

Aplikasi Graf Dalam Jasa Pengiriman Barang

Melvin Fonda / 13512085

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13512085@std.stei.itb.ac.id

ABSTRAK

Tujuan dari penulisan makalah ini adalah untuk menunjukkan aplikasi dari graf yang merupakan salah satu bidang dalam ilmu informatika. Aplikasi graf yang akan dibahas dalam makalah ini adalah aplikasi graf tentang jasa pengiriman barang yang digunakan hampir oleh seluruh jasa pengiriman di seluruh dunia pada saat ini. Dalam makalah ini akan digunakan berbagai algoritma dalam graf dan bagaimana algoritma – algoritma tersebut dapat digunakan dalam jasa pengiriman barang itu sendiri. Algoritma yang dimaksud antara lain Ant Colony System, algoritma Prim, dan Algoritma Kruskal yang di dalamnya terdapat pemanfaatan teori graf. Graf sangat tepat dalam pemanfaatan ini karena dalam jasa pengiriman pengiriman sendiri sangat lekat dengan salah satu persoalan dalam graf yaitu Travelling Salesperson Problem. Dengan pemanfaatan berbagai algoritma dalam teori graf, diyakini bahwa pengeluaran yang dikeluarkan untuk mengantarkan sebuah barang akan dapat diminimalisir dan waktu yang dibutuhkan untuk mengirim suatu barang akan menjadi lebih singkat. Dengan melakukan meminimalisir pada kedua hal tersebut, jasa akan semakin dipercaya oleh konsumen sehingga profit akan meningkat.

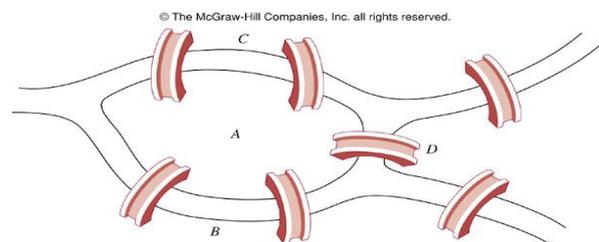
Kata kunci – Graf, Ant Colony System, Prim, Kruskal, Algoritma, Travelling Salesperson Problem, profit.

I. PENDAHULUAN

Dalam dunia sekarang ini, dengan berbagai kesibukan kita ingin melakukan segala sesuatunya dengan cara yang singkat, cepat, dan mudah. Dalam membeli barang melalui internet misalnya. Atau misalnya mengirim suatu barang ke kota atau negara lain. Segala hal ini berhubungan dengan jasa pengiriman barang. Misalnya kita membeli suatu handphone dari sebuah situs perbelanjaan seperti www.amazon.com, maka handphone tersebut akan dikirim ke rumah kita sebagai tujuan dengan melewati beberapa tempat terlebih dahulu. Rute yang dilalui oleh kurir pengirim barang tersebut haruslah rute yang tersingkat. Hal ini bertujuan untuk mempersingkat waktu dan menghemat biaya pengiriman benda tersebut. Secara sadar atau tidak sadar, banyak

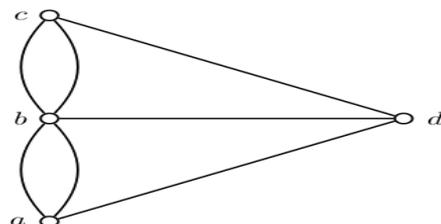
permasalahan yang dapat kita selesaikan dengan menggunakan teori graf.

Apa itu graf? Graf adalah suatu representasi dari beberapa objek diskrit dimana objek-objek tersebut saling terhubung. Graf ditemukan oleh seorang matematikawan dan fisikawan yang bernama Leonhard Euler. Euler adalah seorang matematikawan terkenal pada sekitar abad ke -18 dan salah satu matematikawan paling hebat yang pernah ada. Königsberg adalah sebuah kota di Prussia yang terletak di di atas sungai Pregel, yang merupakan tempat tinggal oleh bangsawan pada sekitar abad ke 16th. (Sekarang kotanya bernama Kaliningrad). Sungai pregel mengalir di kota tersebut dan membuat sebuah pulau.



Gambar 1.1 Jembatan Königsberg

Ada sebuah permasalahan yang terkenal tentang Königsberg yang dikaji oleh Euler. Apakah mungkin untuk berjalan menyusuri kota dan kembali ke tempat semula dengan suatu cara tertentu sehingga hanya melewati setiap jembatan dengan satu kali saja? Euler akhirnya menemukan suatu solusi untuk masalah jembatan Königsberg ini. Euler menggambarkan jembatan ini dalam bentuk graf, garis mempresentasikan jembatan dan simpul mempresentasikan daratan. Setelah melakukan analisis, Euler membuktikan bahwa adalah tidak mungkin untuk melewati setiap jembatan hanya satu kali dan kembali ke tempat awal jika simpulnya berjumlah genap.



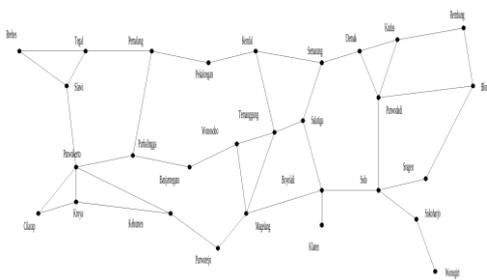
Gambar 1.2 Graf Euler mempresentasikan jembatan Königsberg

Gambar 1.2 menunjukkan bahwa derajat tiap simpul adalah ganjil (Setiap simpul memiliki jumlah sisi ganjil) sehingga tidak mungkin bagi seseorang untuk pergi dari suatu tempat melewati setiap jembatan hanya sekali dan kembali ke tempat asal. Dari Teori Euler inilah asal mula teori graf mulai dimanfaatkan dan diaplikasikan di berbagai bidang dalam kehidupan.

II. TEORI DASAR

2.1 Graph

Teori Graf merupakan salah satu cabang kajian dalam ilmu matematika dan ilmu komputer. Graf digunakan untuk memvisualisasikan objek yang bersifat diskrit dan hubungan antara objek diskrit tersebut. Titik titik dalam graf dapat terhubung atau tidak. Salah satu contoh graf adalah peta.

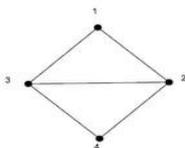


Gambar 2.1 Contoh Graf dalam kehidupan sehari – hari (peta).

Graf terdiri dari beberapa komponen yaitu simpul (vertex atau node) yang dihubungkan oleh sisi (edge) atau busur (arc). Graf biasa divisualisasikan sebagai simpul yang dihubungkan oleh garis (melambangkan sisi) atau garis berarah (busur). Sisi dapat dihubungkan dengan suatu sisi yang sama sehingga dinamakan gelang (loop).

Graph

Graph G_1



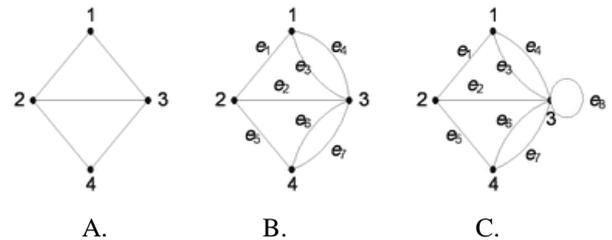
G_1 adalah graph dengan
 $V = \{ 1, 2, 3, 4 \}$
 $E = \{ (1, 2), (1, 3), (2, 3), (2, 4), (3, 4) \}$

Gambar 2.2 Graf dan bagian –bagiannya : Simpul (V) dan Sisi (E).

Berdarkan ada tidaknya gelang (circle) atau sisi ganda pada suatu graf maka graf digolongkan menjadi 2 jenis :

1. Graf sederhana (Simple Graph)
 Graf yang tidak memiliki gelang ataupun sisi ganda disebut graf sederhana.

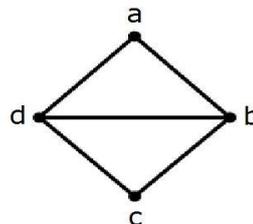
2. Graf tak-sederhana (unsimple – graph)
 Graf tak sederhana adalah lawan dari graf sederhana yaitu graf yang memiliki sisi ganda atau gelang. Graf tak –sederhana terbagi lagi menjadi 2 jenis, yaitu graf ganda (multigraph) atau graf semu (pseudograph). Graf ganda adalah graf yang mengandung sisi ganda. Graf semu adalah graf yang mengandung gelang (loop).



Gambar 2.3 a). Graf sederhana, b,c) Graf tak-sederhana

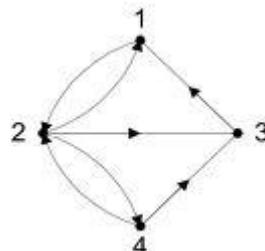
Berdasarkan orientasi arahnya terhadap sisi, secara general graf dapat dibedakan menjadi 2 jenis :

1. Graf tak-berarah (undirected graph)
 Graf yang sisinya tidak memiliki orientasi arah biasa disebut graf tak-berarah. Pada graf tak berarah, urutan dari pasangan simpul yang dihubungkan tidak diperhitungkan. Misalnya : (1,2) dan (2,1) adalah sisi yang sama



Gambar 2.4 graf tak berarah

2. Graf berarah (directed graph atau digraph)
 Sebuah graf yang memiliki orientasi arah (dilambangkan dengan tanda panah) biasa disebut sebagai graf berarah. Berbeda dengan graf tak-berarah, graf berarah memperhatikan urutan dari pasangan simpul. Setiap sisi yang memiliki arah disebut busur. Misalnya (1,2) dan (2,1) adalah dua buah busur yang berbeda. Untuk busur (1,2) simpul 1 disebut simpul asal dan simpul 2 disebut simpul terminal.



Gambar 2.5 Graf berarah

Beberapa Istilah (terminologi) yang penting dalam penggunaan graf :

- a. **Ketetanggan (Adjacent)**
Dua buah simpul dikatakan adjacent apabila kedua simpul tersebut terhubung langsung oleh sisi tertentu.
- b. **Bersisian (Incidency)**
Sebuah sisi dikatakan bersisian dengan sebuah simpul tertentu apabila sisi terhubung langsung dengan simpul tersebut.
- c. **Simpul Terpencil (Isolated Vertex)**
Simpul terpencil adalah sebuah simpul yang tidak terhubung dengan sisi manapun sehingga tidak memiliki sisi yang bersisian dengannya.
- d. **Graf Kosong (Null Graph atau Empty Graph)**
Graf yang himpunan sisinya merupakan himpunan kosong.
- e. **Derajat (degree)**
Jumlah sisi terhubung langsung dengan simpul tertentu.
- f. **Lintasan (Path)**
Lintasan adalah urutan dari simpul dan sisi yang dimulai dari simpul pertama dan berakhir di simpul yang kedua dan setiap sisi bersisian dengan kedua simpul (sebelum dan setelahnya).
- g. **Siklus (Cycle) atau sirkuit (Circuit)**
Siklus adalah lintasan yang memiliki simpul awal dan simpul akhir yang sama.
- h. **Terhubung (Connected)**
Dua buah simpul v_1 dan v_2 disebut terhubung jika terdapat lintasan (path) dari v_1 ke v_2
- i. **Upagraf (Subgraph)**
Upagraf adalah graf yang sisi dan simpulnya merupakan anggota dari sisi dan simpul dari graf lain.
- j. **Graf Berbobot (Weighted Graph)**
Graf berbobot adalah graf yang sisi-sisinya memiliki sebuah nilai (bobot).
- k. **Lintasan Euler dan Sirkuit Euler**
Lintasan Euler adalah lintasan yang melalui masing-masing sisi di graf tepat satu kali. Sedangkan sirkuit Euler adalah sirkuit yang melalui setiap sisi dari graf hanya satu kali.
- l. **Lintasan Hamilton dan Sirkuit Hamilton**
Lintasan hamilton adalah lintasan yang melalui setiap vertex di dalam graf tepat satu kali. Sedangkan sirkuit hamilton adalah sirkuit yang melalui setiap sisi dari graf hanya satu kali.

Beberapa aplikasi dari graf adalah :

A. Persoalan Pedagang Keliling (Travelling Salesperson Problem)

Diberikan beberapa kota yang memiliki suatu jarak tertentu antarkota. Bagaimana cara kita untuk menentukan rute terpendek yang harus dilalui oleh seorang pedagang bila pedagang itu berangkat dari sebuah kota asal dan menyinggahi setiap kota tepat satu kali dan kembali lagi ke kota asal keberangkatan. Persoalan ini dapat diselesaikan dengan menentukan sebuah sirkuit hamilton yang memiliki bobot minimum.

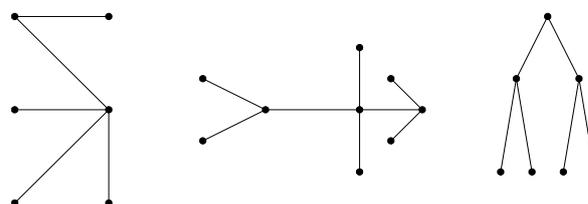
B. Persoalan Tukang Pos Cina (Chinese Postman Problem)

Persoalan ini dikemukakan pada tahun 1962 oleh seorang berkebangsaan Cina bernama Mei Ku-Kuan. Persoalan Chinese Postman Problem ini merupakan persoalan tentang seorang tukang pos yang akan mengantar surat ke alamat-alamat sepanjang jalan di suatu daerah. Bagaimana ia merencanakan rute perjalanannya supaya ia melewati setiap jalan tepat sekali dan kembali lagi ke tempat awal keberangkatan ? Cara yang paling mungkin adalah dengan menggunakan sirkuit Euler dalam teori graf.

2.2 Pohon

Pohon (Tree) dalam istilah komputer atau matematik adalah suatu graf terhubung yang tidak memiliki sirkuit.

Gambar 2.6 Contoh Pohon



Hutan adalah kumpulan pohon yang tidak berhubungan. Pada gambar 2.6 ketiga pohon tersebut membentuk sebuah hutan.

2.2.1 Pohon Merentang (Spanning Tree)

Pohon disebut pohon merentang dari sebuah graf G jika mengandung semua simpul dari G dan merupakan upagraf dari G . Pohon merentang yang memiliki bobot minimum disebut pohon merentang minimum. Dalam penentuan pohon merentang minimum dapat digunakan beberapa algoritma di antaranya adalah :

1. Algoritma Prim

Algoritma prim adalah algoritma untuk mencari minimum spanning tree dalam sebuah graf terhubung berbobot tak berarah. Algoritma Prim

ditemukan tahun 1930 oleh matematikawan Czech Vojtech Jarnik dan di adaptasi oleh Informatikawan Robert C. Prim pada tahun 1957. Secara sederhana Algoritma Prim menggunakan 3 langkah berikut :

- Pilih sebuah vertex yang memiliki nilai terkecil
- Pilih sebuah sisi yang terhubung dengan vertex yang belum ada di dalam pohon lalu tambahkan sisi tersebut ke dalam pohon tersebut
- Ulangi langkah 2 (sampai semua simpul ada di dalam pohon)

```

procedure Prim(input G : graf,
output T : pohon)
{ Membentuk pohon merentang
minimum T dari graf terhubung-
berbobot G.
Masukan: graf-berbobot terhubung
G = (V, E), dengan  $|V| = n$ 
Keluaran: pohon rentang minimum T
= (V, E')
}

```

Deklarasi

i, p, q, u, v : integer

Algoritma

Cari sisi (p,q) dari E yang berbobot terkecil

T ← {(p,q)}

for i ← 1 to n-2 do

Pilih sisi (u,v) dari E yang bobotnya terkecil namun

bersisian dengan simpul di T

T ← T ∪ {(u,v)}

endfor

2. Algoritma Kruskal

Algoritma Kruskal adalah Algoritma yang dipakai dalam teori graf untuk mencari minimum spanning tree untuk graf terhubung berbobot. Beberapa Langkah algoritma Kruskal adalah :

- Urutkan graf berdasarkan bobot dari yang terkecil ke bobot yang terbesar
- Pilih sisi(u,v) dengan bobot minimum yang tidak membentuk sirkuit di T lalu tambahkan (u,v) ke dalam T.
- Ulangi langkah 2 sebanyak N-1 kali

```

procedure Kruskal(input G :
graf, output T : pohon)
{ Membentuk pohon merentang
minimum T dari graf
terhubung -berbobot G.
Masukan: graf-berbobot
terhubung G = (V, E), dengan
 $|V| = n$ 
Keluaran: pohon rentang
minimum T = (V, E')
}

```

Deklarasi

i, p, q, u, v : integer

Algoritma

(Asumsi: sisi-sisi dari graf sudah diurut menaik berdasarkan bobotnya - dari bobot kecil ke bobot besar)

T ← {}

while jumlah sisi T < n-1 do

Pilih sisi (u,v) dari E yang bobotnya terkecil

if (u,v) tidak membentuk siklus di T then

T ← T ∪ {(u,v)}

endif

endfor

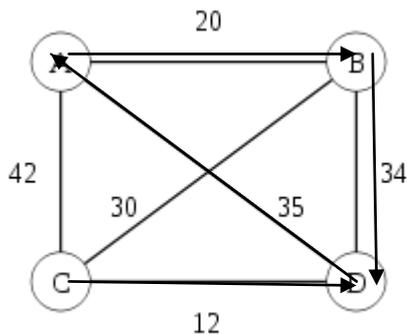
III. ANALISIS

Kebanyakan orang di dunia pada saat ini memiliki kesibukan yang sangat amat. Seperti contohnya seorang karyawan perusahaan masuk kerja jam 7 pagi dan pulang jam 6 sore. Mahasiswa yang dipenuhi berbagai tugas kuliah dan ujian. Hal ini menimbulkan sebuah pola pikir "Time is money". Melihat hal ini, banyak perusahaan yang mencoba mengambil profit dari kesibukan orang-orang ini. Seperti yang telah saya sebutkan tadi www.amazon.com misalnya. Anda tidak perlu keluar dari rumah dan macet-macetan untuk hanya membeli sebuah barang. Anda dapat tinggal duduk di dalam rumah, membuka laptop anda dan mengklik barang yang anda mau. Atau mungkin dalam contoh lain Perusahaan FedEx. Perusahaan ini mengambil untung dengan memberikan jasa pengiriman barang. Bayangkan jika anda ingin mengirim sebuah barang kepada saudara anda yang berada ribuan kilometer dari rumah anda. Dengan FedEx anda dapat mengirim barang tersebut sampai ke tujuan dengan aman serta waktu yang singkat.

Melihat dari hal ini muncul suatu pertanyaan. Bagaimanakah sebuah perusahaan FedEx atau Amazon mengirim barang tersebut? Setiap kota yang dilewati oleh kurir tersebut dapat kita visualisasikan sebagai simpul dari sebuah graf, dan rute yang dilewati dapat kita gambarkan sebagai suatu garis dalam sebuah graf. Mungkin jika dalam sebuah permasalahan yang sederhana seperti mengirim sejumlah 10 barang, yang lokasinya berjauhan efisiensi waktu dan jarak dari pengiriman tidak perlu kita perhitungkan. Tetapi bayangkan jika jasa pengiriman tersebut melayani permintaan pengiriman dari seluruh

dunia seperti amazon. Dalam hal ini, pemilihan rute sangatlah penting. Semakin singkat rute yang diberikan maka akan sangat besar penghematan yang dapat dilakukan dalam hal bahan bakar dari transportasi yang digunakan untuk mengirim barang. Efisiensi waktu dari sangat penting karena konsumen dalam hal ini memiliki posisi sebagai seorang “raja” sehingga jika dibutuhkan waktu yang singkat untuk mendapatkan suatu barang yang dipesan konsumen akan merasa puas, tetapi sebaliknya jika waktu yang dibutuhkan sangatlah banyak maka konsumen akan berpikir 2 kali untuk kembali menggunakan jasa tersebut.

Rute yang diambil seorang kurir yang mengantarkan barang tersebut mirip dengan Persoalan Pedagang Keliling (Travelling Salesperson Problem). Travelling Salesperson Problem adalah cara kita untuk menentukan rute terpendek yang harus dilalui oleh seorang pedagang bila pedagang itu berangkat dari sebuah kota asal dan menyinggahi setiap kota tepat satu kali dan kembali lagi ke kota asal keberangkatan. Persoalan ini dapat diselesaikan dengan menentukan sebuah sirkuit hamilton yang memiliki bobot minimum. Persoalan ini pertama kali diuji pada 1930 dan merupakan salah satu optimasi yang paling dicari di dunia. Walaupun sangat sulit untuk menghitung jarak terpendek karena sangat banyak kemungkinan yang dapat terjadi, tetapi dengan suatu metode tertentu sekalipun ribuan kota yang dihitung akan kita dapatkan jarak terpendek tersebut.

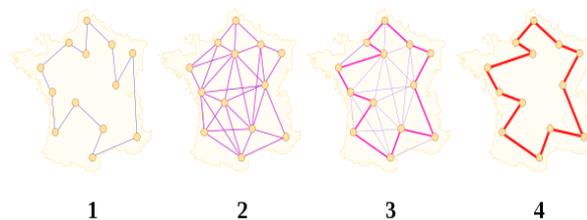


Gambar 3.1 Contoh Graf berbobot

Misalnya dalam gambar di atas, seorang kurir akan mencoba berbagai jarak. Misalkan vertex dalam graf diatas dilambangkan sebagai kota dan sisi dilambangkan sebagai jalan yang dilewati. Dalam kasus ini kita misalkan perjalanan kurir dinilai dari kota C. Dari kota C kurir akan menempuh jarak sejauh 12 satuan ke kota D, lalu menempuh jarak 35 satuan ke kota A. Kemudian kurir akan menuju kota B dengan jarak 20 satuan, lalu kurir akan menuju kota D dengan 34 satuan sebelum akhirnya kembali ke kota C dengan jarak 12 satuan. Solusi ini bukan satu-satunya solusi yang dapat ditemukan dari persoalan ini. Ini adalah persoalan tentang bagaimana cara kita meminimalisir sebuah persoalan dengan memulai di suatu vertex, mengunjungi tiap tiap vertex hanya sekali, lalu kembali ke vertex yang sama.

Tahun 1997 seorang ahli dalam bidang artificial

intelligence Marco Dorigo menemukan suatu metode untuk Travelling Salesperson Problem ini dengan simulasi oleh banyak koloni semut. Dengan simulasi ini Marco memodelkan kebiasaan seekor semut untuk mencari jalur terpendek antara sumber makanan menuju sarang mereka. Para semut mengikuti jejak yang telah ditinggalkan oleh semut lainnya. Ant Colony System ini mencoba berbagai kemungkinan dengan mengirimkan semut semut virtual untuk mengeksplorasi berbagai rute dalam peta. Ternyata dari percobaan ini setiap semut memilih kota setelahnya untuk dikunjungi berdasarkan jarak tiap kota lalu menyimpan pheromone (suatu zat yang dikeluarkan semut sebagai jejak untuk semut lainnya) pada setiap jalur yang mereka lewati. Setelah melalui setiap kemungkinan rute, pada kondisi ini semut yang telah menyelesaikan rute terpendek telah menyimpan pheromone pada rute yang ia selesaikan. Jumlah dari pheromone yang disimpan berbanding terbalik dengan panjang rute. Semakin pendek rute yang dilewati, semakin banyak semut tersebut menyimpan pheromone. Dalam simulasi ini jika kita bandingkan dengan kurir FedEx ataupun amazon. Maka akan kita gambarkan kurir sebagai seekor semut. Dengan menjelajah semua kota dan memberi tanda pada kota yang dijelajahi maka kurir tersebut akan menemukan suatu rute terpendek untuk mengantar barang tersebut.



Gambar 3.2 Gambar Graf Ant Colony System

Cara lain untuk mengantar barang adalah dengan “Minimum Spanning Tree”. Misalnya seorang kurir ingin mengantarkan suatu barang ke kota A menuju kota F dengan terlebih dahulu melalui kota B,C,D, dan E. Dengan menggunakan algoritma Prim atau algoritma Kruskal kita dapat menentukan jarak minimum dari kota A menuju kota F. Minimum spanning tree meminimalisir jarak yang ditempuh dari kota A menuju kota F dengan memilih jarak terpendek dari tiap kota. Setelah kurir mengantarkan suatu barang dari kota A menuju kota F, saat tiba di kota F ia akan diberikan barang yang perlu diantarkan lagi misalnya menuju kota G, H, I, dan J sebelum akhirnya sampai di kota asal yaitu kota A. Saat kembali mengantarkan barang ke kota G, H, I, J akan kembali digunakan spanning tree sehingga dapat meminimalisir jarak tempuh.

Beberapa algoritma yang kita pikir mungkin tidak terlalu sulit tetapi juga dimanfaatkan oleh perusahaan jasa FedEx. Misalnya dalam mengantar barang di Amerika, sebuah mobil pengantar barang berusaha mencari jalan dimana jalan tersebut ia tidak akan menemukan lampu merah. Bila ia terpaksa harus bertemu lampu merah, maka

untuk melewati lampu merah tersebut ia akan berbelok ke kanan (acuan jalan di Amerika di sebelah kanan dan boleh langsung belok kanan) dan mencari jalan untuk sampai ke tujuan sehingga mobil tersebut tidak akan bertemu lampu merah dan hal ini akan menghemat waktu dari pengiriman barang.

IV. KESIMPULAN

Dalam mencari jarak terpendek untuk mengirim suatu barang dapat digunakan berbagai cara yang telah disebutkan seperti Ant Colony Sistem, ataupun minimum spanning Tree. Setiap jasa pengiriman di seluruh dunia pasti akan mencari sebuah metode yang dapat menambah efisiensi waktu dan menghemat bahan bakar pengiriman sehingga profit dapat dimaksimalkan tanpa menghilangkan rasa percaya dari konsumen. Bayangkan jika metode yang digunakan dapat menghemat waktu 1 menit saja untuk tiap kota yang dilalui. Coba anda kalikan dengan jumlah pembeli di amazon atau jumlah pengirim di FedEx. Graf secara nyata dapat menghasilkan sebuah visualisasi dari pengiriman ini sehingga pengiriman akan semakin mudah untuk di analisis sehingga dari data yang ada akan makin banyak rute yang dapat dibuat. Probabilitas untuk menemukan suatu rute yang dianggap terpendek pun akan dapat ditemukan. Dan hingga sampai sekarang berbagai metode dan cara pun masi terus dicari untuk mengoptimalisasi hal ini.

V. ACKNOWLEDGEMENT

Ucapan terima kasih sebesar-besarnya saya tujukan kepada dosen Informatika saya Bapak Rinaldi Munir.atas bimbingan di dalam maupun luar kelas. Berbagai referensi saya dapatkan dari kuliah ataupun kegiatan di luar kuliah sehingga saya dapat menyelesaikan makalah ini. Saya juga berterima kasih kepada teman – teman dan pihak lain yang telah membantu saya dalam proses pembuatan makalah ini.

REFERENSI

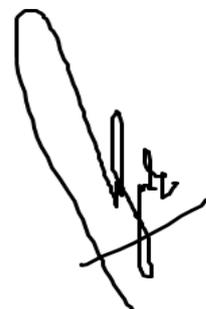
- [1] Rinaldi Munir (2013), Diktat Matematika Diskrit : Graf , Pohon.
Tanggal /Waktu akses : 14-12-2013/15.00
- [2] [Golumbic, Martin](#) (1980), *Algorithmic Graph Theory and Perfect Graphs*, [Academic Press](#).
Tanggal /Waktu akses : 14-12-2013/17.25
- [3] <http://darkrabbitblog.blogspot.com/2011/12/teori-graf-ii.html>
Tanggal /Waktu akses : 14-12-2013/18.35
- [4] <http://dc126.4shared.com/doc/hiCP4IW8/preview.html>
Tanggal /Waktu akses : 14-12-2013/19.25
- [5] <http://danyatriokintoko.blogspot.com/2013/02/jenis-jenis-graf.html>
Tanggal /Waktu akses : 14-12-2013/20.00
- [6] <http://www.cse.ust.hk/~dekai/271/notes/L07/L07.pdf>
Tanggal /Waktu akses : 14-12-2013/21.00

- [7] [Thomas H. Cormen](#), [Charles E. Leiserson](#), [Ronald L. Rivest](#), and [Clifford Stein](#). *Introduction to Algorithms*, Third Edition. MIT Press, 2009. [ISBN 0-262-03384-4](#). Section 23.2: The algorithms of Kruskal and Prim, pp. 631–638.
Tanggal /Waktu akses : 14-12-2013/22.25
- [8] Papadimitriou, C. H.; Yannakakis, M. (1993), "The traveling salesman problem with distances one and two", *Math. Oper. Res.* **18**: 1–11,
Tanggal /Waktu akses : 14-12-2013/23.00
- [9] <http://bridges-2013.blogspot.com/2013/01/bridges-of-konigsberg.html>
Tanggal /Waktu akses : 14-12-2013/23.20

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 14 Desember 2013



Melvin Fonda /13512085