

SISTEM TRACER PAKET PADA UNIT PROCESSING CENTER POS INDONESIA (PERSERO) MENGGUNAKAN METODE TRAVELLING SALES PERSON PROBLEM

Dahlan Abdullah¹⁾, Richki Hardi²⁾

^{1,2)}Program Studi Teknik Informatika, Universitas Malikussaleh
Reuleut, Aceh Utara, Aceh-Indonesia
E-mail : dahlan.unimal@gmail.com

ABSTRACT

Traveling salesperson problem-TSP problem is an optimization problem to find the optimal way for the traveling salesman who wants to visit several cities, and returned to the original departure city. TSP is a difficult problem when viewed from the point of computing. Several methods have been used to solve these problems but until now not been found mangkus algorithms to solve them. The easiest way to solve TSP is to try all possible routes and find the optimal route. However, at the time of the very practical now required to complete an algorithm that quickly so that the TSP solution obtained near optimal solutions.

Keywords: Graph algorithm, traveling salesperson problem, Package Tracking, Web

PENDAHULUAN

Latar belakang Masalah

Perkembangan ilmu pengetahuan dan teknologi kian pesat, hal tersebut dapat dilihat dan dirasakan secara langsung maupun tidak langsung. Perkembangan tersebut tengah berdampak pada segala aspek kehidupan manusia. Globalisasi yang terjadi sekarang ini mengakibatkan terjadinya perubahan-perubahan yang dampaknya mempengaruhi segala aspek kehidupan dan terjadi secara berkelanjutan, termasuk di lingkungan perusahaan pengiriman barang standar nasional. Globalisasi yang terjadi dewasa ini mengakibatkan terjadinya perubahan-perubahan yang dampaknya mempengaruhi segala aspek kehidupan dan terjadi secara berkelanjutan pada lingkungan perusahaan pengiriman barang standar nasional.

PT. Pos Indonesia adalah suatu perusahaan ekspedisi skala nasional yang melayani jasa antaran barang antar kota antar propinsi yang memiliki berbagai cabang yang terletak di seluruh penjuru Indonesia. Kantor pos Lhokseumawe mempunyai beberapa daerah operasional antaran paket. Untuk menyampaikan paket dari satu kota ke kota lainnya, kantor pos harus dapat melakukan pemilihan rute yang tepat agar paket dapat sampai ke kota tujuan dalam waktu yang cepat. Pe-rute-an (*routing*) adalah menentukan lintasan yang dilalui oleh paket dari kota pengirim (asal) ke kota penerima (tujuan).

Proses antaran paket yang sedang berjalan di kantor pos, khususnya daerah Aceh menggunakan

rute transportasi umum, dengan armada yang sangat terbatas sehingga membutuhkan waktu yang lama.

Jika kantor pos kecamatan (KPC) ingin mengirim paket ke kota lain maka paket tersebut harus diolah terlebih dahulu oleh kantor pos pemeriksa (KPRK) untuk kemudian dikirim ke kota tujuan, walaupun jarak antara Kantor pos kecamatan lebih dekat dengan kota tujuan.

Permasalahannya adalah bagaimana menentukan rute yang tepat sehingga paket dapat sampai ke kota tujuan dalam waktu yang sesingkat mungkin dengan menggunakan rute tersebut, paket yang sampai ke suatu kota atau kantor pos dapat diarahkan ke kantor pos berikutnya yang tepat sehingga paket menuju kota atau kantor pos penerima dengan delay (*delay*) waktu yang minimum. Dengan kata lain, harus menentukan lintasan terpendek dan mendekati yang akan dilalui oleh paket tersebut dari kantor pos pengirim ke kantor pos penerima.

Dalam proses antaran paket dari satu kota ke kota lain tentunya perlu ada pertimbangan efisiensi waktu dan biaya oleh Perusahaan sehingga diperlukan ketepatan dalam menentukan rute terpendek antar suatu kota. Hasil penentuan rute terpendek bisa didapatkan dengan menggunakan metode pendekatan algoritma *Travelling Salesperson Problem* (TSP), yaitu algoritma yang mencari panjang lintasan terpendek dan mendekati optimal dari titik asal ke titik tujuan

dan kembali ke titik asal dalam sebuah graf berbobot tersambung dengan biaya minimal.

Rumusan Masalah

Adapun permasalahan dalam penelitian ini dapat dirumuskan sebagai berikut :

1. Bagaimana memilih dan menentukan rute-rute terpendek yang mendekati optimal untuk antaran paket dari kota asal ke kota tujuan kemudian kembali ke kota asal.
2. Bagaimana mengimplementasikan algoritma TSP dalam menentukan rute pada proses antaran paket.
3. Bagaimana membuat aplikasi untuk pencarian rute antaran paket di kantor pos Lhokseumawe.

Keaslian Penelitian

Penelitian sejenis yang pernah dilakukan oleh peneliti sebelumnya dilakukan dengan cara menggunakan algoritma djikstra dalam menentukan lokasi titik terdekat pengeboran batubara. Dengan dasar tersebut diatas penulis akan mencoba melakukan penelitian yang bersifat baru, sampai dengan saat ini sepanjang yang penulis ketahui, belum ada dan belum pernah dilakukan penelitian tentang penggunaan sistem *tracking* antaran paket pada unit pelayanan PT. Pos Indonesia menggunakan metode pendekatan Algoritma *Graph "Travelling Salesperson Problem (TSP)"*

Manfaat Penelitian

Manfaat dari penelitian adalah diharapkan dapat menjadi salah satu acuan bagi kantor Pos Lhokseumawe dalam menentukan rute pengiriman paket dari satu kota ke kota lainnya sehingga dapat menghemat waktu dalam proses pengiriman paket. Diharapkan dapat memberikan sumbangan bagi pengembang ilmu dibidang komputer dan informatika serta memanfaatkan kemajuan teknologi untuk kemajuan masyarakat, pembelajaran bagi mahasiswa teknik informatika khususnya dan sebagai implementasi ilmu pengetahuan dari penelitian tersebut.

Tujuan Penelitian

Tujuan penelitian ini adalah merancang dan mengimplementasikan sebuah sistem penelusuran paket yang dapat memberikan kemudahan bagi suatu permasalahan *tracking* dengan menggunakan metode pendekatan algoritma TSP pada kantor Pos Lhokseumawe.

Batasan Masalah

Berdasarkan latar belakang masalah tersebut, agar hasil penelitian ini maksimal maka pembahasan masalah hanya dibatasi pada:

1. Pencarian rute terpendek antaran paket menggunakan algoritma Graph TSP.

2. Rute di dalam sistem ini mengacu pada titik yang telah ditentukan oleh PT.Pos sebagai sarana untuk memberikan laporan.
3. Data yang digunakan dalam sistem ini adalah data sekunder (rute, data jarak, dan data kantor pos di Aceh) yang bersumber dari kantor pos Lhokseumawe dan kantor perhubungan Aceh Utara.
4. Dilengkapi fasilitas menu tambahan yaitu pelacakan nomor resi paket, ekspedisi berita seputar perusahaan, pesan singkat yang berfungsi untuk pemberitahuan atau informasi penting lainnya, kritik dan saran bagi perusahaan dan konsumen serta fasilitas daftar tarif yang berfungsi untuk mengetahui jumlah biaya yang harus dibayar berdasarkan berat paket yang dikirim.

Tinjauan Pustaka dan Landasan Teori

Tinjauan Pustaka

Penelitian ini juga mengacu pada penelitian yang dilakukan oleh Wijayanto (2005), dengan judul SIG untuk pemetaan transceiver station BTS. Telkom Flexi PT.Telkom cabang Bantul. Pada penelitian tersebut peneliti menggunakan perangkat lunak arcview tetapi penggunaannya belum menggunakan hotlink untuk menampilkan informasi yang lebih detail sehingga informasi yang dihasilkan hanya berupa atribut dari theme yang ada. Arcview memiliki fasilitas hotlink sehingga dapat membantu menampilkan informasi yang lebih lengkap dan menarik. Dan penelitian ini penulis telah menggunakan fasilitas hotlink sehingga dapat ditampilkan informasi yang lebih luas dan lebih detail. Karena fasilitas hotlink dapat menerima masukan yang berupa file text, image, dan file doc, sehingga dapat ditampilkan informasi yang lebih luas dan lebih menarik.

Dari penelitian tugas akhir oleh Arleadi (2002) *Analisis Perancangan e-Commerce pada toko buku online* tersebut hanya merancang proses perancangan dari produsen penjual buku ke konsumen yang akan membeli buku tersebut dari situs yang dibuatnya, tidak menampilkan informasi secara detail dan menarik seperti halnya pada penelitian penulis ini. Tugas akhir tentang *sistem informasi pemesanan produk secara online dengan aplikasi ASP* di Timboel Keramik Kasongan Bantul Yogyakarta oleh Pamujianto (2004), terlihat bahwa sistem yang dibuat sebatas informasi dan pemesanan produk belum mencakup penjualan secara online yang dapat diakses langsung oleh calon konsumen melalui web address tertentu.

Tugas akhir oleh Waluyo Basuki (2005) *Perancangan Web e-Commerce sebagai media informasi dan pemasaran pada software house IQSOFT Yogyakarta* memiliki sistem dengan menggunakan fasilitas seperti forum diskusi, buku tamu, berita dan free download, dan jenis transaksi

ini menggunakan transaksi pembayaran tunai/cash jika orang yang bersangkutan tinggal dalam satu daerah, misalnya dalam wilayah Yogyakarta, namun sistem ini mengalami kelemahan pada administrasi yaitu tidak adanya sistem otomatisasi penghapusan data pemesan yang telah melakukan konfirmasi, pada hal ini penulis akan melengkapi kekurangan-kekurangan pada penelitian sebelumnya sehingga dunia internet atau web dapat dirasakan lebih handal dan professional.

Landasan Teori

Pengertian Sistem

Sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi artinya suatu sistem saling bekerjasama membentuk satu kesatuan. Komponen sistem-sistem dapat saling mengisi untuk mencapai suatu sasaran atau tujuan tertentu.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen (*components*), batasan sistem (*boundary*), lingkungan luar sistem (*environment*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objective*) atau tujuan (*goal*), Mulyawarman (2004).

Karena itu sebuah sistem terdiri dari bagian-bagian saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud. Untuk itu sebuah sistem bukanlah seperangkat unsur yang tersusun secara tidak teratur, tetapi terdiri dari unsur yang dapat dikenal sebagai saling melengkapi karena satunya maksud, tujuan dan sasaran, Kadir (2002)

Algoritma Graph-Travelling Salesperson Problem

Di dunia nyata, seringkali kita menghadapi permasalahan yang memiliki satu atau lebih ciri-ciri berikut ini:

1. Ruang masalah sangat besar, kompleks, dan sulit dipahami;
2. tidak bisa diselesaikan menggunakan metode-metode konvensional;
3. Terdapat batasan waktu, misalnya dalam sistem waktu nyata (*real timesystem*).
4. Solusi yang diharapkan tidak harus paling optimal, tetapi 'bagus' atau bisa diterima;
5. Kurang atau bahkan tidak ada pengetahuan yang memadai untuk merepresentasikan masalah ke dalam ruang pencarian yang lebih sempit;
6. Tidak tersedia analisis matematika yang memadai;

Jika kita lihat sekilas, TSP adalah masalah optimasi yang sederhana dan mudah diselesaikan dengan algoritma searching konvensional. Hal ini benar jika jumlah node (bisa berupa alamat surat atau kota) hanya sedikit, misalnya 10 node. Tetapi,

masalah ini menjadi serius dan-sulit diselesaikan secara konvensional ketika jumlah node yang harus dikunjungi sangat banyak, misalnya 500 node.

Pada TSP, jumlah kemungkinan urutan kunjungan bisa dihitung menggunakan rumus permutasi dengan sedikit modifikasi. Rumus dasar permutasi adalah

$$nP_k = \frac{n!}{(n-k)!}$$

di mana n adalah jumlah seluruh objek dalam semesta dan k adalah jumlah objek yang diseleksi sebagai suatu solusi. Pada kasus TSP, n adalah jumlah seluruh kota yang harus dikunjungi, dan k adalah jumlah kota yang merepresentasikan satu solusi. Untuk kasus TSP, nilai n adalah sama dengan k . Dengan memperhatikan bahwa TSP bersifat siklus (*cycle*) dan bergantung pada graph yang merepresentasikan urutan kota, maka kita perlu sedikit memodifikasi rumus tersebut. Siklus bisa diartikan bahwa sebuah jalur dengan urutan 1-2-3 memiliki biaya yang sama dengan dua jalur lainnya: 2-3-1 dan 3-1-2. Terdapat dua jenis graph, yaitu asimetris dan simetris.

Pada graph asimetris, biaya dari kota 1 ke kota 2 tidak sama dengan biaya dari kota 2 ke kota 1. Sedangkan pada graph simetris, biaya dari kota 1 ke Kota 2 adalah sama dengan biaya dari kota 2 ke kota 1. Perbedaan graph ini akan mempengaruhi jumlah kemungkinan jalur.

Untuk TSP pada graph asimetris, jumlah jalur yang mungkin adalah permutasi dari jumlah kota dibagi dengan jumlah kota. Hal ini dapat dipahami karena secara siklus, sebuah jalur dengan urutan 1-2-3 memiliki biaya yang sama dengan dua jalur lainnya: 2-3-1 dan 3-1-2. Tetapi, jalur dengan urutan 1-2-3 tidaklah sama dengan jalur 3-2-1. Jadi, jumlah jalur yang mungkin untuk TSP pada graph asimetris dengan 10 kota adalah

$$nP_k = \frac{10P_{10}}{10} = \frac{10!}{10} = 362.880$$

Sedangkan untuk TSP pada graph simetris, jumlah jalur yang mungkin adalah permutasi dari jumlah kota dibagi dengan dua kali jumlah kota. Hal ini disebabkan sebuah jalur dengan urutan 1-2-3 adalah sama dengan lima jalur lainnya: 3-1-2, 2-3-1, 3-2-1, 1-3-2, dan 2-1-3. Jadi, jumlah jalur yang mungkin untuk TSP pada graph simetris dengan 10 kota adalah

$$\frac{10P_{10}}{2 \times 10} = \frac{10!}{20} = 181.440$$

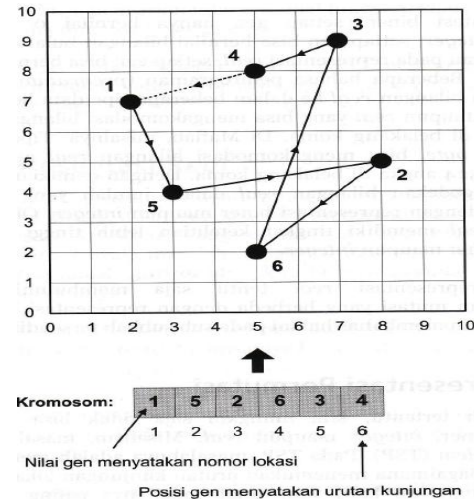
untuk graph asimetris yang hanya berukuran 20 node, jumlah jalur kunjungan yang mungkin sudah mencapai 121.645.100.408.832.000 (lebih dari 121.645 trilyun). Bagaimana jika terdapat 80 node yang harus dikunjungi untuk TSP pada graph asimetris? Jumlah kemungkinan urutan kunjungan menjadi sangat besar, yakni $8,94 \times 10^{116}$. Bagaimana

jika terdapat 100 node, 500 node atau bahkan ribuan node yang harus dikunjungi? Tentu saja hal ini sangat sulit (atau bahkan tidak bisa) diselesaikan menggunakan Algoritma konvensional dengan kondisi kecepatan prosesor dan ukuran memori yang ada saat ini. Algoritma konvensional, seperti Dijkstra misalnya, memerlukan waktu yang sangat lama untuk menyelesaikan TSP pada graph asimetris 50 node. Algoritma Dijkstra sudah tidak realistis untuk TSP pada graph asimetris yang berisi lebih dari 50 node. Dengan demikian TSP memiliki dua ciri yang pertama di atas, yaitu ruang masalah sangat besardan sangat sulit (bahkan tidak bisa) diselesaikan menggunakan metode konvensional.

Algoritma Dijkstra memang menjamin ditemukannya solusi paling optimal. Dalam kasus ini, algoritma Dijkstra pasti menghasilkan jalur pengiriman dengan total biaya paling minimal. Tetapi, algoritma Dijkstra mungkin kurang tepat untuk kasus ini karena waktu prosesnya terlalu lama. Untuk masalah seperti ini, algoritma yang kita butuhkan adalah algoritma yang bisa menemukan jalur pengiriman dalam waktu yang cepat meskipun jalur tersebut bukan yang paling optimal. Misalkan, metode A hanya membutuhkan waktu 30 menit untuk menemukan jalur pengiriman, yang total biayanya 8 jam 20 menit (lebih besar dari pada jalur pengiriman yang ditemukan Dijkstra 8 jam). Tetapi, jika kita jumlahkan waktu pencarian jalur dan waktu pengiriman surat, maka tukang pos hanya membutuhkan waktu 30 menit + 8 jam 20 menit = 8 jam 50 menit. Dalam hal ini, metode A menjamin waktu nyata karena tukang pos memiliki waktu lebih banyak, yaitu 9 jam. Dengan demikian, TSP mungkin saja memiliki ciri nomor 3 dan 4 di atas, yaitu terdapat batasan waktu dan solusi yang diharapkan tidak harus paling optimal tetapi bisa diterima. Representasi permutasi Untuk masalah tertentu, kita mungkin saja tidak bisa menggunakan representasi biner, integer maupun real. Misalkan, masalah Travelling Salesman Problem (TSP). Pada TSP, masalahnya adalah mencari "urutan" Bukan "nilai". Bagaimana menemukan urutan kunjungan lokasi (satu lokasi hanya dikunjungi satu kali) yang total "nilai"-nya paling optimal (bisa minimal atau maksimal bergantung tujuannya). "Nilai," di sini bisa berupa jarak, biaya, kenyamanan, dan sebagainya. Bagi seorang kurir, misalnya, tujuannya adalah menemukan urutan lokasi pengantaran paket yang total jaraknya paling minimal. Bagi seorang wisatawan, tujuannya bisa berupa urutan lokasi wisata yang tingkat kenyamanan jalannya paling maksimal. bagaimana membangun kromosom dengan representasi permutasi untuk masalah TSP? Satu kata kunci yang harus selalu kita ingat "Satu kromosom harus menyatakan satu solusi". Dua kata kunci lain yang juga perlu kita perhatikan adalah

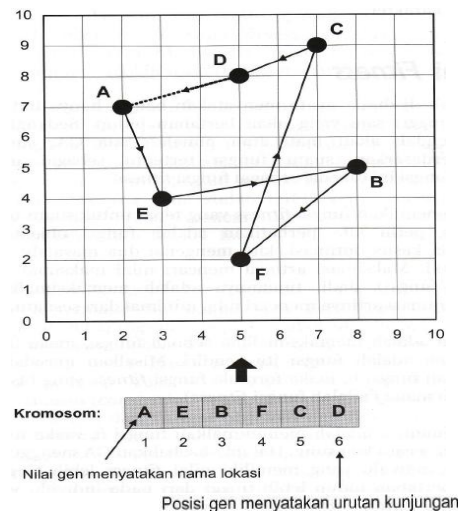
"posisi" dan "nilai" gen. Posisi gen (indeks pada kromosom) bisa digunakan untuk menyatakan urutan kunjungan lokasi.

Dengan demikian, jika terdapat 6 lokasi untuk masalah TSP, maka panjang kromosomnya adalah 6 gen yang bernilai integer dalam interval [1, 6] yang menyatakan nomor lokasi. Sedangkan posisi gen, ke-1 sampai ke-6, menyatakan urutan kunjungan. Sebagai ilustrasi, perhatikan contoh kromosom permutasi pada gambar berikut ini.



Gambar 1. Contoh kromosom permutasi yang mengodekan masalah TSP dengan 6 lokasi yang diberi Nomor 1 sampai 6. Kromosom tersebut menyatakan satu solusi yang mungkin berupa urutan kunjungan 1-5-2-6-3-4

Keempat jenis representasi di atas (biner, integer, real, maupun permutasi) menggunakan angka (numerik) sebagai nilai-nilai gen (allele). Sebenarnya bisa saja kita merepresentasikan nilai gen menggunakan huruf. Misalnya pada masalah TSP terdapat 6 lokasi dengan nama A, B, C, D, E, dan F. Kita bias menyatakan nama-nama lokasi tersebut secara langsung ke dalam kromosom. Perhatikan gambar berikut ini.



Gambar 2. Contoh kromosom permutasi yang mengodekan masalah TSP dengan 6 lokasi yang diberi nama A, B, C, D, E, dan F. Kromosom tersebut menyatakan satu solusi yang mungkin berupa urutan kunjungan A-E-B-F-C-D

Permasalahan matematika tentang *Traveling Salesman Problem* dikemukakan pada tahun 1800 oleh matematikawan Irlandia William Rowan Hamilton dan matematikawan Inggris Thomas Penyngton. Awalnya dari permainan Icosian Hamilton yang membutuhkan pemain untuk menyelesaikan perjalanan dari 20 titik menggunakan hanya jalur-jalur tertentu.



Gambar 3. permainan Icosian Hamilton

Bentuk umum dari TSP pertama dipelajari oleh para matematikawan mulai tahun 1930. Diawali oleh Karl Menger di Vienna dan Harvard. Setelah itu permasalahan TSP dipublikasikan oleh Hassler Whitney dan Merrill Flood di Princeton.

Prosedur Sederhana Pemecahan TSP

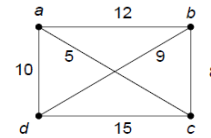
Dalam penyelesaian masalah TSP kita dapat membagi kedalam 2 metode, yaitu metode optimal dan metode aproksimasi. Metode optimal akan menghasilkan hasil yang optimal (minimum) sedangkan metode aproksimasi akan menghasilkan hasil yang mendekati optimal.

Metode Optimal

Sejak permasalahan TSP ditemukan pada tahun 1800 oleh matematikawan Irlandia Sir William Rowan Hamilton dan matematikawan Inggris Thomas Penyngton Kirkman, pusat perhatian studi ini adalah menemukan secara pasti nilai minimum dari persoalan TSP dengan konsekuensi dibutuhkan waktu yang cukup lama untuk menyelesaikannya.

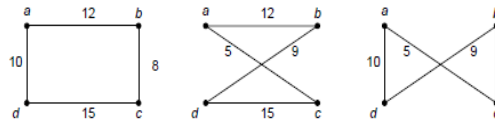
Complete Enumeration

Metode ini akan mengenumerasi setiap kemungkinan yang terdapat dalam graf, setelah itu algoritma ini akan membandingkan lintasan mana yang paling minimum. Misal untuk kasus berikut ini :



Gambar 4. contoh empat titik lintasan kasus

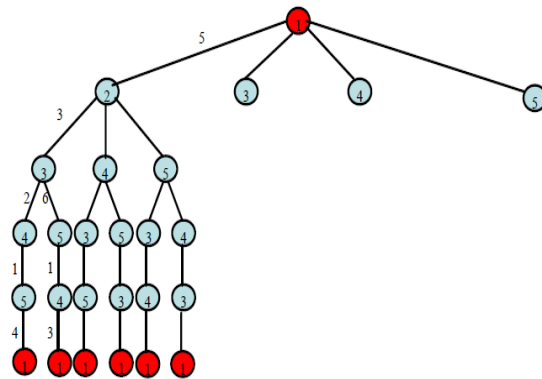
jumlah titik yang terdapat adalah empat buah dan banyak kemungkinan lintasannya adalah tiga buah. Yaitu :



Gambar 5. contoh tiga macam lintasan kasus

Lintasan pertama = (a,b,c,d,a) atau (a,d,c,b,a) Mempunyai panjang = $10 + 12 + 8 + 15 = 45$. Lintasan Kedua = (a,c,d,b,a) atau (a,b,d,c,a) Mempunyai panjang = $12 + 5 + 9 + 15 = 41$, Lintasan Ketiga = (a,c,b,d,a) atau (a,d,b,c,a) Mempunyai panjang = $10 + 5 + 9 + 8 = 32$ Dari hasil enumerasi ini didapat hasil minimum yaitu 32. Tetapi jumlah enumerasi dari algoritma ini adalah $(n - 1)!$ yang tidak akan efisien jika jumlah n bernilai sangat besar.

Branch and Bound



Gambar 6. Branch and Bound

Sama dengan complete enumeration, pada algoritma Branch and Bound ternyata memiliki kompleksitas algoritma $(n-1)!$, dimana n adalah jumlah kota. Untuk kasus diatas hasil yang di capai adalah 15

Dynamic Programming

Misalkan $G=(V,E)$ adalah graf lengkap berarah dengan sisi-sisi yang diberi harga $c_{ij} > 0$ untuk setiap j dan j adalah simpul-simpul di dalam V . Misalkan $|V| = n$ dan $n > 1$. Setiap simpul diberi nomor 1, 2, ..., n. Asumsikan perjalanan (tur) dimulai dan berakhir pada simpul 1. Setiap tur terdiri dari sisi $(1,k)$ untuk beberapa $k \in V - \{1\}$ dan sebuah lintasan dari simpul k ke simpul 1.

Lintasan dari simpul k ke simpul 1 tersebut melalui setiap simpul di dalam $V - \{1, k\}$ tepat hanya sekali. Prinsip Optimalitas: jika tur tersebut optimal maka lintasan dari simpul k ke simpul 1 juga menjadi lintasan k ke 1 terpendek yang melalui simpul-simpul di dalam $V - \{1, k\}$.

Misalkan $f(i, S)$ adalah bobot lintasan terpendek yang berawal pada simpul i , yang melalui semua simpul di dalam S dan berakhir pada simpul 1. Nilai $f(1, V - \{1\})$ adalah bobot tur terpendek. Berdasarkan prinsip optimalitas tersebut, diperoleh hubungan rekursif sebagai berikut:

$$f(1, V - \{1\}) = \min_{2 \leq k \leq n} \{c_{1k} + f(k, V - \{1, k\})\} \quad (1)$$

Dengan merampatkan persamaan (1), diperoleh

$$f(i, \emptyset) = c_{i,1}, \quad 2 \leq i \leq n$$

(basis)

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

(rekurens) (2)

Persamaan (1) dapat dipecahkan untuk memperoleh $\{1\}$ jika kita mengetahui $f(k, V - \{1, k\})$ untuk semua pilihan nilai k . Nilai f tersebut dapat diperoleh dengan menggunakan persamaan (2).

Kita menggunakan persamaan(2) untuk memperoleh $f(i, S)$ untuk $|S| = 1$, kemudian kita dapat memperoleh $f(i, S)$ untuk $|S| = 2$, dan seterusnya. Bila $|S| = n - 1$, nilai i dan S ini diperlukan sedemikian sehingga $i \neq 1, 1 \notin S$ dan $i \notin S$.

Tinjau persoalan TSP untuk $n = 4$:

$$\begin{bmatrix} 0 & 10 & 15 & 20 \\ 5 & 0 & 9 & 10 \\ 6 & 13 & 0 & 12 \\ 8 & 8 & 9 & 0 \end{bmatrix}$$

Ini adalah matriks ketetanggan dari suatu graf yang akan kita cari lintasan terpendeknya.

Tahap 1 :

$$f(i, \emptyset) = c_{i,1}, \quad 2 \leq i \leq n$$

Diperoleh :

$$f(2, \emptyset) = c_{21} = 5;$$

$$f(3, \emptyset) = c_{31} = 6;$$

$$f(4, \emptyset) = c_{41} = 8;$$

terdapat tiga kemungkinan yang dapat dijadikan lintasan pertama.

Tahap 2 :

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

Untuk $|S| = 1$

Diperoleh:

$$f(2, \{3\}) = \min\{c_{23} + f(3, \emptyset)\} = \min\{9 + 6\} = \min\{15\} = 15$$

$$f(3, \{2\}) = \min\{c_{32} + f(2, \emptyset)\} = \min\{13 + 5\} = \min\{18\} = 18$$

$$f(4, \{2\}) = \min\{c_{42} + f(2, \emptyset)\} = \min\{8 + 5\} = \min\{13\} = 13$$

$$f(2, \{4\}) = \min\{c_{24} + f(4, \emptyset)\} = \min\{10 + 8\} = \min\{18\} = 18$$

$$f(3, \{4\}) = \min\{c_{34} + f(4, \emptyset)\} = \min\{12 + 8\} = \min\{20\} = 20$$

$$f(4, \{3\}) = \min\{c_{43} + f(3, \emptyset)\} = \min\{9 + 6\} = \min\{15\} = 15$$

Tahap 3:

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

untuk $|S| = 2$ dan $i \neq 1, 1 \notin S$ dan $i \notin S$.

Diperoleh:

$$\begin{aligned} f(2, \{3, 4\}) &= \min\{c_{23} + f(3, \{4\}), c_{24} + f(4, \{3\})\} \\ &= \min\{9 + 20, 10 + 15\} \\ &= \min\{29, 25\} = 25 \end{aligned}$$

$$\begin{aligned} f(3, \{2, 4\}) &= \min\{c_{32} + f(2, \{4\}), c_{34} + f(4, \{2\})\} \\ &= \min\{13 + 18, 12 + 13\} \\ &= \min\{31, 25\} = 25 \end{aligned}$$

$$\begin{aligned} f(4, \{2, 3\}) &= \min\{c_{42} + f(2, \{3\}), c_{43} + f(3, \{2\})\} \\ &= \min\{8 + 15, 9 + 18\} \\ &= \min\{23, 27\} = 23 \end{aligned}$$

Dengan menggunakan persamaan (1) diperoleh:

$$\begin{aligned} f(1, \{2, 3, 4\}) &= \min\{c_{12} + f(2, \{3, 4\}), c_{13} + f(3, \{2, 4\}), c_{14} + f(4, \{2, 3\})\} \\ &= \min\{10 + 25, 15 + 25, 20 + 23\} \\ &= \min\{35, 40, 43\} = 35 \end{aligned}$$

Jadi, bobot tur yang berawal dan berakhir di simpul 1 adalah 35.

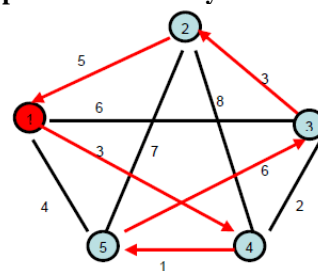
Lintasan yang dilalui di dalam tur tersebut dapat direkonstruksi jika kita menyimpan pada setiap $f(i, S)$ nilai j yang meminimumkan ruas kanan persamaan(2).

Misalkan $J(i, S)$ adalah nilai yang dimaksudkan tersebut. Maka, $J(1, \{2, 3, 4\}) = 2$. Jadi, tur mulai dari simpul 1 selanjutnya ke simpul 2.

Simpul berikutnya dapat diperoleh dari $f(2, \{3, 4\})$, yang mana $J(2, \{3, 4\}) = 4$. Jadi, simpul berikutnya adalah simpul 4.

Simpul terakhir dapat diperoleh dari $f(4, \{3\})$, yang mana $J(4, \{3\}) = 3$. Jadi, tur yang optimal adalah 1, 2, 4, 3, 1 dengan bobot (panjang) = 35.

Metode Aproksimasi Greedy Heuristic



Gambar 7. Greedy Heuristic

Pada algoritma ini, pemilihan lintasan akan dimulai pada lintasan yang memiliki nilai paling minimum, setiap mencapai suatu kota, algoritma ini akan memilih kota selanjutnya yang belum dikunjungi dan memiliki jarak yang paling minimum. Algoritma ini disebut juga *Nearest Neighbour*.

Kompleksitas algoritma ini memang sangat mengagumkan yaitu $O(n)$, tetapi hasil yang kita dapat bisa sangat jauh dari hasil yang optimal, semakin banyak kota semakin besar pula perbedaan hasil yang dicapai. Misalnya untuk contoh kasus yang sama dengan algoritma *Branch and Bound* sebelumnya yang menghasilkan nilai 15, maka algoritma ini menghasilkan nilai 18 berbeda sebesar 20% dari hasil sebelumnya padahal jumlah kota hanya 5 buah.

Algoritma algoritma lainnya

Heuristics

Teknik ini digunakan untuk mencari jawaban dari masalah kombinatorial dengan secepat mungkin. Algoritma tradisional akan gagal ketika menghadapi permasalahan yang sangat rumit, seperti permasalahan TSP dengan jumlah kota (n) yang sangat besar.

Metode Heuristic memberikan pendekatan untuk menyelesaikan permasalahan optimasi kombinatorial. *Combinatorial search* akan memberi kita hasil yang mungkin dan mencari yang hasil yang mendekati optimal dari hasil-hasil tersebut. Tetapi mungkin memang tidak ada metode Heuristik yang menghasilkan solusi yang merupakan solusi optimal. Metode Heuristik seperti *genetic algoritma*, *simulated annealing*, dan *neural network* mengusahakan suatu cara untuk mencari hasil yang baik tapi bukan yang terbaik.

Genetic Algoritma

Algoritma Genetic mendapatkan inspirasi dari proses evolusi dan seleksi alam. Melalui proses seleksi alam, organisme atau makhluk hidup beradaptasi untuk mengoptimalkan kesempatan mereka untuk tetap bias hidup didalam sebuah lingkungan. Mutasi secara random menghasilkan kode genetik dari makhluk hidup, yang kemudian diturunkan kepada anaknya. Jika mutasi meningkatkan kualitas genetik, maka secara otomatis sang anak akan terbantu untuk selamat.

Simulated Annealing

Inspirasi tentang metode ini datang dari proses fisika mengenai pendinginan lelehan material yang berubah menjadi padat. Ketika lelehan besi didinginkan terlalu cepat, maka retakan dan gelembung udara akan muncul, menghancurkan permukaan dan integritas strukturnya. Oleh karena itu untuk menghasilkan besi yang baik, besi harus

didinginkan secara pelan-pelan dan diberi jeda. *Annealing* adalah teknik metalurgi yang menggunakan ilmu penjadwalan proses pendinginan untuk menghasilkan efisiensi dalam menggunakan energi dan menghasilkan besi yang optimal.

Neural Network

Neural Network adalah paradigma komputasional yang terinspirasi dari arsitektur otak manusia. Ketika otak memiliki intuisi yang sangat baik dalam memecahkan persoalan, maka mesinpun seharusnya dibangun seperti itu.

METODE PENELITIAN

Lokasi Penelitian

Lokasi penelitian adalah PT.Pos Indonesia (Persero), kantor cabang Lhokseumawe, Dinas Perhubungan Aceh Utara, Bappeda Aceh Utara dan Bappeda Lhokseumawe.

Alat dan Bahan Penelitian

Alat Penelitian

1) Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini yaitu :Spesifikasi perangkat keras (*hardware*) yang digunakan pada penelitian ini yaitu berupa Laptop dengan spesifikasi tinggi Intel Core2Duo, Memory 2GB, dan nVidia Graphic 512MB, Serta alat cetak printer Canon MP450 untuk memudahkan peneliti dalam melakukan penelitian

2) Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian ini yaitu:

- a. Microsoft Windows dan Linux
- b. PHP
- c. MySQL
- d. Macromedia Dreamweaver
- e. Adobe Photoshop
- f. Fireworks MX
- g. Macromedia Flash
- h. Microsoft Office Word 2007
- i. Server Apache PHP Triad, vertrigo dan wampserver

Bahan Penelitian

Bahan penelitian yang dibutuhkan adalah sebagai berikut :

- a. Data kantor pos di Nanggroe Aceh Darussalam
- b. Data Paket dikantor pos
- c. Data rute yang dilalui pada saat antaran paket
- d. Data lokasi kantor pos
- e. Data jarak antar kota dan kabupaten di Nanggroe Aceh Darussalam
- f. Peta jalan Nanggroe Aceh Darussalam.
- g. Data berita seputar perusahaan
- h. Data *profile* Perusahaan
- i. Data Tarif

Metode Pengumpulan Data

Metodologi yang digunakan adalah analisis dan desain terstruktur dengan tahap-tahap sebagai berikut :

Penelitian Lapangan (Field Research)

- a. Dalam melakukan penelitian ini penulis melakukan *Observasi*, Yaitu metode pengumpulan data dengan menggunakan pengamatan langsung dan pencatatan dengan sistematis terhadap gejala atau fenomena yang terkait tanpa mengajukan pertanyaan.
- b. wawancara dengan Kepala bagian Pengolahan data dan Kepala bagian pusat informasi di kantor pos Lhokseumawe. Teknik analisis terhadap sistem yang ada atau sedang berjalan
- c. Implementasi, Yaitu metode dengan cara mengimplementasikan hasil perancangan yang telah dibuat menjadi suatu tampilan yang menarik sehingga memudahkan dalam pembelajaran tentang objek penelitian.
- d. Metode Uji Coba, Yaitu suatu metode dimana perancangan yang telah diimplementasikan kedalam program dapat diuji cobakan kebenarannya kepada orang lain yang ingin mempelajarinya.

Penelitian Kepustakaan (Library Research)

Metode ini merupakan metode pengumpulan data dengan cara mempelajari literature, paket modul dan panduan, buku-buku pedoman, buku-buku perpustakaan dan segala kepustakaan lainnya yang dianggap perlu dan mendukung.

Langkah-langkah Penelitian

Langkah-langkah dalam melakukan penelitian ini adalah sebagai berikut

- a. Tahap Perancangan Sistem
- b. Sistem dirancang dengan menggunakan *DFD (Data Flow Diagram)* yaitu untuk menentukan proses input dan proses output dalam sistem.
- c. Perancangan graf dan algoritma TSP yaitu untuk
 - Menentukan graf yang akan dipakai.
 - Menggambar graf sesuai dengan Peta jalan seluruh Aceh.
 - Menentukan titik-titik didalam graf.
 - Memasukkan bobot nilai dalam graf.
 - Menentukan rute-rute yang bisa dilewati untuk antaran paket dari titik awal ke titik-titik selanjutnya.
 - Menentukan rute terpendek atau nilai minimumnya dengan menggunakan algoritma TSP.
- d. Tahap Pembuatan Sistem
Langkah-langkah yang digunakan untuk membuat sistem adalah sebagai berikut :
 - Menentukan bahasa pemrograman yang akan dipakai.

- Membuat tabel-tabel database.
- Merancang menu interface sistem.
- Mengimplementasikan sistem kedalam bahasa pemrograman.

e. Tahap Pengujian Sistem

Langkah-langkah yang digunakan dalam menguji sistem adalah sebagai berikut:

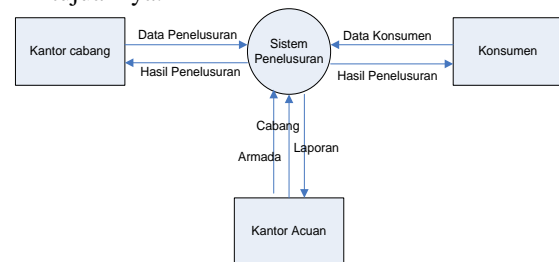
- Melakukan Test Case
- Memberikan jenis uji Black Box test
- Memberikan jenis uji Alpha test

HASIL DAN PEMBAHASAN

Analisa Kebutuhan Sistem

Diagram Konteks (Context Diagram)

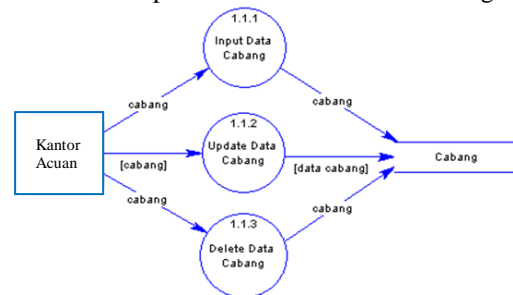
Context diagram merupakan diagram yang menggambarkan aliran data pada aplikasi Sistem Penentuan Rute optimal Antaran Paket secara garis besar yang mewakili keseluruhan dari sistem yang akan dibuat. Diagram konteks juga merupakan penggambaran sistem secara garis besar dan menggambarkan hubungan masukan dan keluaran antara sistem dengan dunia luarnya. Diagram ini menginventarisasi data yang masuk ke sistem beserta sumbernya serta informasi yang dihasilkan sistem beserta tujuannya.



Gambar 8. Diagram Konteks

DFD Level 1 Proses 1

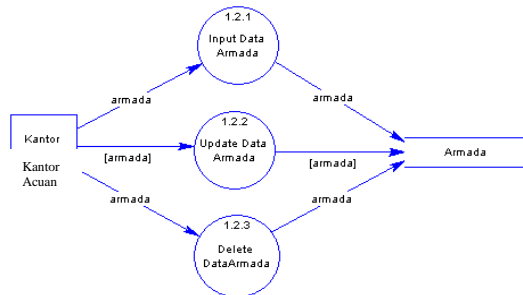
DFD level 1 proses 1 menunjukkan rekam data cabang, diawali dengan penginputan data cabang oleh Kantor Acuan. Dalam proses ini juga dilakukan proses update data cabang dan hapus data cabang. Proses tersebut kemudian diinputkan dalam *data store* cabang.



Gambar 9. DFD level 1 Proses 1

DFD Level 1 Proses 2

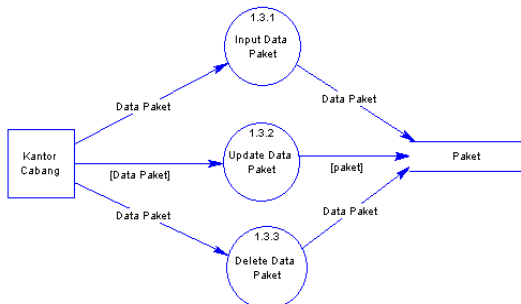
DFD Level 1 proses 2 menunjukkan rekam data armada, diawali dengan penginputan data armada oleh Kantor Acuan. Dalam proses ini juga dilakukan proses update data armada dan hapus data armada. Proses tersebut kemudian diinputkan dalam *data store* armada.



Gambar 10. DFD level 1 proses 2

DFD Level 1 Proses 3

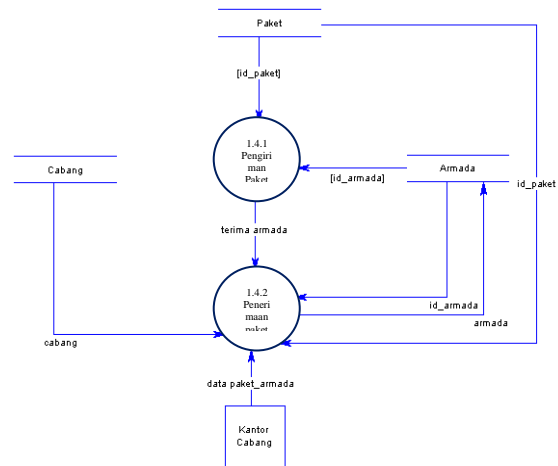
DFD Level 1 proses 3 menunjukkan rekam data paket, diawali dengan penginputan data paket oleh Kantor Cabang. Dalam proses ini juga dilakukan proses update data paket dan hapus data paket. Proses tersebut kemudian diinputkan dalam *data store* paket.



Gambar 11. DFD Level 1 proses 3

DFD Level 1 Proses 4

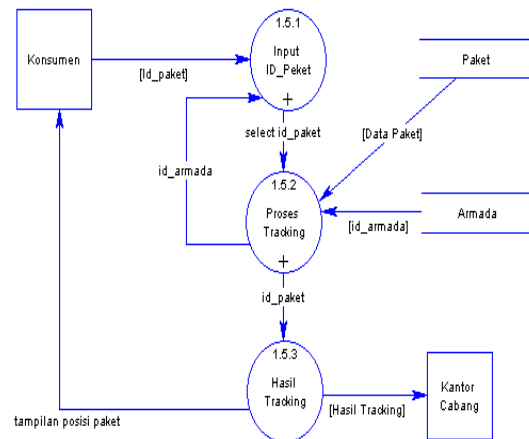
DFD Level 1 Proses 4 menunjukkan pengiriman paket. Untuk proses pengiriman, dimana aliran data terjadi dari kantor cabang ke penyimpanan data pada sistem. Proses pertama, proses pengiriman paket. Kemudian pada proses kedua yaitu penerimaan paket, membaca data dari *data store* armada, *data store* cabang serta data hasil proses pertama. Pada proses kedua juga membaca data berupa data rute dan paket dari kantor cabang.



Gambar 12. DFD level 1 Proses 4

DFD Level 1 Proses 5

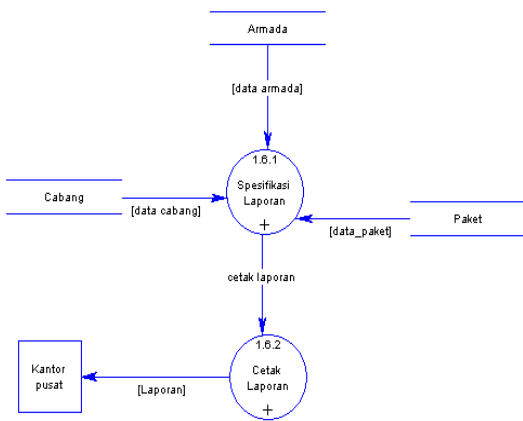
DFD Level 1 Proses 5 menunjukkan pelacakan paket. Dimana aliran data terjadi dari konsumen ke sistem, terjadi proses pengecekan jika tidak sesuai akan kembali pada status keadaan dan mengecek ID Armada, yang jika ID paket ditemukan akan dihasilkan output berupa hasil tracking atau hasil penelusuran. Hasil penelusuran diberikan kepada konsumen dan Kantor Cabang.



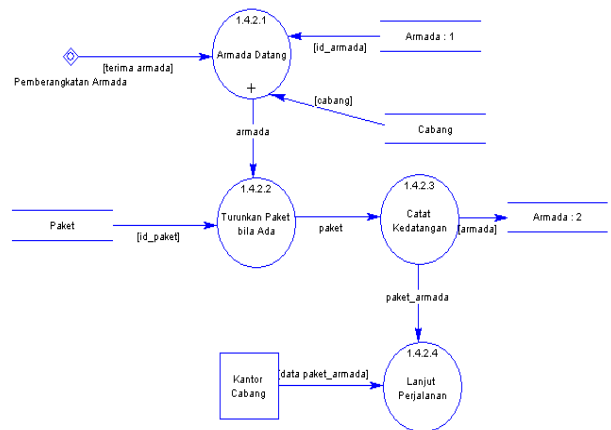
Gambar 13. DFD level 1 Proses 5

DFD Level 1 Proses 6

DFD Level 1 Proses 6 merupakan pembuatan laporan. Dimana kantor acuan hanya memberikan spesifikasi laporan yang diinginkan, kemudian sistem mengolah dan dihasilkan suatu laporan. Proses spesifikasi laporan membaca data dari *data store* armada dan *data store* rute.



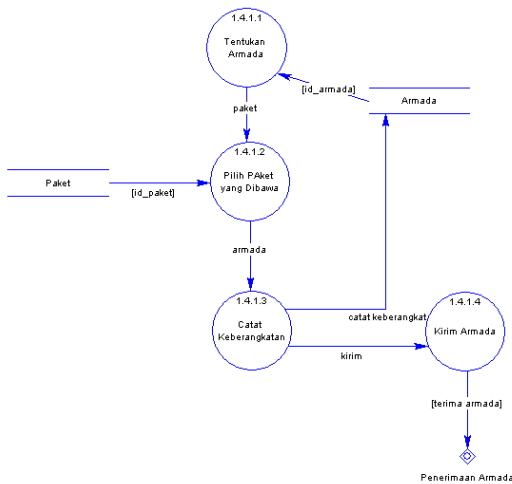
Gambar 14. DFD Level 1 proses 6



Gambar 16. DFD Level 2 proses 4.2

DFD Level 2 Proses 4.1

DFD level 2 menunjukkan proses pemberangkatan armada yang merupakan pengembangan dari DFD level 1 proses 4 pengiriman Paket. Sistem akan menampilkan daftar rute yang akan dilalui armada.



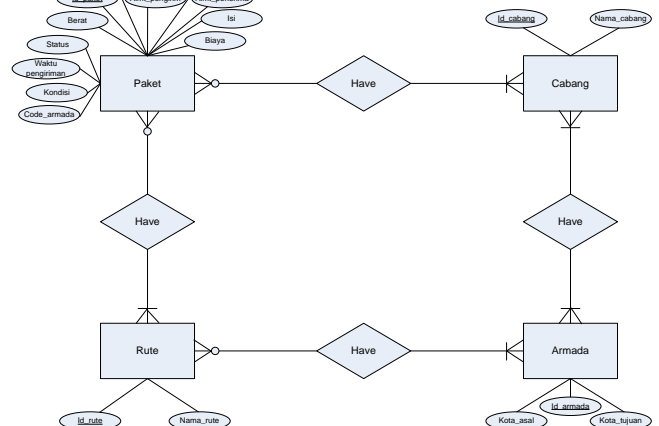
Gambar 15. DFD Level 2 proses 4.1

DFD Level 2 Proses 4.2

DFD level 2 Proses 4.2 menunjukkan dari pengembangan proses 4 penerimaan armada. Sistem akan menampilkan daftar armada yang menuju suatu cabang. Ketika ada penambahan atau pengurangan paket, sistem akan mencatatnya.

Entity Relationship Diagram (ERD)

ERD adalah model konseptual yang mendeskripsikan hubungan antar penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data yang relatif kompleks. Pada aplikasi sistem penelusuran rute terpendek ini, perancangan ERD-nya sebagaimana terlihat pada gambar berikut ini.



Gambar 17. Entity Relationship Diagram

Perhitungan Matriks Ketetangaan Graf

Matriks ketetangaan dari graf diatas adalah:

Tabel 1. Matriks ketetanggan perhitungan rute optimal untuk graf kantor pos di Aceh:

	FA	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	
A	0	40				56															
B		0	60																		
C			0	40												140					
D				0	30																
E					0																
F						0	106						70								
G							0	102	112					211							
H								0	80	80					140						
I									0	156											
J										0	156										
K											0	40									
L												0	160			162					
M													0	45		45					
N														0	140						
O															0	106					
P																0					160
Q																	0				74
R																		0			200
S																			0		80
T																					0

Rute Terpendek Menggunakan Algoritma TSP

Table 2. Perhitungan Rute optimal dari simpul awal a = A (Lsm) kesemua simpul lainnya. (Untuk Nilai S):

Lelatan yang dipilih	Simpul Lelatan	S																			
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
Initial	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	A	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	AB	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	AF	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	ABC	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	AK	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F	ABCD	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	AFG	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	AFKL	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
I	ABCDE	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J	ABCO	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
K	AFGH	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L	AFKLM	1	0	0	0	1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
M	AFGHJ	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N	ABCOP	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
O	AFKLMN	1	0	0	0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
P	AFKLMJ	1	0	0	0	1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
Q	AFKLMNQ	1	0	0	0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
R	ABCOPS	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
S	ABCOPST	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1
T	ABCOPSR	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0

Table 3. Perhitungan Rute optimal dari simpul awal a = A (Lsm) kesemua simpul lainnya. (Untuk Nilai D)

Lelatan yang dipilih	Simpul Lelatan	D																				
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	
Initial	-	0	40				56															
A	A	1	0				56															
B	AB	1	1	0			56															
C	AF	1	0	100			56															
D	ABC	1	1	40	140		56	162			125											
E	AK	1	0	100	140		56	162			125											
F	ABCD	1	1	40	100	140		56	162		125	166										
G	AFG	1	0	100	140	170		56	162	274		125	166									
H	AFKL	1	0	100	140	170		56	162	274		125	166	326								
I	ABCDE	1	1	40	100	140	170		56	162	274		125	166	326	371	328	434				
J	AFKLO	1	0	100	140	170		56	162	274		125	166	326	371	328	434	452				
K	AFGH	1	0	100	140	170		56	162	274	334		125	166	326	371	328	434				
L	AFKLM	1	0	100	140	170		56	162	274	334	415		125	166	326	371	328	434			
M	AFGHJ	1	0	100	140	170		56	162	274	334	415	125		166	326	371	328	434			
N	AFKLMN	1	0	100	140	170		56	162	274	334	415	125	166		326	371	328	434	452		
O	AFKLOP	1	0	100	140	170		56	162	274	334	415	125	166	326		371	328	434	452		
P	AFKLMJ	1	0	100	140	170		56	162	274	334	415	125	166	326	371		328	434	452		
Q	AFKLMNQ	1	0	100	140	170		56	162	274	334	415	125	166	326	371	328		434	452		
R	AFKLOPS	1	0	100	140	170		56	162	274	334	415	125	166	326	371	328	434		452		
S	AFKLOPST	1	0	100	140	170		56	162	274	334	415	125	166	326	371	328	434	452		794	594
T	AFKLOPSR	1	0	100	140	170		56	162	274	334	415	125	166	326	371	328	434	452	794		594

Dari perhitungan diatas maka Rute optimal antaran paket pada kantor pos Lhokseumawe ke kantor pos tujuan adalah sebagai berikut :

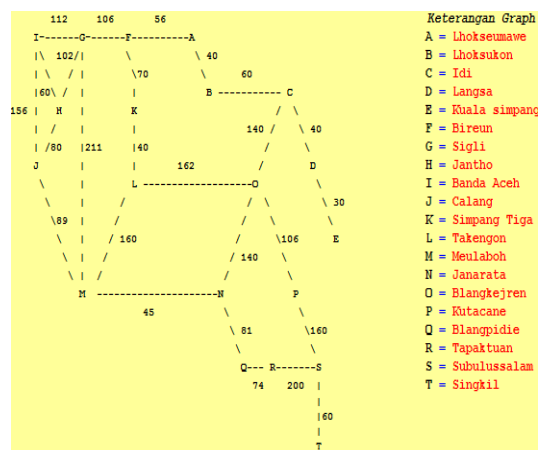
Tabel 4. Lintasan optimal dari simpul asal ke simpul tujuan:

Simpul Asal	Simpul Tujuan	Lintasan	Jarak
A (Lsm)	B (Lhoksukon)	A,B	40 Km
	F (Bireun)	A,F	56 Km
	C (Idi)	A,B,C	100 Km
	K (Simpang Tiga)	A,F,K	126 Km
	D (Langsa)	A,B,C,D	140 Km
	G (Sigli)	A,F,G	162 Km
	L (Takengon)	A,F,K,L	166 Km
	E (Kuala Simpang)	A,B,C,D,E	170 Km
	O (Blangkejren)	A,F,K,L,O	240 Km
	H (Jantho)	A,F,G,H	274 Km
	M (Meulaboh)	A,F,K,L,M	326 Km
	I (Banda Aceh)	A,F,G,H,I	334 Km
	N (Janarata)	A,F,K,L,M,N	371 Km
	P (Kutacane)	A,F,K,L,O,P	434 Km
	J (Calang)	A,F,K,L,M,J	415 Km
	Q (Blang Pidie)	A,F,K,L,M,N,Q	452 Km
	S (Subulussalam)	A,F,K,L,O,P,S	594 Km
	T (Singkil)	A,F,K,L,O,P,S,T	654 Km
	R (Tapak tuan)	A,F,K,L,O,P,S,R	794 Km

Implementasi Sistem Pengumpulan Implementasi Sistem Halaman Index Sistem Tracking



Gambar 18. Halaman utama antar muka



Node awal : A
 Node Akhir : L cari
 Jalur terpendeknya adalah :
 A -> F -> K -> L -> M -> N -> M -> L -> K -> F -> A
 Dengan total jarak 371 (742) Kilometer

Gambar 19. Hasil dari pencarian rute terpendek menggunakan algoritma TSP

Gambar diatas menunjukkan pencarian rute terpendek dari *node* asal A ke *node* tujuan N. Proses pencarian rute terpendek dari *node* asal A ke *node* akhir N menggunakan algoritma TSP melalui proses penentuan titik ke titik yang terdekat berdasarkan bobot jarak. Algoritma TSP akan mencari semua lintasan yang mungkin dilewati menuju titik akhir untuk kemudian ditentukan lintasan terpendeknya. Dan lintasan terpendek dari *node* asal A ke *node* akhir N dapat melewati A – F – K– L – M - N dengan total jarak 371 Km. Pencarian rute terpendek antaran paket dari kota asal (Lhokseumawe) ke semua kota tujuan dapat digambarkan kedalam peta. Adapun tampilannya dapat dilihat pada gambar berikut :

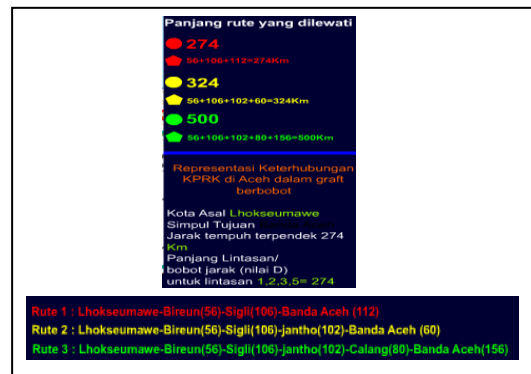


Gambar 20. Tampilan peta rute sebelum di proses



Gambar 21. Tampilan peta rute proses menuju kota tujuan

Setelah proses menuju kota tujuan selesai maka tampilan laporan dapat dilihat pada gambar berikut :



Gambar 22. Tampilan laporan dari rute yang dipilih

KESIMPULAN

Setelah membuat aplikasi sistem *tracking* antaran paket dengan menggunakan Algoritma TSP pada PT. Pos Indonesia Persero Lhokseumawe, maka dapat diambil kesimpulan sebagai berikut:

1. Sistem ini dapat menemukan urutan kunjungan lokasi (satu lokasinya dikunjungi satu kali) yang total "nilai"-nya paling optimal (bisa minimal atau maksimal bergantung tujuannya). "Nilai," di sini bisa berupajarak, biaya, kenyamanan, dan sebagainya. tujuannya adalah menemukan urutan lokasi pengantaran paket yang total jaraknya paling minimal.
2. Aplikasi sistem *Tracking* paket ini dapat digunakan untuk meningkatkan pencarian paket dan penentuan rute dalam pengiriman paket serta mempersingkat waktu pencarian rute antaran paket secara efektif dan efisien serta menyediakan informasi yang cepat dan mudah.
3. Sistem *Tracking* paket ini menyediakan keamanan data kepada setiap kantor cabang dan juga kantor pusat yang mempunyai hak akses, yaitu dengan memberikan user ID dan password yang dapat di enkripsi.

Beberapa saran yang dapat diberikan untuk pengembangan sistem *tracking* ini adalah sebagai berikut:

1. Sistem *Tracking* paket hanya menyediakan fasilitas penentuan rute antaran paket dengan lintasan optimal, daftar tarif, ekspedisi paket, kritik saran, profile, dan berita. Untuk pengembangan sistem ini lebih lanjut, dapat ditambahkan modul-modul lain yang mendukung sistem ini
2. Dalam menentukan rute optimal algoritma TSP tidak selamanya dapat memberikan rute yang nilainya minimal, karena prinsip yang digunakan oleh algoritma TSP disini adalah semua cara dicoba untuk mencari rute yang optimal, untuk bisa mendapatkan pencarian rute optimal secara baik

kedepannya dalam pencarian rute terpendek bisa menggunakan algoritma yang lebih luas ruang lingkup kerjanya.

DAFTAR PUSTAKA

- [1] Betha, Sidik, Ir, 2002, *Pemrograman Web dengan PHP*, Penerbit Informatika, Bandung
- [2] Handoyo, Hendri Purwo, dkk, *Pemecahan Masalah Jalur Terpendek dengan Travelling SalesPerson Problem*, Jurusan Teknik Informatika Sekolah tinggi Teknologi Telkom, Bandung.
- [3] Hardi, Richki, 2007. *Sistem Ekspedisi Paket Sentral Pengolahan Pos Yogyakarta PT. Pos Indonesia (Persero) Berbasis WEB*. Skripsi S1 Universitas Ahmad Dahlan, Yogyakarta.
- [4] Hardi, Richki, 2009. *Tugas Analisa Algoritma Graph*.
- [5] Kadir, Abdul, 2008. *Dasar Pemograman Web Dinamis Menggunakan PHP*. Penerbit Andi, Yogyakarta.
- [6] Munir, Rinaldi, 2005. *Buku Teks Ilmu Komputer Matematika Diskrit Edisi Ketiga*. Penerbit Informatika, Bandung.
- [7] Nugroho, Bunafit, 2004, *Aplikasi Pemograman Web Dinamis dengan PHP dan MySQL*, Penerbit Gava Media, Yogyakarta.
- [8] Pradhana, Aditya Bayu, *Studi Dan Implementasi Persoalan Lintasan Terpendek Suatu Graf*, Program Studi Teknik Informatika, Institut Teknologi Bandung.
- [9] Rafiudin, Rahmat. 2004. *Panduan Menjadi Seorang Webmaster*. Penerbit Andi, Yogyakarta.
- [10] Setioko, Budy, *Solusi Chinese Postman Problem yang Berprinsip Greedy*. Jurusan Teknik Informatika, Sekolah Tinggi Teknologi Telkom, Bandung.
- [11] Sigit, Poncow, *Analisis dan Perancangan Sistem*, Khusus untuk kalangan sendiri.
- [12] Tanuhardja, Jeffrey, *Perencanaan Rute Perjalanan di JawaTimur dengan Dukungan GIS Menggunakan algoritma*. Jurusan Teknik Informatika, Universitas Kristen Petra.
- [13] Wahyudi, Bambang, 2004, *Pengantar Struktur Data dan Algoritma*, Andi, Yogyakarta.



Dahlan Abdullah, Menyelesaikan S1 dibidang Teknik Informatika pada Universitas Islam Indonesia Yogyakarta - Indonesia dan mendapat Gelar Sarjana Teknik (ST) pada tahun 1999. Saat ini sedang menyelesaikan Program Megister Komputer di STMIK Eresha Jakarta. Aktif melakukan Penelitian dibidang Jaringan Komputer, Database, Radio Net, Komputer Aplikasi, Robotika, Web Based Application, Sistem Informasi Manajemen dan Infrastruktur Jaringan Komputer. Pernah menduduki Jabatan sebagai Kepala UPT. Pusat Komputer (PUSKOM) Universitas Malikussaleh dari tahun 2005 sampai dengan 2011 dan Jabatan Saat ini sebagai Kepala UPT. Perpustakaan Universitas Malikussaleh Aceh-Indonesia. Aktif juga sebagai anggota dari INHERENT (Indonesian Higher Education Network), JARDIKNAS (National Education Network), dan Sekretaris Jenderal APTIKOM (Association of Computer and Information University) Provinsi Aceh, sebagai Koordinator ICTPura Provinsi Aceh, Koordinator Relawan TIK Provinsi Aceh, Koordinator E-KTP Provinsi Aceh, Wakil Ketua Palang Merah Indonesia Cabang Kota Lhokseumawe dan beberapa jabatan lainnya.



Richki Hardi, Menyelesaikan S1 di Prodi Teknik Informatika Universitas Ahmad Dahlan Yogyakarta dengan gelar Sarjana Teknik (ST) tahun 2007 dan S2 di Prodi Sistem Komputer dan Informatika (SKI) Universitas Gadjah Mada Yogyakarta dengan gelar Master of Engineering (M.Eng) tahun 2010, banyak menyelesaikan karya tulis di bidang System Prima Number of Binery code using single processor dan menjadi pembicara di beberapa seminar tentang Teknik Informatika.