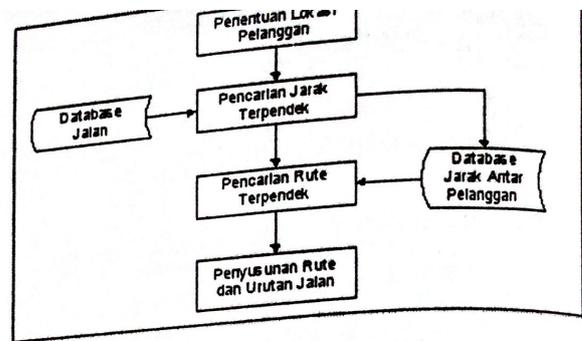
Rancangan perangkat lunak terdiri dari beberapa bagian, yakni pembuatan *database* jaringan jalan yang direpresentasikan dalam perangkat lunak *user interface*, serta pengolahan dalam perangkat lunak optimisasi, seperti diilustrasikan gambar 3.

3.1 Arsitektur sistem dari perangkat lunak yang dirancang dalam gambar 4.

3.2 Dalam pencarian terpendek antar semua konsumen, terlebih dahulu ditentukan di ruas mana konsumen berada.



Gambar 3. Skema perangkat lunak.



Gambar 4. Arsitektur Sistem

Tabel 3. Database pelanggan.

nomor	Nama	Kode_jalan	nomor	kode_ruas	jarak
1	SPBU A	1	4	154	0.321
2	SPBU B	169	15	10	0.264
3	SPBU C	23	6	53	0.703

Dengan data dari *database* jalan dan *database* pelanggan di atas, dilakukan pencarian jarak terpendek antar tiap pelanggan. *Database* jarak antar konsumen dari tabel 3 sebagai berikut :

Tabel 4. Database jarak antar pelanggan.

Pelanggan_1	Pelanggan_2	Jarak	Pelanggan_1	Pelanggan_2	Jarak
SPBU A	SPBU B	12.531	SPBU B	SPBU C	8.235
SPBU A	SPBU C	25.102	SPBU C	SPBU A	21.681
SPBU B	SPBU A	15.461	SPBU C	SPBU B	8.235

3.3 Diagram alir perangkat lunak. Program dibagi menjadi beberapa sub program berikut:

1. *Main*, merupakan sub program utama yang mengendalikan sub-sub program lainnya.
2. *Optimize*, sub program yang bertugas menjalankan metode tetangga terdekat atau koloni semut.
3. *TraceStart*, sub program ini bertujuan mendapatkan jarak antara *node* tertentu dengan semua ruas jalan yang ada.
4. *TraceFinish*, sub program yang akan melakukan penelusuran rute terpendek dari *node* asal ke *node* tujuan.
5. *EachTour*, sub program yang berfungsi untuk mengatur optimisasi setiap *tour* yang ada menggunakan metode koloni semut.
6. *Insertion*, sub program yang melakukan optimisasi dengan metode sisipan *node*

3.3.1 Memasukkan Data

Data yang dimasukkan meliputi nama pelanggan yang memesan dan jumlah pemesanan masing-masing pelanggan pada hari itu.

3.3.2 Mendapatkan Rute

Dari data pemesanan yang masuk, rute akan dicari oleh sub program *Optimize*. Pencarian ini kemudian dilanjutkan dengan koloni semut untuk VRP guna menemukan pengelompokan konsumen, sisipan *node* oleh sub program *Insertion*.

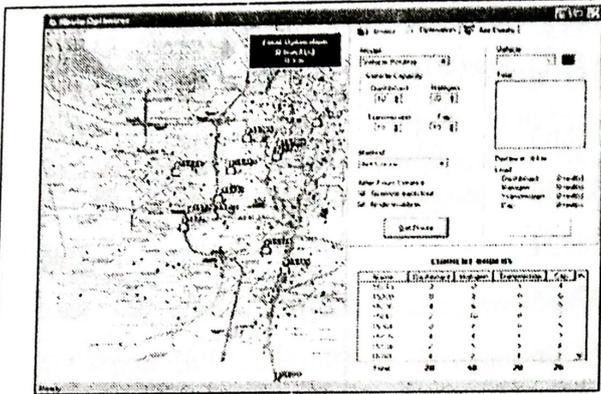
3.3.3 Mendapatkan Lintasan Terpendek

Setelah *route* terbentuk, berikutnya adalah mendapatkan lintasan terpendek oleh sub program *TraceStart* dan *TraceFinish*. Hasil yang didapatkan berupa urutan konsumen dan jalan yang harus dilalui.

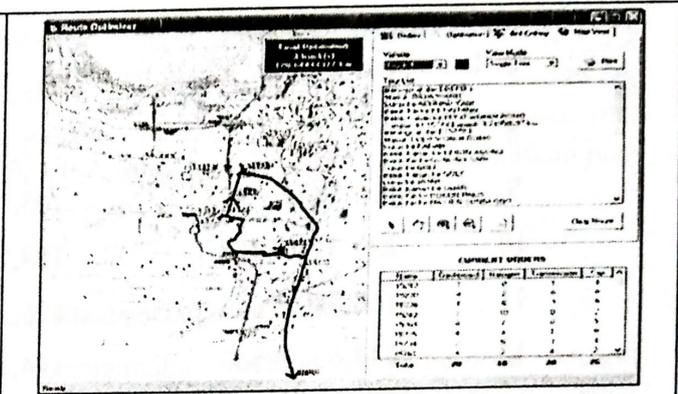
4. PENGUJIAN DAN CONTOH NUMERIK

Pengujian perangkat lunak dilakukan dengan beberapa tahap sebagai berikut :

1. Memasukkan Data Pemesanan. Pertama kali yang harus dilakukan adalah memasukkan data pemesanan meliputi nama pelanggan serta jumlah masing-masing barang yang dipesan.
2. Optimisasi dilakukan dengan model, metode, serta kapasitas yang dipilih, untuk mendapatkan urutan konsumen, jarak tempuh, serta muatan masing-masing armada. Ditunjukkan oleh gambar 5



Gambar 5. Halaman Optimisasi.



Gambar 6. Halaman Map View mode Single Tour

3. Pencarian Parameter Koloni Semut Terbaik. Jika dipilih metode koloni semut untuk VRP, maka akan dilakukan pencarian parameter terbaik. Waktu yang dibutuhkan bergantung pada kapabilitas PC, jumlah pelanggan dan jumlah serta pemesanan masing-masingnya.
4. Pemetaan Rute. Halaman *Map View* menampilkan hasil optimisasi dalam bentuk grafis. Ditunjukkan oleh gambar 5 dan 6
5. Pengujian Sistem. Sistem yang dirancang dalam penelitian ini telah diterapkan untuk data pemesanan migas di PT. X Surabaya. Dari data pemesanan acak didapatkan jarak rata-rata sebesar 164,34312372 km dan varians sebesar 40,35555456 km. Apabila diinginkan tingkat ketelitian taksiran μ sebesar 0,05 dengan selang kepercayaan 90% maka jumlah data yang dibutuhkan adalah :

$$n = \left(\frac{Z_{\alpha/2} \cdot \sigma}{e} \right)^2 = \left(\frac{1,96 \times 40,35555456}{0,1 \times 164,34312372} \right)^2 = 23,16 \approx 23 \quad (18)$$

dengan : $(1-\alpha) \cdot 100\%$ = Selang kepercayaan

e = Tingkat ketelitian taksiran μ

σ = Varians populasi

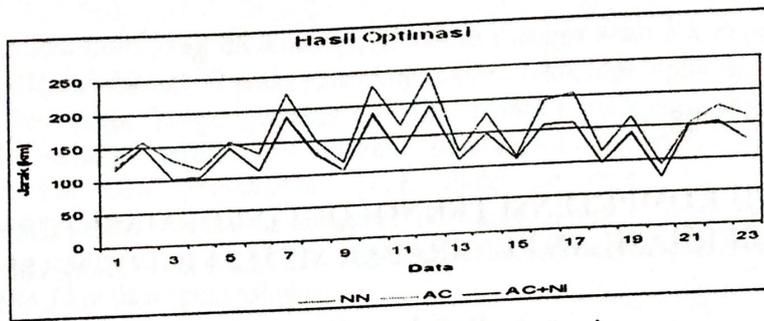
Sehingga jumlah data yang dibutuhkan untuk pengujian sistem adalah sebanyak 23 data rute pengiriman part. Hasil dari pengujian yang dilakukan pada 23 data pemesanan, dapat dilihat pada tabel 5.

Tabel 5. Hasil pengujian pada 23 data pemesanan

No.	Jarak Tempuh dengan metode Tetangga Terdekat (km)	Hasil Optimasi (km)	
		Koloni Semut	Koloni Semut dan Sisipan <i>Node</i>
1	133,55590126	122,39735093	118,14951166
2	158,06246004	151,28963654	151,28963654
3	130,04085341	101,48661641	101,48661641
4	115,25241889	103,29422870	103,28132539
5	154,65346859	145,35311621	145,17616949
6	136,26975894	108,89576331	108,89576331
7	223,59365303	188,72771178	187,03007169
8	151,93621591	132,87768863	129,99274649
9	116,80042689	105,08702656	104,77752092
10	229,04303458	188,28028521	185,16261124
11	168,78927640	125,17497865	125,17497865
12	243,63479563	194,48828706	194,48828706
13	123,33687403	111,32917287	111,32917287
14	179,12583383	149,49519727	149,10805912
15	112,46316296	109,44309825	109,44309825
16	194,17463948	158,36311057	158,28344963
17	204,00859137	159,53949647	159,53949647
18	115,68185950	96,77335753	96,77335753
19	165,38691023	142,44329811	140,79762603
20	91,16647344	73,62365209	73,62365209
21	152,07669004	149,05752343	149,05752343
22	178,07774157	122,39735093	118,14951166
23	162,10412673	151,28963654	151,28963654
Σ	4930,29371160	4217,70454639	4201,41244153
	Penghematan (%)	14,45	14,78

Dari hasil pengujian pada 23 data pemesanan dapat dilihat pada tabel 5 bahwa ketiga metode yang digunakan dapat memberikan perencanaan rute yang optimal. Jarak tempuh yang dihasilkan oleh algoritma koloni semut adalah sebesar 4217,70454639 km atau setara dengan 14,45% penghematan. Algoritma koloni semut yang diterapkan pada tiap *tour* disertai metode sisipan *node* menghasilkan jarak tempuh 4201,41244153 atau penghematan sebesar 14,78 %.

Hasil penggunaan teknik optimasi untuk penentuan rute kendaraan dapat dilihat pada gambar 7.



Gambar 7. Grafik Hasil Optimisasi

5. PENUTUP

5.1 Kesimpulan

Hasil penelitian ini memberikan beberapa kesimpulan sebagai berikut :

1. Penggunaan algoritma koloni semut untuk *vehicle routing problem* dengan parameter yang tepat akan memberikan hasil yang tidak dijamin optimal, tetapi mendekati optimal.
2. Sistem informasi geografis memberikan banyak keunggulan dalam perencanaan rute kendaraan dalam merepresentasikan rute yang harus dilalui, baik secara grafis melainkan juga dengan penjelasan dalam tabel.

5.2 Saran

Saran-saran yang dapat diajukan penelitian lanjutan adalah sebagai berikut:

1. Perlu dicari optimisasi jaringan distribusi menggunakan metode yang berbeda sebagai pembandingan algoritma *ant colony*, sekaligus saling melengkapi kelemahan masing-masing metode.
2. Jarak tempuh terpendek merupakan optimisasi yang sangat berpengaruh pada penghematan biaya, dan akan lebih aplikatif jika dikembangkan menjadi optimisasi dengan mempertimbangkan faktor multi *resource* (multi depot).

DAFTAR PUSTAKA

1. Buriol, L., Franca, P.M., Moscato, P., (2004), A New Memetic Algorithm for the Asymmetric Traveling Salesman Problem, *Journal of Heuristics*, 10: 483–506.
2. Gabrel, V., Knippel, A., Minoux, M., (2003), A Comparison Heuristics for the Discrete Cost Multicommodity Network Optimization Problem, *Journal of Heuristics*, 9: 429–445.
3. Jiang, J., Han, G., Chen, J., (2002) Modelling Turning Restrictions In Traffic Network For Vehicle, <http://www.isprs.org/commission4/proceedings/pdfpapers/410.pdf>
4. Melechovsky, J., Prins, C., Calvo, R.W., (2005), A Metaheuristic to Solve a Location Routing Problem with Non-Linear Costs, *Journal of Heuristics*, 11: 375–391
5. Polacek, M., Hartl, R.F., Doerner, K., Reimann, M., (2004), A Variable Neighborhood Search for the Multi Depot Vehicle Routing Problem with Time Windows, *Journal of Heuristics*, 10: 613–627.
6. Suprayogi, Imawati, D., (2005) Algoritma *Sequential Insertion* dengan *Forward* dan *Backward Pass* untuk Memecahkan Masalah *Vehicle Routing Problem* dengan *Multiple Trips* dan *Time Windows*, *Teknik dan Manajemen Industri* 25 (1), 41-54.
7. Sutapa, I.N., Widyadana, I.G.A., Christine, (2003), Studi Tentang *Travelling Salesmen* dan *Vehicle Routing Problem* dengan *Time Windows*, *Teknik Industri*, 5(2), 81-89
8. Yudhistira, Suprayogi, Halim, A.H. (2005), Algoritma Heuristik Penjadwalan dan Alat Angkut untuk Pendistribusian Produk Majemuk dengan Sumber Tunggal dan Destinasi Majemuk, *Teknik dan Manajemen Industri*, 25 (1), 19-31.