Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

TINJAUAN PUSTAKA

2.1 Vehicle Routing Problem

Vehicle Routing Problem pertama kali diformulasikan oleh Dantzig dan Ramser (1959) sebagai masalah utama bidang transportasi, distribusi, dan logistik. Menurut mereka VRP adalah masalah untuk merancang serangkaian rute kendaraan di mana armada kendaraan tetap melayani pengiriman dengan kapasitas seragam yang harus diketahui permintaan pelanggan untuk satu komoditas dari depot dengan biaya minimum.

Vehicle routing digambarkan dengan jaringan jalan, yang kemudian dituangkan dalam sebuah graf, baik graf berarah $G = (V, A)$, graf tidak *berarah* $G = (V, E)$, maupun graf campuran $G = (V, A \cup E)$. Penggunaan bentuk graf ini disesuaikan dengan daerah yang akan dikunjungi kendaraan pengangkut. Graf tak berarah digunakan jaringan jalan skala besar, meliputi negara dan negara bagian atau provinsi. Sedangkan graf berarah digunakan untuk jaringan jalan skala kecil, misal untuk menggambarkan jalan-jalan dalam satu kota.

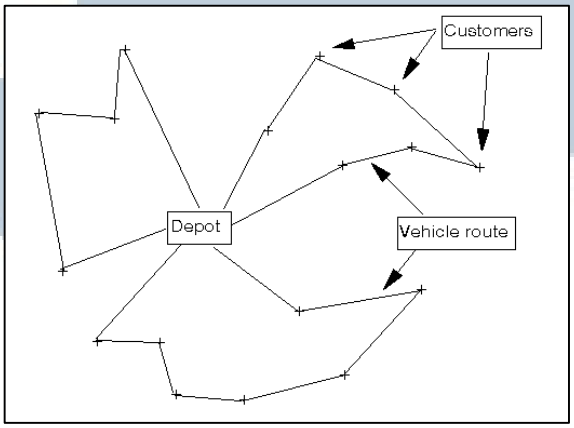
Verteks menggambarkan depot, pelanggan ataupun persimpangan jalan. Himpunan verteks dilambangkan dengan $V = (v_0, \ldots, v_n)$. Verteks v_0 mewakili depot, di mana terdapat kendaraan pengangkut identik sejumlah k dengan kapasitas Q. Sedangkan verteks lainnya melambangkan kota atau pelanggan, yang memiliki permintaan d_i. *Arc* atau *edge* menggambarkan jalan-jalan yang ada. *Edge* dapat bersifat berarah $(i,j) \in A$, di mana $A = \{(v_1,v_j): i \neq j, v_1,v_j \in V\}$ dan tidak berarah, $e \in E$. . Biaya dan jarak perjalanan dilambangkan oleh c_{ij}, yang

Implementasi Algoritma Cluster..., Iskandar Mirza, FTI UMN, 2012

`

didefinisikan pada *A*, sedangkan waktu non-negatif dilambangkan oleh t_{ij}, yang juga didefinisikan pada *A*.

Setiap verteks v_i dalam *V* diasosiasikan dengan sejumlah barang q_i . yang akan diantarkan oleh satu kendaraan. VRP bertujuan untuk menentukan sejumlah *k* rute kendaraan dengan total biaya yang minimum, bermula dan berakhir di sebuah depot. Adapun setiap verteks dalam *V* dikunjungi tepat sekali oleh satu kendaraan. Jadi, biaya dari solusi masalah ini S adalah : $F_{VRP}(S) = \sum C(R_i)$. (Cordeau dkk, 2002)



Gambar 2. 1 *Output* penyelesaian VRP (Dantzig & Ramser, 1959)

2.2 Kategori *Vehicle Routing Problem*

Pada perkembangannya, VRP memiliki beberapa karakteristik sehingga dapat dibagi-bagi dalam beberapa kategori masalah, seperti yang dapat ditunjukkan dalam tabel 2.1. Kategori ini dibuat oleh *Bodin dan Golden* (1981), yang memaparkan berbagai karakteristik umum, yang akan membedakan VRP. Keseluruhan tabel memberikan gambaran singkat tentang masalah *routing*.

`

Tabel 2. 1 Karakteristik VRP

Sumber : Bodin and Golden, 1981

NO.	KARAKTERISTIK	VARIAN YANG MUNGKIN
1.	Jumlah kendaraan pengangkut	Satu kendaraan pengangkut Multi kendaraan pengangkut
2.	Tipe kendaraan pengangkut	Homogen (satu tipe kendaraan pengangkut) Heterogen (multi tipe kendaraan pengangkut) Tipe khusus
3.	Tempat asal kendaraan (pusat)	Satu pusat Multi pusat
4.	Tipe permintaan	Permintaan deterministik (telah diketahui sebelumnya) Permintaan stokastik Permintaan kepuasan sebagian
5.	Lokasi permintaan	Simpul Garis Campuran

UNIVERSITAS MULTIMEDIA NUSANTARA

`

Table 2. 1 Karakteristik VRP (lanjutan)

NO.	KARAKTERISTIK	VARIAN YANG MUNGKIN
6.	Jaringan yang mendasari	Tidak berarah Berarah Campuran *Euclid*
7.	Batasan kapasitas kendaraan pengangkut	Semuanya sama Jalur yang berbeda Kapasitas tak terbatas
8.	Rute maksimum	Sama untuk semua rute Berbeda untuk setiap rute Tidak ditentukan
9.	Sistem pengoperasian	Pengambilan saja Pengiriman saja Pengambilan dan pengantaran
10.	Biaya	Pengambilan saja Pengiriman saja Pengambilan dan pengantaran
11.	Tujuan	Meminimalkan biaya total rute Meminimalkan jumlah kendaraan pengangkut yang diperlukan Meminimalkan fungsi kegunaan berdasarkan pada pelayanan dan kenyamanan

UNIVERSITAS MULTIMEDIA NUSANTARA

`

Table 2. 1 Karakteristik VRP (lanjutan)

NO.	KARAKTERISTIK	VARIAN YANG MUNGKIN
		Meminimalkan fungsi kegunaan berdasarkan pada prioritas pelanggan Meminimalkan fungsi kegunaan berdasarkan pada prioritas pelanggan

2.3 Vehicle Routing Problem with Time Windows

Menurut Kallehauge dkk, *Vehicle Routing Problem with Time Windows* (VRPTW) adalah perluasan dari VRP. Jika pada VRP ditambahkan *time window* pada masing–masing konsumen, maka permasalahan tersebut menjadi VRPTW. Untuk VRPTW selain adanya kendala kapasitas kendaraan, terdapat tambahan kendala yang mengharuskan kendaraan untuk melayani tiap konsumen pada *time frame* tertentu. Kendaraan boleh datang sebelum *time window* "*open*", tetapi konsumen tersebut tidak dapat dilayani sampai *time window "open".* Kendaraan tidak diperbolehkan untuk datang setelah *time window "closed"* (Kallehauge dkk, 2001).

Tangiah mendefinisikan VRPTW sebagai permasalahan untuk menjadwalkan sekumpulan kendaraan, dengan kapasitas dan *travel time* terbatas, dari *central depot* kesekumpulan konsumen yang tersebar secara geografis, dengan *demand* diketahui, dalam *time windows* tertentu (Tangiah, 1995). *Time windows* adalah *two sided*, yang berarti bahwa tiap konsumen harus dilayani saat atau setelah *earliest time*, dan sebelum *latest time* dari konsumen tersebut. Jika kendaraan datang ke konsumen sebelum *earliest time* dari konsumen tersebut,

`

maka akan menghasilkan *idle* atau waktu tunggu. Kendaraan yang datang ke konsumen setelah *latest time* adalah *tardy*. Terdapat pula waktu *service* yang diperlukan untuk melayani tiap konsumen. Biaya rute dari suatu kendaraan adalah total dari waktu *travel* (proposional dengan jarak), waktu tunggu, dan waktu *service*, yang diperlukan untuk mengunjungi sekumpulan konsumen.

Menurut Kyung, dkk, VRPTW terbagi menjadi dua kasus yaitu kasus *hard time windows* dan kasus *soft time windows*. Pada kasus *hard time windows*, pengiriman akan ditolak apabila tidak sesuai dengan waktu yang telah ditentukan oleh konsumen, sedangkan pada kasus *soft time windows* konsumen akan menerima pengiriman walaupun tidak sesuai dengan waktu yang telah ditentukan sekaligus memberikan penalti atau biaya tambahan atas keterlambatannya (Kyung dkk, 1988).

2.4 Teknik-Teknik Penyelesaian Vehicle Routing Problem

Pada dasarnya, terdapat 3 macam penyelesaian VRP:

- Metode eksak, yaitu pendekatan dengan menghitung setiap solusi yang mungkin sampai hasil terbaik diperoleh.
- Metaheuristik, adalah suatu metode untuk melakukan eksplorasi yang lebih dalam pada daerah yang menjanjikan dari ruang solusi yang ada.
- Heuristik, yaitu cara untuk menyelesaikan permasalahan dengan mengabaikan keakuratan solusi, tapi menghasilkan solusi yang lebih cepat daripada metode eksak dan metaheuristik.

VRP adalah sebuah problem pemrograman integer yang masuk kategori masalah NP-Hard (*Nondeterministic Polynomial time Hard*), yang berarti usaha

`

komputasi akan semakin sulit dan banyak seiring dengan meningkatnya ruang lingkup masalah. Pada VRP, hanya ruang lingkup masalah kecil dan sederhana yang dapat dicari solusi optimalnya. Hingga saat ini bila masalah VRP yang dihadapi relatif besar dan kompleks maka waktu yang dibutuhkan untuk mencari solusi optimal masalah tersebut relatif lama. Berdasarkan beberapa penelitian sebelumnya, nilai optimal dari fungsi objektif sulit untuk didapat dengan menggunakan *exact algorithm* (contohnya: *branch and Bound* dan *dynamic programming*). Pendekatan *exact algorithm* dinyatakan tidak cukup baik untuk masalah yang dihadapi (VRP yang besar dan kompleks), sehingga dikembangkan metode heuristik sebagai salah satu alternatif untuk menyelesaikan masalah tersebut (Cordeau dkk, 2002).

2.4.1. Heuristik untuk menyelesaikan Vehicle Routing Problem

Menurut Judea Pearl, metode heuristik adalah teknik yang dirancang untuk memecahkan masalah yang mengabaikan apakah solusi dapat dibuktikan benar, tapi yang biasanya menghasilkan solusi yang baik atau memecahkan masalah yang lebih sederhana yang mengandung atau memotong dengan pemecahan masalah yang lebih kompleks (Judea Pearl, 1984). Metode Heuristik ini bertujuan untuk mendapatkan performa komputasi yang cepat atau penyederhanaan konseptual, dengan mengorbankan tingkat keakuratan atau presisi.

Beberapa metode heuristik telah dikembangkan untuk menyelesaikan VRP. Secara umum, metode heuristik dibagi menjadi dua kelas yaitu heuristik klasik (*classical heuristic*) yang banyak dikembangkan antara tahun 1960 sampai

UNIVERSITAS MULTIMEDIA NUSANTARA

`

dengan tahun 1990 dan meta-heuristik, dimana meta-heuristik terkadang dipandang sebagai prosedur perbaikan yang sangat kompleks

Laporte dan Semet (1999) membagi heuristik klasik menjadi tiga:

- *constructive heuristics*
- *two-phase heuristics*
- *improvement heuristics*.

Two phase heuristics juga terbagi menjadi dua kategori yang disebut

- *cluster first, route second heuristic*
- *route first, cluster second heuristics.*

2.4.2. Two Phase Heuristic (Cluster First – Route Second)

Dalam menyelesaikan masalahnya metode ini dipecah menjadi dua komponen yaitu mengelompokkan permasalahan ke dalam rute yang layak (*clustering*) dan baru dilakukan pembuatan rute (*routing*), sehingga cara tersebut dapat dinyatakan dengan ; *cluster first – route second* dan *route first – cluster second*.

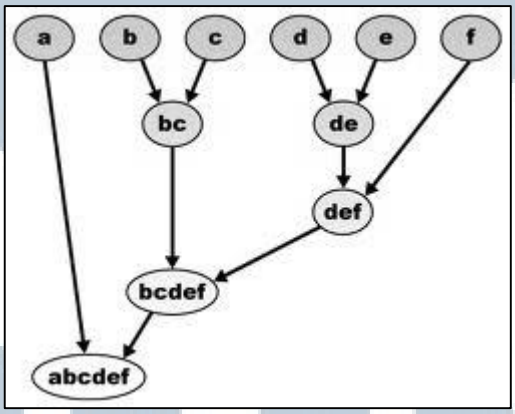
2.4.2.1. Clustering

Clustering merupakan suatu proses pengelompokan data ke dalam kelas-kelas atau *cluster-cluster* berdasarkan suatu kemiripan atribut – atribut diantara kelompok data tersebut. *Clustering* merupakan fungsionalitas pada *data mining*. Tujuan dari proses *clustering* yaitu untuk mengelompokkan data ke dalam suatu *cluster*, sehingga objek pada suatu *cluster* memiliki kemiripan yang besar dengan objek lain pada *cluster* yang sama, tetapi tidak mirip dengan objek pada *cluster* yang lain. Salah satu ciri *clustering* yang baik atau optimal adalah jika menghasilkan *cluster* yang berisi data dengan tingkat kemiripan (*similarity*) yang

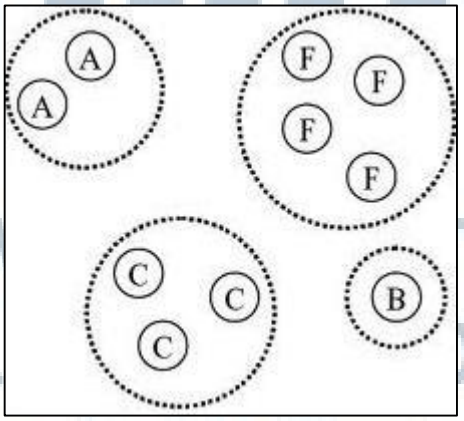
`

tinggi pada *cluster* yang sama dan tingkat kemiripan rendah pada *cluster* yang berbeda (McCallum dkk, 2000).

Dalam mengembangkan metode *clustering* terdapat beberapa pendekatan, diantaranya dengan pendekatan partisi dan pendekatan hirarki. *Clustering* dengan pendekatan hirarki, mengelompokkan data dengan membuat suatu hirarki berupa dendogram (*tree*) dimana data yang mirip akan ditempatkan pada hirarki yang berdekatan dan yang tidak pada hirarki yang berjauhan. *Clustering* dengan pendekatan partisi, mengelompokkan data dengan memilah-milah data yang dianalisa ke dalam *cluster-cluster* yang ada (McCallum dkk, 2000).



Gambar 2. 2 *Clustering* hirarki (McCallum dkk, 2000)



Gambar 2. 3 *Clustering* partisi (McCallum dkk, 2000)

`

Untuk melakukan *clustering* data bertipe numerik, digunakan pengukuran jarak berdasarkan konsep geometri (*geometric distance measure*) seperti *Euclidean distance.* Namun untuk atribut yang bersifat *categorical, geometric distance measure* tidak dapat diterapkan. (McCallum dkk, 2000).

2.4.2.2. Cluster Partisi yang Efisien Menggunakan Algoritma Kanopi

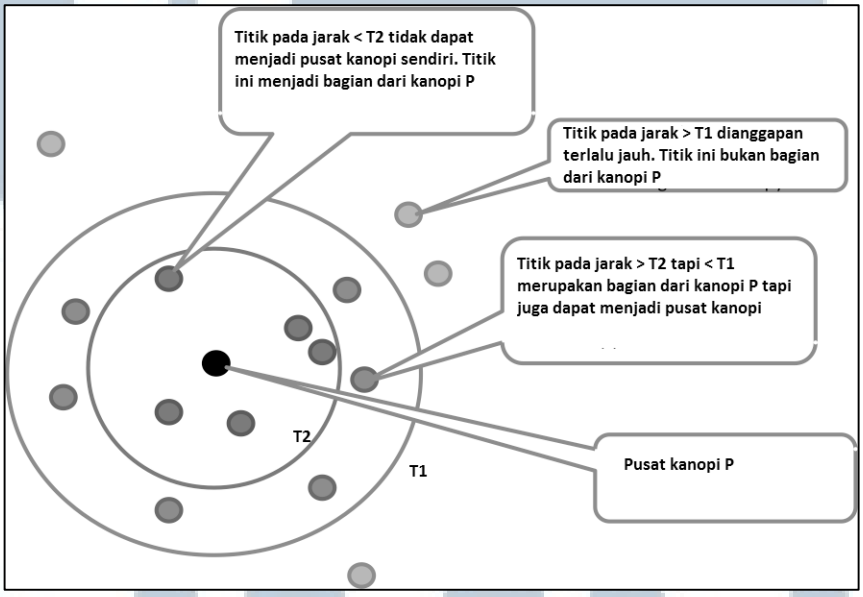
Teknik *clustering* dengan kanopi sangat sederhana, cepat, dan akurat untuk membagi objek ke dalam *cluster*. Setiap objek direpresentasikan sebagai titik dalam dua dimensi ataupun multidimensi. Tujuan utama dari kanopi adalah mengurangi jumlah perhitungan yang dibutuhkan dalam *clustering* dengan mempartisi data menjadi *subset* yang saling melengkapi, dengan hanya mengukur jarak antara sepasang titik data yang menjadi bagian dari salah satu *subset*.

Kanopi adalah sebuah *subset* dari elemen-elemen (data, titik, atau barang) yang berdasarkan perhitungan jarak, berada di dalam jarak batas dari sebuah titik pusat kanopi. Sebuah elemen dapat muncul pada lebih dari satu kanopi, dan setiap elemen harus muncul paling tidak dalam satu buah kanopi. Kanopi dibuat dengan tujuan sebuah titik tidak muncul dalam kanopi lain dan titik yang cukup jauh terpisah tidak mungkin dalam konteks yang sama. Karena ukuran jarak yang digunakan untuk membuat kanopi hanya perkiraan, mungkin tidak ada jaminan untuk properti ini, tetapi dengan memungkinkan kanopi tumpang tindih satu sama lain, dan dengan memilih batas jarak cukup besar, hasil yang diperoleh dapat terjamin dalam beberapa kasus.

Langkah-langkah dalam pembuatan *cluster* kanopi dengan *cheap distance metric* adalah:

`

1. Diketahui jarak batas T1 dan T2; T1 > T2 dan diketahui sekumpulan titik.
2. Tentukan Pusat kanopi: Iterasi melalui sekumpulan titik, jika titik adalah pada jarak Tentukan keanggotaan kanopi - untuk setiap titik pada *input* set, jika titik berada pada jarak < T1 dari titik manapun dalam daftar pusat kanopi (dihasilkan dalam Langkah 1) maka titik adalah anggota dari *cluster*.



Gambar 2. 4 Kanopi *clustering* (McCallum dkk, 2000)

2.4.2.3. Pembuatan Rute (Routing)

A. Holding List (Daftar Tunggu)

Pada algoritma untuk memecahkan VRPTW yang diperkenalkan oleh Lau dkk (2003), *holding list* adalah daftar berisi konsumen yang belum terlayani. Tujuan *holding list* pada masalah VRPTW adalah memberikan kemungkinan perluasan pada konsumen yang akan dilayani dan dicari rutenya selama *time windows* tersebut "*open*". *Holding list* menurut Lau dkk, terdiri dari proses:

`

- relokasi dari *holding list*, yaitu memindahkan konsumen dari *holding list* ke rute yang ada
- relokasi ke *holding list*, yaitu memindahkan konsumen dari rute yang ada ke *holding list*.
- Tukar dengan *holding list*, yaitu menukar konsumen dari rute yang telah ada dengan konsumen lain di holding list.

B. Pengurutan Rute

Tujuan tahap ini yaitu melakukan urutan kunjungan kendaraan yang ada pada setiap rute yang telah dibentuk, sehingga waktu tempuh kendaraan dapat diminimalkan. Beberapa prosedur yang dapat digunakan dalam melakukan pengurutan kunjungan dengan metode heuristik diantaranya :

a. *Windows tightness rule*, konsumen dengan batas waktu yang terketat akan dipilih untuk dilayani

b. *Total capacity rule*, konsumen dengan pesanan tidak melebihi kapasitas motor dimasukkan dalam daftar rute perjalanan.

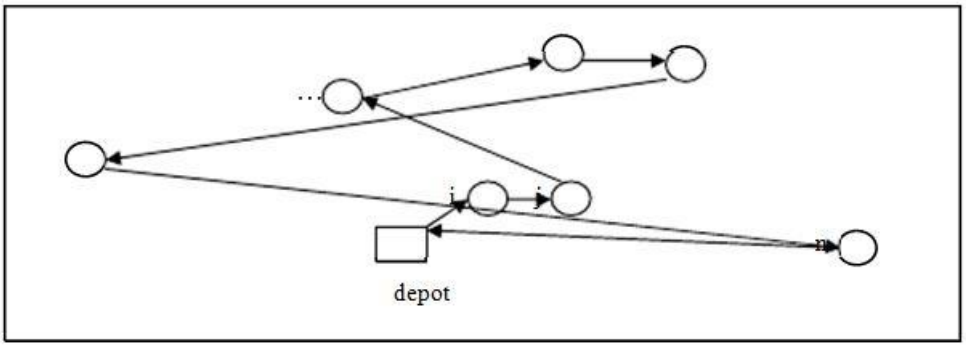
c. *Nearest-to-depot*, metode ini membangun rute dengan cara menambahkan kunjungan yang terdekat dengan depot. pada setiap iterasinya, setelah diawali dari depot, dilakukan pencarian pelanggan terdekat dengan depot untuk ditambahkan pada akhir rute tersebut. Rute baru dimulai dengan cara yang sama, jika tidak terdapat posisi yang fisibel untuk menempatkan pelanggan baru karena kendala kapasitas atau *time windows* tidak terpenuhi. Algoritma metode ini adalah sebagai berikut:

    1. Misalkan kendaraan yang tersedia dilambangkan dengan w.
    2. Mulailah suatu rute yang berawal dari depot.

`

3. Temukan pelanggan berikutnya *v* yang terdekat dengan titik awal dari rute *w*. Jika tidak ada pelanggan yang memungkinkan, tutup rute *w* dan pilih kendaraan baru lainnya lalu kembali ke langkah 2. Jika tidak ada lagi kendaraan maka proses selesai.
4. Tambahkan *v* pada akhir dari rute tersebut.
5. Kembali ke langkah 3. (ILOG, 1999)

Gambar 2.5 memberikan deskripsi tentang metode ini. Pada gambar tersebut, setelah diawali dari depot, dilakukan pencarian pelanggan yang terdekat dengan depot yaitu pelanggan *i*. Selanjutnya dari semua pelanggan lainnya yang tersisa, dicari yang terdekat dengan depot yaitu pelanggan *j*, untuk ditambahkan pada rute yang ada atau setelah pelanggan *i*. Pelanggan pada kasus VRPTW dipilih yang tidak melanggar kendala kapasitas dan *time windows*. Jika kapasitas kendaraan telah terpenuhi, mulai dengan rute baru sampai semua pelanggan terlayani.



Gambar 2. 5 Metode *nearest* to depot (ILOG, 1999)

UNIVERSITAS MULTIMEDIA NUSANTARA

`

2.4. Empat Kriteria Heuristik yang Baik

Menurut Cordeau dkk (2002) terdapat empat atribut heuristik yang baik, yaitu:

1. Akurasi

   Akurasi mengukur tingkat *value* sebuah solusi heuristik terhadap *value* yang optimal. Karena kata optimal biasanya tidak terdapat pada kasus VRP, perbandingan harus dilakukan dengan nilai *value* terbaik yang didapat.

   Barr dkk (1995) beranggapan bahwa menganalisa sebuah hasil heuristik sangat sulit. Para peneliti VRP sering melaporkan hasil yang diperoleh hanya berdasarkan kombinasi terbaik dari suatu algoritma parameter. Isu lain yang berkaitan dengan akurasi adalah konsistensi. Sudah pasti, *user* lebih memilih heuristik yang berjalan dengan baik sepanjang waktu daripada berjalan dengan "sempurna" tapi dapat pula berjalan dengan sangat buruk pada suatu waktu.

2. Kecepatan

   Seberapa penting kecepatan komputasi dalam *vehicle routing* ? Hal itu bergantung pada tingkat perencanaan pada masalah apa yang akan dipecahkan dan pada tingkat apa akurasi dibutuhkan. Sebagai contoh, Gendreau dkk (2001) menggambarkan peran penting yang dilakukan oleh komputasi paralel dalam relokasi sebuah mobil ambulan, dimana strategi tersebut harus ditentukan dalam rata-rata setiap tiga menit. Di lain pihak, pengaturan armada truk pengiriman paket, haruslah memiliki perencanaan

UNIVERSITAS MULTIMEDIA NUSANTARA

`

jangka panjang dan perhitungan yang dilakukan beberapa jam atau beberapa hari, karena pertaruhan biaya yang akan dikeluarkan.

3. Kesederhanaan

   Beberapa heuristik VRP jarang diimplementasikan karena terlalu rumit, sulit dipahami, dan sulit dikodekan oleh *programmer*. Algoritma yang mengandung banyak parameter akan sulit dimengerti dan kemungkinan besar tidak dapat dipakai. Masalah ini diangkat oleh Golden dkk (1998). Mereka berpendapat tidak hanya jumlah algoritmik parameter yang harus dibatasi, tetapi juga harus masuk akal untuk *end user*. Sebagai tambahan, heuristik haruslah *robust* untuk memastikan heuristik tersebut dapat bekerja dengan benar, meskipun tidak semua detail diimplementasikan.

4. Fleksibilitas

   Sebuah heuristik VRP yang baik harus cukup fleksibel untuk mengakomodasi segala kendala yang ada dalam kehidupan nyata. Sementara sebagian besar literatur VRP berfokus pada kapasitas dan panjang rute, seringkali perubahan dapat dibuat dengan tambahan batasan, namun ini tidak selalu dapat terjadi, dan performa dari heuristik dapat menurun secara drastis (Cordeau dkk, 2002).

UNIVERSITAS MULTIMEDIA NUSANTARA