

Algoritma *Sequential Insertion* untuk Memecahkan *Vehicle Routing Problem* dengan *Multiple Trips, Time Window* dan *Simultaneous Pickup Delivery*

Suprayogi, Yusuf Priyandari*

Jurusan Teknik Industri, Universitas Sebelas Maret Surakarta

Abstrak

Penelitian ini mengembangkan modifikasi algoritma *sequential insertion* Suprayogi dan Imawati (2005) untuk dapat membentuk solusi layak bagi *vehicle routing problem* dengan *multiple trips, time windows*, dan *simultaneous pickup-delivery* (VRP-MTTWPD). Algoritma yang dikembangkan ini lebih luas kemampuannya karena juga dapat membentuk solusi model VRPMTTW. Algoritma diuji pada sejumlah contoh data hipotetik dan diperoleh informasi bahwa kualitas solusi dipengaruhi aturan yang digunakan untuk menentukan *seed customer*.

Kata kunci: *sequential insertion, vehicle routing problem, multiple trips, time windows, simultaneous pickup-delivery.*

1. Pendahuluan

Biaya transportasi merupakan salah satu komponen utama dalam struktur biaya logistik total. Stock dan Lambert (2001) menyatakan bahwa 60% dari biaya logistik total adalah biaya transportasi. Penciptaan efisiensi transportasi dapat memberikan kontribusi pada penurunan biaya logistik total. Salah satu permasalahan perencanaan transportasi adalah penentuan rute dan jadwal kendaraan yang secara umum dikenal dengan istilah masalah penentuan rute kendaraan (*vehicle routing problem, VRP*). VRP secara umum didefinisikan sebagai masalah penentuan rute bagi sejumlah kendaraan yang bertujuan meminimisasi ongkos total dan memenuhi sejumlah batasan yang mencerminkan karakteristik dari situasi nyata (Gendreau *et.al*, 1997).

Dua algoritma utama untuk pembentukan solusi VRP adalah menggabungkan rute yang ada dengan menggunakan kriteria penghematan (*savings criterion*) dan mencoba secara berurutan memasukkan pelanggan dalam rute kendaraan dengan menggunakan kriteria biaya penyisipan (*cost insertion*) (Laporte *et.al*, 2000). Algoritma yang kedua telah populer digunakan untuk menyelesaikan permasalahan rute dan penjadwalan kendaraan (*vehicle routing and scheduling problems*) (Campbell dan Savelsbergh, 2002).

Algoritma penyisipan (*insertion*) sangat terkenal sebab algoritma ini sangat cepat dalam memberikan solusi, mudah untuk diimplementasikan dan mudah dikembangkan untuk menangani pembatas-pembatas yang sulit. Beberapa literatur yang mengembangkan algoritma ini adalah Solomon (1987) untuk memecahkan VRP *with time window* (VRPTW), Vigo (1996) untuk permasalahan *asymmetric capacitated vehicle routing* dan Suprayogi dan Imawati (2005) untuk memecahkan VRP *with multiple trips and time window* (VRPMTTW). Algoritma *insertion* juga sering digunakan sebagai pembentuk solusi awal (Imawati, 2004). Solusi awal

* Correspondence: priyandari@uns.ac.id

tersebut kemudian diperbaiki oleh teknik perbaikan solusi yang lain misalnya algoritma *local search*, *simulated annealing* atau *tabu search*.

Suprayogi dan Imawati (2005) mengembangkan algoritma *sequential insertion* untuk membentuk solusi model VRPMTTW. Algoritma tersebut belum mengakomodasi pembentukan solusi model VRP-MTTWPD (VRP dengan *multiple trips*, *time windows*, dan *simultaneous pickup-delivery*) yang memiliki karakteristik proses *pickup-delivery* secara simultan. Penggunaan algoritma tersebut pada model VRP-MTTWPD dapat memunculkan solusi tidak layak akibat muatan kendaraan yang melebihi kapasitas. Algoritma tersebut hanya melakukan pemeriksaan total muatan kendaraan untuk setiap rute, padahal ada kemungkinan jumlah *pickup* lebih besar dari *delivery* di suatu titik pelanggan yang memunculkan terjadinya kelebihan muatan.

Berdasarkan uraian di atas, tulisan ini akan mengembangkan algoritma *sequential insertion* untuk model VRP-MTTWPD yang membentuk solusi layak. Tulisan ini dibagi dalam beberapa bagian yakni pembahasan model VRP-MTTWPD, algoritma *insertion*, pengembangan algoritma *sequential insertion* untuk VRP-MTTWPD, pengujian algoritma, dan terakhir penarikan kesimpulan.

2. Model VRP-MTTWPD

Model VRP-MTTWPD (VRP dengan *multiple trips*, *time windows*, dan *pick-up-delivery*) merujuk pada model VRP-MTTWPD Mahaputra (2006). Karakteristik model ini adalah pelanggan mempunyai *time window*, satu kendaraan bisa memiliki lebih dari satu rute selama horison perencanaan, dan pelanggan menerima suplai barang dari depot serta mengirim barang ke depot. Solusi akhir model VRP-MTTWPD adalah mendapatkan jumlah tur atau kendaraan yang digunakan untuk memenuhi seluruh permintaan pengantaran (*delivery*) dan pengambilan (*pickup*) pelanggan, urutan kunjungan ke pelanggan untuk tiap kendaraan, dan jadwal (saat keberangkatan, saat kedatangan, dan saat pelayanan) di tiap pelanggan dan depot.

Solusi VRP-MTTWPD harus memenuhi sejumlah tujuan yakni meminimasi jumlah tur (NV), waktu durasi tur total (TDT), dan rentang antara waktu durasi tur maksimum dengan waktu durasi tur minimum (RDT) (Mahaputra, 2006). Tujuan pertama dan kedua umum dijumpai dalam VRP dan keduanya merepresentasikan minimisasi biaya operasional. Tujuan ketiga merepresentasikan penciptaan keseimbangan waktu durasi antartur. Penciptaan keseimbangan waktu durasi antartur juga dapat dirumuskan dengan minimisasi waktu durasi tur maksimum (MDT).

Model VRP-MTTWPD dapat dirumuskan secara matematik berbasis *set partitioning problem* (SPP). Rumusan ini dikembangkan dari rumusan matematik model VRPMT Olivera dan Viera (2004). Solusi model VRP-MTTWPD adalah himpunan kandidat tur layak yang mencakup tur dengan rute tunggal maupun majemuk yang dinyatakan oleh S . Himpunan pelanggan dinyatakan oleh N . A_{it} adalah konstanta yang bernilai 1 jika tur t mencakup pelanggan i dan bernilai 0 jika tur t tidak mencakup pelanggan i . x_t merupakan variabel biner yang bernilai 1 jika tur t dipilih sebagai solusi optimal, dan bernilai 0 jika sebaliknya.

Fungsi tujuan model VRP-MTTWPD sebagai berikut:

$$\text{Min } Z = W_1 \cdot NV + W_2 \cdot TDT + W_3 \cdot RDT \quad (1)$$

Dengan pembatas:

$$\sum_t A_{it} x_t = 1, \quad \forall i \in N \quad (2)$$

$$\sum_t x_t = NV \quad (3)$$

$$\sum_t DT_t x_t = TDT \quad (4)$$

$$MDT = \max_{\{t, x_t=1\}} \{DT_t\} \quad (5)$$

$$x_t \in \{0,1\}, \forall t \in S \quad (6)$$

Fungsi tujuan (1) meminimasi tiga tujuan VRP-MTTWPD yang masing-masing tujuan diberi bobot W_1 , W_2 dan W_3 dengan $W_1 \gg W_2 \gg W_3$ untuk mencerminkan prioritas tujuan pertama lebih besar dari tujuan kedua dan ketiga, dan prioritas tujuan kedua lebih besar dari tujuan ketiga. Pembatas (2) menjamin bahwa tiap pelanggan hanya dilayani oleh satu kendaraan (tur). Pembatas (3), (4) dan (5) masing-masing berkaitan dengan pembatas untuk menentukan NV , TDT , dan MDT . Pembatas (6) merupakan pembatas variabel keputusan biner (0,1) untuk variabel keputusan x_t .

Rumusan di atas dipakai dengan terlebih dahulu membangkitkan semua kemungkinan kandidat tur layak (S) yang dibangkitkan dari tiap subset pelanggan N . Tiap tur dapat berupa tur dengan rute tunggal maupun rute majemuk. Kandidat tur layak adalah tur yang memenuhi persyaratan berikut:

- Pelanggan i yang terdapat dalam tiap tur t hanya dilayani sekali.
- Jumlah muatan kendaraan saat menuju atau meninggalkan pelanggan i tidak melebihi kapasitas kendaraan.
- Saat pelayanan untuk setiap pelanggan i tidak boleh melanggar batasan *time window* pelanggan maupun depot.

Karakteristik model VRP-MTTWPD disajikan di bawah ini.

2.1 Pelanggan, Depot dan Kendaraan

Model VRP-MTTWPD memiliki sejumlah n pelanggan yang direpresentasikan sebagai himpunan pelanggan $N = \{1, \dots, n\}$ dan sebuah depot yang direpresentasikan dengan angka 0 (nol). Himpunan titik yang menunjukkan depot dan semua pelanggan dalam model ini dinyatakan dengan $G = \{0, 1, \dots, n\}$. Setiap pelanggan dan depot memiliki waktu pelayanan bongkar muat yang diasumsikan homogen sebesar s . Waktu bongkar muat di depot diasumsikan terjadi setelah kendaraan menempuh sebuah rute. Jenis kendaraan diasumsikan homogen berkapasitas Q dengan jumlah tidak terbatas $NV = \{1, 2, \dots, \infty\}$. Karakteristik lain yang berkaitan dengan pelanggan, depot, dan kendaraan sebagai berikut:

$$q_i \leq Q \text{ dan } p_i \leq Q, \forall i \quad (7)$$

$$d_{ij} = d_{ji}, \quad \forall i \neq j \quad (8)$$

$$t_{ij} = d_{ij} / v \quad \forall i \neq j \quad (9)$$

Pertidaksamaan (7) menyatakan bahwa Permintaan pengantaran (q) dan permintaan pengambilan (p) di setiap pelanggan tidak melebihi kapasitas kendaraan (Q). Persamaan (8) menyatakan bahwa jarak antara pelanggan i dan pelanggan j adalah simetris. Persamaan (9) menyatakan lama waktu tempuh kendaraan dari satu pelanggan ke pelanggan lain dengan kecepatan kendaraan rata-rata v .

2.2 Rute dan Tur

Rute merepresentasikan urutan kunjungan sebuah kendaraan ke pelanggan-pelanggan, dimulai dari depot dan kembali ke depot. Sebuah rute terdiri dari satu depot dan setidaknya satu pelanggan. Rute didefinisikan sebagai himpunan vertek, $r = \{v_0, v_b, \dots, v_{np+1}\}$ dengan $v_0 = v_{np+1} = 0$ yang merepresentasikan depot, dan $(v_i, v_{i+1}) \in E \forall i \in [0, np]$ atau $v_i \in N \forall i \in [0, np]$ dengan np adalah jumlah pelanggan yang dikunjungi oleh rute r .

Tur adalah himpunan rute yang harus dilalui oleh kendaraan k . Sebuah tur didefinisikan sebagai himpunan rute, $t_k = \{r_{k,1}, r_{k,2}, \dots, r_{k,nr}\}$ dengan nr adalah banyaknya rute dalam tur t_k . Sebuah tur minimal memiliki sebuah rute. Solusi VRP-MTTWPD akan terdiri dari sejumlah tur dan didefinisikan sebagai $S = \{t_1, t_k, \dots, t_{nt}\}$ dengan nt adalah banyaknya tur.

$$M_{v_i v_j}^t = \sum_{m=1}^{np} q_{v_m} - \sum_{m=1}^i q_{v_m} + \sum_{m=1}^j p_{v_m} \quad \text{untuk } i \neq j, \text{ dan } i, j \neq 0 \quad (10)$$

$$M_{v_0 v_i}^t, M_{v_i v_j}^t, M_{v_i v_0}^t \leq Q \quad (11)$$

Persamaan (10) menyatakan total muatan kendaraan setelah melakukan pelayanan di pelanggan i menuju ke pelanggan j untuk setiap rute yang terbentuk. Pertidaksamaan (11) menyatakan total muatan kendaraan tidak boleh melebihi kapasitas kendaraan Q baik saat berangkat, setelah melakukan pelayanan di pelanggan dan ketika kembali ke depot.

2.3 Time Window

Time window pelanggan direpresentasikan oleh $[e_i, l_i]$ dimana e_i adalah saat pelanggan ke- i siap menerima pelayanan (*ready time*) dan l_i adalah batas terakhir pelanggan ke- i dapat menerima pelayanan (*deadline*). Lebar *time window* pelanggan ($l_i - e_i$) harus lebih besar atau sama dengan s . *Time window* depot direpresentasikan oleh $[e_0, l_0]$ dengan e_0 saat awal kendaraan boleh berangkat dari depot dan l_0 adalah saat paling akhir pelayanan harus dilakukan di depot. *Time window* depot merupakan horizon perencanaan untuk operasi model VRP-MTTWPD.

Kendaraan boleh datang ke pelanggan i sebelum e_i , namun tidak bisa melakukan pelayanan sebelum e_i sehingga terjadi waktu menunggu sebesar wt . Proses pelayanan di pelanggan i tidak boleh lebih dari l_i sehingga kendaraan harus datang ke pelanggan i tepat pada waktu atau sebelum $l_i - s$.

2.4 Waktu durasi tur

Waktu durasi tur adalah jangka waktu beroperasinya kendaraan saat berangkat dari depot menuju pelanggan pertama dalam rute pertama sampai kendaraan kembali ke depot untuk bongkar muatan bagi rute terakhir. Waktu durasi setiap tur t_k (DT_k) mencakup waktu perjalanan, waktu tunggu (jika ada), dan total waktu pelayanan pada tur tersebut. Waktu keberangkatan dari depot tidak harus selalu saat $e_0 = 0$, tetapi disesuaikan dengan metode *backward* (De Jong *et.al*, 1996, dalam Imawati, 2004) untuk mendapatkan waktu durasi yang minimum. Total waktu durasi tur (TDT) merupakan penjumlahan waktu durasi seluruh tur. Waktu durasi tur maksimum (MDT) adalah nilai waktu durasi terpanjang dari sebuah tur yang terbentuk dalam solusi. Adanya waktu durasi tur yang terlalu panjang merepresentasikan adanya ketidakseimbangan waktu operasi setiap kendaraan.

3. Algoritma Insertion

Algoritma *insertion* ada dua macam yaitu algoritma *parallel insertion* dan *sequential insertion*. Algoritma *parallel insertion* pernah dibahas oleh Potvin dan Rousseau (1993) untuk membentuk solusi VRPTW. Prinsip dasar dari algoritma *parallel*

insertion adalah pertama-tama membentuk sejumlah rute. Selanjutnya, tiap pelanggan akan disisipkan pada posisi tertentu pada salah satu rute yang memberikan kriteria terbaik. Jika tidak dimungkinkan lagi terjadi penyisipan maka satu tambahan rute dibangkitkan. Algoritma berhenti jika semua pelanggan telah ditugaskan.

Algoritma *sequential insertion* pernah dibahas oleh Suprayogi dan Imawati (2005) untuk membentuk solusi VRPMTTW. Algoritma ini diawali dengan memilih sebuah pelanggan untuk menduduki posisi sebagai pelanggan awal (*seed customer*) pada rute dan tur pertama. Selanjutnya, tiap pelanggan akan dicoba disisipkan pada sebuah busur (*edge*) tertentu yang memberikan kriteria terbaik bagi rute saat ini, batasan *time window* terpenuhi, dan total muatan kendaraan bagi rute saat ini tidak melebihi kapasitas kendaraan. Jika tidak ada pelanggan yang memenuhi kelayakan kapasitas kendaraan maka algoritma akan mencoba untuk membuat rute baru pada tur saat ini. Jika pembentukan rute baru ini berhasil dan masih memenuhi kelayakan *time window* maka algoritma akan mencoba untuk menyisipkan kembali tiap pelanggan pada rute saat ini seperti proses sebelumnya. Proses di atas dilakukan berulang sampai pembentukan rute baru gagal karena tidak terpenuhinya batasan *time window* sehingga perlu dibentuk satu tur baru. Proses penyisipan pelanggan ke dalam tur baru ini seperti proses pada tur pertama. Prosedur ini terus dilakukan berulang hingga semua pelanggan telah ditugaskan.

Suprayogi dan Imawati (2005) menyatakan bahwa kualitas solusi algoritma ini dipengaruhi oleh kriteria pemilihan pelanggan yang akan diposisikan sebagai pelanggan awal (*seed customer*) dalam rute dan tur yang akan dibentuk. Ada beberapa kriteria pelanggan awal yang dapat digunakan yaitu *earliest deadline*, *earliest ready time*, *shortest time window*, *longest travel time*, dan *random*. Adapun kriteria pemilihan pelanggan awal (*seed customer*) yang dimasukkan dalam rute selanjutnya (kedua, ketiga, dan seterusnya) pada suatu tur dapat menggunakan waktu penyelesaian tur (*completion time*).

3.1 Pengembangan Algoritma Sequential Insertion untuk VRP-MTTWPD

Berikut ini adalah algoritma *sequential insertion* yang dikembangkan untuk model VRP-MTTWPD.

Langkah 1

Buat tur pertama $t=1$, dan rute pertama $r=1$ yang dimulai dari depot.

Langkah 2

Jika semua pelanggan sudah ditugaskan, maka lanjutkan langkah 9.

Apabila $r=1$, pilih satu pelanggan yang belum ditugaskan untuk menduduki posisi sebagai pelanggan awal (*seed customer*) pada rute dan tur saat ini dan lanjutkan langkah 4. Apabila $r \neq 1$, maka lanjutkan langkah 3.

Langkah 3

Cobalah bentuk rute baru pada rute saat ini menggunakan tiap pelanggan j yang belum ditugaskan. Pilihlah pelanggan j yang memenuhi *time window* dan memberikan waktu durasi tur minimum untuk ditugaskan sebagai pelanggan awal bagi rute baru dalam tur saat ini. Kemudian lanjutkan langkah 4.

Jika tidak ada pelanggan j yang memenuhi *time window*, lanjutkan langkah 8.

Langkah 4

Hitung total permintaan pengantaran (Σq) dan total permintaan pengambilan (Σp) rute saat ini untuk tiap penyisipan pelanggan j yang belum ditugaskan. Jika semua upaya penyisipan menyebabkan muatan kendaraan berlebih ($\Sigma q > Q$ atau $\Sigma p > Q$) lanjutkan langkah 7.

Langkah 5

Coba sisipkan satu-persatu semua pelanggan j (yang memenuhi langkah 4) diantara semua busur (k,l) dalam rute saat. Hitung muatan kendaraan di tiap titik pelanggan (M_{ij}) dan periksa kelayakan *time window* untuk penyisipan pelanggan pada busur (k,l) dalam rute saat ini. Untuk setiap penyisipan yang memenuhi batasan muatan kendaraan dan *time window*, hitung dan catat waktu durasi tur yang dibentuknya. Apabila ternyata tidak ada pelanggan j yang memenuhi batasan muatan kendaraan dan *time window*, maka lanjutkan langkah 7

Langkah 6

Pilih sebuah pelanggan j yang memberikan waktu durasi tur terkecil. Pilihan ini sekaligus menentukan posisi busur pelanggan j dalam rute dan tur saat ini. Tugaskan pelanggan j ini ke dalam rute dan tur saat ini dan perbarui informasi mengenai rute, tur dan pelanggan yang sudah ditugaskan. Ulangi langkah 4.

Langkah 7

Buat rute baru untuk tur saat ini, $r = r + 1$. kembali ke langkah 2.

Langkah 8

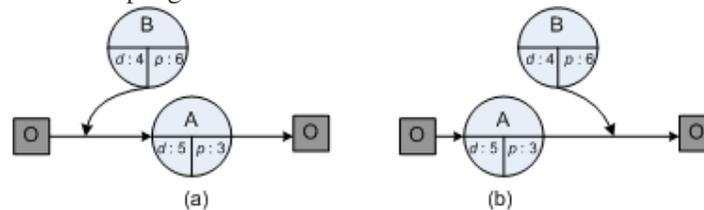
Buatlah tur baru, $t = t + 1$ dan rute saat ini $r = 1$ yang diawali dari depot.

Kembali ke langkah 2.

Langkah 9

Semua pelanggan telah ditugaskan dan hentikan prosedur ini.

Perbedaan utama algoritma Suprayogi dan Imawati (2005) dengan algoritma ini terletak pada langkah 5. Algoritma Suprayogi dan Imawati (2005) tidak menghitung muatan kendaraan di setiap titik pelanggan setelah dilakukan penyisipan pelanggan baru dalam suatu busur. Algoritma yang dikembangkan ini menghitung muatan kendaraan di setiap titik pelanggan setiap telah dilakukan penyisipan sebuah pelanggan baru di setiap posisi yang mungkin. Hal tersebut dilakukan karena perbedaan posisi penyisipan pelanggan baru ke dalam sebuah busur mempengaruhi muatan kendaraan.



Gambar 1. Pengaruh posisi pelanggan terhadap muatan kendaraan

Gambar 1 memberikan ilustrasi bahwa posisi penyisipan pelanggan ke dalam suatu busur mempengaruhi muatan kendaraan. Misalkan kapasitas kendaraan sebesar 10 unit. Apabila pelanggan B diposisikan pada busur O-A maka kendaraan akan mengalami kelebihan muatan setelah mengunjungi pelanggan B menuju pelanggan A. Muatan kendaraan berlebih karena terdapat 5 unit barang yang akan dikirim ke pelanggan A ditambah 6 unit barang yang diambil dari pelanggan B. Apabila pelanggan B diposisikan pada busur A-O, maka tidak terjadi kelebihan muatan.

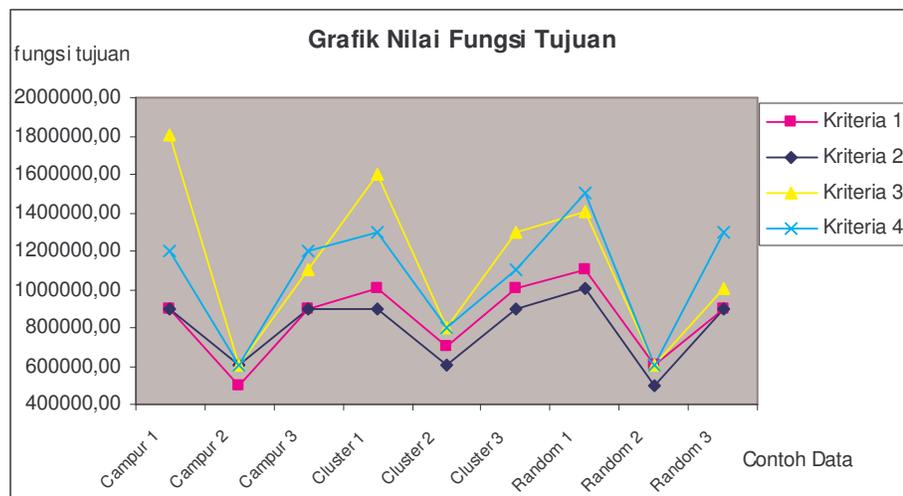
Algoritma Suprayogi dan Imawati (2005) tidak dapat dipakai untuk model VRP-MTTWPD, sebaliknya algoritma ini dapat dipakai untuk model VRPMTTW. Dengan demikian, algoritma ini lebih luas kemampuannya dibanding algoritma sebelumnya.

3.2 Pengujian Algoritma Sequential Insertion untuk VRP-MTTWPD

Algoritma *sequential insertion* untuk model VRP-MTTWPD dibuat ke dalam sebuah program dan diuji pada sembilan contoh data hipotetik dari Mahaputra (2006). Setiap contoh data terdiri dari 100 pelanggan dan memiliki karakteristik yang berbeda dalam sebaran lokasi pelanggan dan lebar *time window*.

Pemilihan pelanggan awal bagi rute pertama untuk setiap tur menggunakan kriteria *earliest deadline* (kriteria 1), *earliest ready time* (kriteria 2), *shortest time window* (kriteria 3), dan *longest travel time* (kriteria 4). Pemilihan pelanggan awal bagi rute kedua dan seterusnya untuk setiap tur menggunakan kriteria waktu penyelesaian tur (*completion time*) minimum. Bobot fungsi tujuan yang digunakan adalah $W1 = 100000$, $W2 = 0,4$ dan $W3 = 0,00005$.

Solusi algoritma *sequential insertion* disajikan dalam Lampiran Tabel 1. Gambar 2 menyajikan *plot* nilai fungsi tujuan untuk sembilan contoh data dengan empat kriteria pemilihan pelanggan awal. Kriteria 2 yakni *earliest ready time* relatif membentuk solusi rata-rata yang lebih baik dari kriteria lainnya. Karakteristik model VRP dan karakteristik data yang diuji akan mempengaruhi kualitas solusi untuk setiap pemilihan kriteria Imawati (2004).



Gambar 2. Grafik nilai fungsi tujuan untuk empat kriteria pada algoritma *sequential insertion* VRP-MTTWPD

Ditinjau dari lama waktu pembentukan solusi, rata-rata memerlukan waktu kurang dari 1 detik. Ini menunjukkan bahwa algoritma ini sangat cepat dalam membentuk solusi meskipun solusi yang dihasilkan tidak optimal. Algoritma *sequential insertion* ini biasanya digunakan sebagai pembentuk solusi awal. Solusi awal tersebut kemudian diperbaiki kualitasnya menggunakan algoritma perbaikan solusi lainnya.

4. Kesimpulan

Algoritma *sequential insertion* yang dikembangkan dalam kajian ini dapat membentuk solusi layak bagi model VRP-MTTWPD dan dapat mengakomodir pembentukan solusi model VRPMTTW. Kualitas solusi algoritma ini dipengaruhi oleh kriteria pemilihan *seed customer* bagi rute pertama. Algoritma ini sangat cepat dalam membentuk solusi model VRP-MTTWPD sehingga dapat dipakai sebagai pembentukan solusi awal untuk diperbaiki oleh algoritma lainnya.

Daftar Pustaka

Campbell, AM dan Savelsbergh, M (2002). *Efficient Insertion Heuristics for Vehicle Routing Problem and Scheduling Problem*, Georgia Institute of Technology.

Gendreau, M., Hertz, A., Laporte, G. (1992), New Insertion and Postoptimization Procedures for The Traveling Salesman Problem, *Operations Research*, vol. 40, No. 6, November-Desember, pp. 1085-1094.

Imawati, D. (2004), *Pemecahan Vehicle Routing Problem with Multiple Trips and Time Window dengan Menggunakan Pendekatan Local Search dan Simulated Annealing*, Tugas Akhir, Departemen Teknik dan Manajemen Industri Institut Teknologi, Bandung.

Laporte, G, Gendreau, M., Potvin, JY dan Semet, F. (2000), Classical and Modern Heuristics for The Vehicle Routing Problem, *International Transactions in Operational Research*, 7, pp. 285-300.

Mahaputra, MS. (2006), *Pemecahan Masalah Penentuan Rute Kendaraan yang Mempertimbangkan Multiple Trips, Time Windows, dan Simultaneous Pick-up and Delivery dengan Menggunakan Algoritma Genetik*, Tesis, Departemen Teknik dan Manajemen Industri Institut Teknologi, Bandung.

Olivera, A., Viera, O. (2004), *Adaptive memory programming for the vehicle routing problem with multiple trips*, Technical report to Elsevier Science. Instituto de Computacion, Facultad de Ingenieria, Universidad de la Republica, Herrera y Reissig 525, Montevideo, Uruguay.

Potvin, J.Y dan Rousseau, J.M (1993), A Parallel Route Building Algorithm for The Vehicle Routing and Scheduling Problem with Time Windows, *European Journal of Operations Research*, 66, pp. 331-340

Solomon, MM (1987), Algorithms for The Vehicle Routing and Scheduling Problems with Time Windows Constraint, *Operations Research*, 35, pp. 254-265

Stock, JR., dan Lambert, DM. 2001. *Strategic logistic management*. Edisi ke-4. New York. McGraw-Hill.

Suprayogi dan Imawati, D (2005) Algoritma *Sequential Insertion* dengan *Forward* dan *Backward Pass* untuk memecahkan *Vehicle Routing Problem* dengan *Multiple Trips* dan *Time Windows*, *Jurnal Teknik dan Manajemen Industri*, Vol. 25 Nomor 1 April 2005.

Vigo, D (1996), A Heuristic Algorithm for The Asymmetric Capacitated Vehicle Routing Problem, *European Journal of Operational Research*, 89, pp. 108-126