

Pemodelan Sistem Lalu Lintas dengan Graf Ganda Berarah Berbobot

Teddy Pandu Wirawan

Jurusan Teknik Informatika ITB, Bandung 40132, email: t_pandu09@students.itb.ac.id

Abstrak

Makalah ini membahas penerapan teori graf dalam pemodelan sistem lalu lintas kota. Graf yang akan digunakan untuk pemodelan sistem lalu lintas kota adalah graf ganda berarah berbobot.

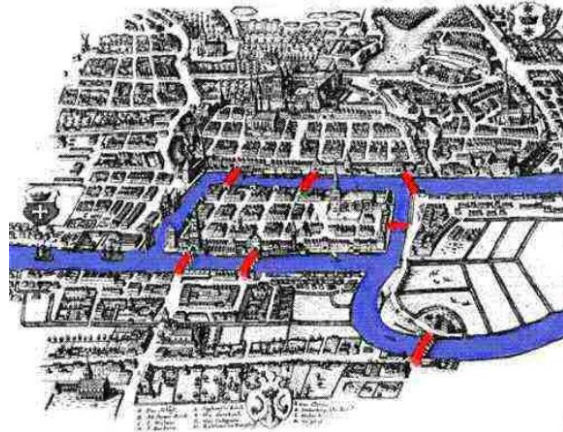
Penggunaan graf ganda berarah berbobot ini dimaksudkan untuk memperbaiki pemodelan yang ada selama ini. Biasanya pemodelan sistem lalu lintas dengan graf hanya menggunakan graf sederhana dimana setiap simpul melambangkan suatu persimpangan dan sisi melambangkan jalan.

Pada pemodelan menggunakan graf ganda berarah berbobot, setiap factor pendukung juga diperhitungkan, seperti (1) Lebar jalan, dan (2) Banyaknya kendaraan yang lewat per satuan waktu.

Kata Kunci: graf ganda, graf ganda berarah, graf berarah, graf berbobot, jalan, volume kendaraan.

1. PENDAHULUAN

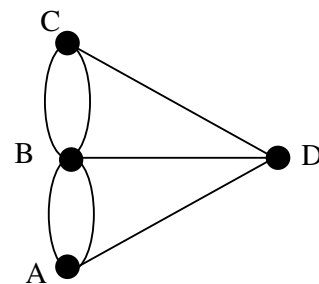
Menurut catatan sejarah, teori graf pertama kali digunakan oleh seorang ahli matematika dari Swiss yang bernama Euler untuk merepresentasikan Jembatan Königsberg, dan menyelesaikan permasalahan jembatan tersebut. Königsberg adalah sebuah kota di sebelah timur Prussia (Jerman sekarang) dimana terdapat sungai Pregel dan merupakan tempat tinggal Duke of Prussia pada abad ke-16 (tahun 1736). Kota tersebut saat ini bernama Kaliningrad, dan merupakan pusat ekonomi dan industri utama di Russia Barat. Sungai Pregel membagi kota menjadi 4 daratan dengan mengalir mengitari pulau Kneiphof lalu bercabang menjadi dua buah anak sungai seperti tampak pada gambar:



Gambar 1 Jembatan Königsberg

Pada abad ke-18 dibangunlah tujuh jembatan yang menghubungkan keempat daratan tersebut. Pada hari Minggu, masyarakat Königsberg biasanya berjalan-jalan dari daratan satu ke daratan lainnya melalui jembatan tersebut. Mereka berpikir apakah mungkin untuk berjalan menyeberangi ketujuh jembatan tanpa melalui jembatan yang sama dari suatu daratan dan kembali ke tempat semula.

Masalah ini pertama kali dipecahkan oleh Leonhard Euler. Solusi Euler merepresentasikan masalah ini ke dalam sebuah graf dengan keempat daratan sebagai simpul (node), dan ketujuh jembatan sebagai sisi (edge). Graf yang dibuat Euler diperlihatkan pada gambar di bawah :



Gambar 2 Graf yang merepresentasikan jembatan Königsberg

Dengan graf tersebut, Euler berhasil menemukan jawaban kenapa Kita tidak dapat melalui ketujuh jembatan tersebut masing-masing sekali dan kembali ke tempat semula. Jawaban yang ditemukan Euler adalah karena tidak semua simpul pada graf tersebut berderajat genap. Simpul A, C, dan D berderajat 3, sedangkan simpul B berderajat 5.

Seiring dengan berjalannya waktu, teori graf juga semakin berkembang. Banyak sekali penerapan teori graf untuk merepresentasikan berbagai hal. Teori graf terutama digunakan untuk merepresentasikan hal-hal yang menyangkut *networking*, misalnya pemodelan sistem terdistribusi, pemodelan rute perjalanan, pemodelan isomer senyawa kimia, dan lain-lain.

Salah satu kegunaan graf adalah untuk memodelkan sistem lalu lintas. Karena di perkotaan besar seperti Bandung ini, sistem lalu lintasnya sudah sangat rumit. Banyak jalan besar yang dilalui oleh kendaraan setiap harinya. Dengan padatnya lalu lintas, maka pengelolaan sistem lalu lintas sangatlah diperlukan. Dengan pemodelan sistem lalu lintas menggunakan graf, pengelolaan lalu lintas akan semakin mudah.

2. KONSEP DASAR GRAF

Graf didefinisikan sebagai pasangan himpunan (V, E) yang dalam hal ini :

$V =$ himpunan tidak kosong dari simpul $= \{v_1, v_2, v_3, \dots, v_n\}$

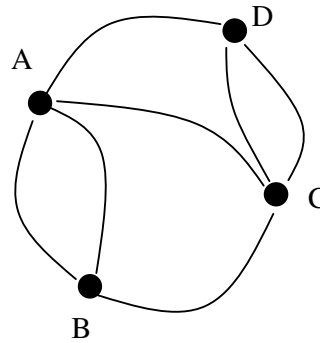
dan

$E =$ himpunan sisi yang menghubungkan simpul-simpul $= \{e_1, e_2, e_3, \dots, e_n\}$

Jadi sebuah graf dimungkinkan tidak mempunyai sisi satu buahpun, tetapi simpulnya harus ada, minimal satu. Graf yang hanya mempunyai satu buah simpul tanpa sebuah sisipun dinamakan graf trivial.

2.1 Graf Ganda

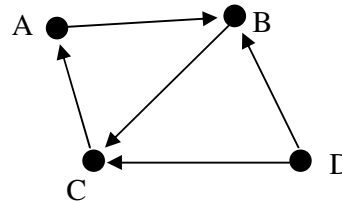
Graf ganda adalah graf yang memiliki lebih dari satu sisi untuk menghubungkan dua simpul. Pada graf dibawah, ditunjukkan graf yang memiliki sisi ganda. Yang dimaksud sisi ganda pada graf di bawah adalah sisi yang menghubungkan simpul A dan simpul B. Karena terdapat dua sisi yang menghubungkan simpul A dan simpul B, maka graf tersebut dinamakan graf ganda



Gambar 3 Graf ganda

2.2 Graf Berarah

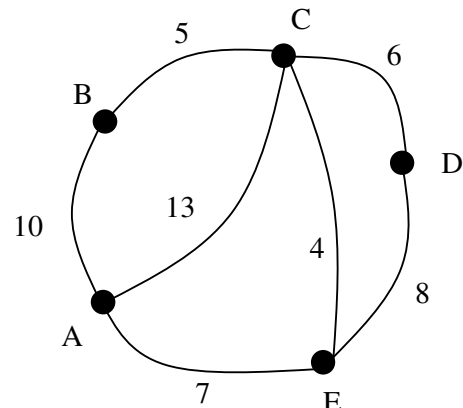
Graf yang setiap sisinya diberikan orientasi arah disebut graf berarah. Kita lebih suka menyebut sisi berarah dengan sebutan busur (arc). Pada graf berarah, (v_j, v_k) dan (v_k, v_j) menyatakan dua buah busur yang berbeda, dengan kata lain $(v_j, v_k) \neq (v_k, v_j)$. Untuk busur (v_j, v_k) , simpul v_j dinamakan simpul asal (initial vertex), dan simpul v_k dinamakan simpul terminal (terminal vertex). Contoh graf berarah ditunjukkan pada gambar di bawah.



Gambar 4 Graf berarah

2.3 Graf Berbobot

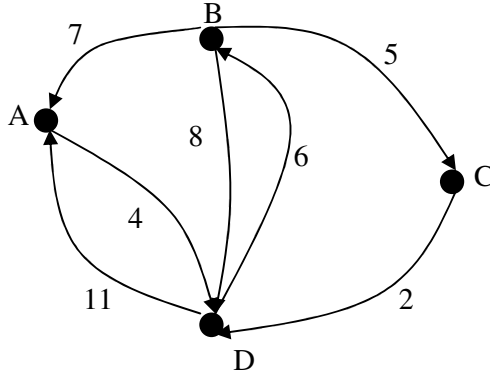
Graf berbobot adalah graf yang setiap sisinya diberi harga (bobot). Bobot pada tiap sisi dapat merepresentasikan sesuatu, misalnya jarak, prioritas, harga, dan lain lain.



Gambar 5 Graf berbobot

2.4 Graf Ganda Berarah Berbobot

Graf Ganda Berarah Berbobot adalah gabungan dari ketiga graf diatas. Untuk lebih jelasnya, gambar 6 di bawah, akan memberikan gambaran tentang graf ganda berarah berbobot.



Gambar 6 Graf ganda berarah berbobot

3. PEMODELAN SISTEM LALU LINTAS

Pada lalu lintas di perkotaan, daya dukung jalan menjadi faktor penting dalam menentukan lancarnya lalu lintas. Daya dukung jalan ini meliputi lebar jalan, kondisi jalan, rata-rata volume kendaraan yang lewat tiap satuan waktu, dan lain-lain. Pada lalu lintas perkotaan, jarak, menjadi sesuatu yang tidak terlalu penting. Dengan melewati jalan yang memiliki daya dukung yang baik, walaupun jaraknya lebih panjang, akan membuat kita sampai lebih cepat ke tujuan. Lebih cepat, daripada melewati jalan yang berjarak pendek, tapi kondisi jalannya rusak, volume kendaraan yang lewat besar, dan lain-lain. Pemanfaatan teori graf dalam sistem lalu lintas dapat dilakukan sebagai berikut :

1. Simpul dalam suatu graf digunakan untuk menghubungkan suatu persimpangan jalan.
2. Sisi dari suatu graf digunakan untuk melambangkan jalan.
3. Arah pada sisi merepresentasikan arah jalan yang dapat dilalui. Jadi bila terdapat jalan *One Way* atau satu arah, arah panah hanya akan menunjuk ke arah tertentu dan tidak sebaliknya.
4. Bobot / nilai dari sisi graf merepresentasikan daya dukung jalan tersebut.

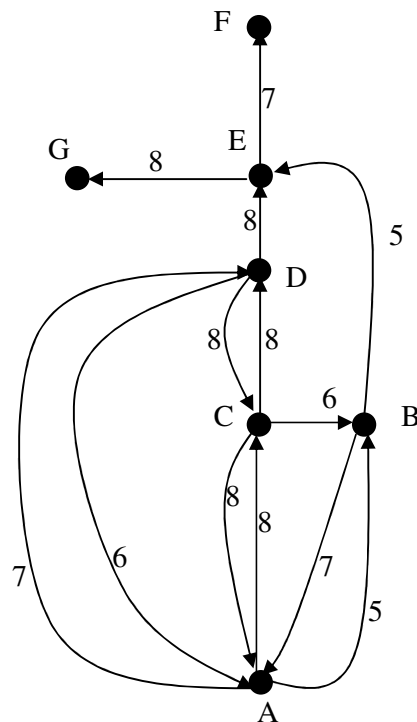
Dengan pemodelan semacam ini, diharapkan pengguna jalan dapat mengetahui jalur paling nyaman untuk sampai ke suatu tujuan. Dengan pemodelan ini, pengguna jalan dapat mengetahui dan menghindari jalan-jalan yang kemungkinan besar macet, atau jalan yang rusak, sehingga dapat menghindarinya dan memilih jalan lain yang memiliki daya dukung yang baik.

3.1 Pemberian Bobot pada Suatu Jalan

Pemberian bobot / nilai pada suatu jalan didasarkan pada :

1. Lebar jalan
 - Untuk jalan dengan lebar dibawah 1 meter diberi nilai 4
 - Antara 1 meter sampai 2 meter, nilai : 3
 - Lebih dari 2 meter sampai 3 meter, nilai : 2
 - Lebih dari 3 meter, nilai : 1
2. Kondisi Jalan
 - Jalan rusak parah, nilai : 4
 - Jalan rusak ringan, nilai : 3
 - Jalan baik, nilai : 2
 - Jalan sangat baik, nilai : 1
3. Volume kendaraan yang lewat tiap jam
 - Diatas 100 kendaraan/jam, nilai : 5
 - 81 sampai 100 kendaraan/jam, nilai : 4
 - 51 sampai 80 kendaraan/jam, nilai : 3
 - 31 sampai 60 kendaraan/jam, nilai : 2
 - 0 sampai 30 kendaraan/jam, nilai : 1

Bobot total adalah ketiga nilai diatas dijumlahkan. Berikut adalah contoh pemodelan sisitem lalu lintas dengan graf ganda berarah berbobot



Gambar 7 contoh pemodelan

Pada contoh di atas, misalkan saja

1. A adalah Simpang Dago
2. B adalah persimpangan antara jalan Taman Sari dan Jalan Ganeca
3. C adalah persimpangan Jalan Dago dan Jalan Ganeca
4. D adalah persimpangan antara Jalan Dipati Ukur dengan Jalan Dago
5. E adalah Perempatan Jalan merdeka, Riau, dan Dago
6. F dan G adalah persimpangan yang tidak masuk dalam pembahasan karena ini hanya potongan dari suatu jalan.

Jalan yang menghubungkan A ke C dan sebaliknya, juga jalan yang menghubungkan C ke D dan sebaliknya, diberi nilai 8. Karena pada jalan ini (dalam hal ini Jalan Dago) memiliki kondisi yang baik (nilai 1), badan jalan juga lebar (nilai 2), tapi volume kendaraan yang lewat sangat besar (nilai 5). Apabila Kita sekarang berada di Simpang Dago (simpul A) dan ingin pergi ke Bandung Indah Plaza yang terletak pada jalan yang menghubungkan simpul E dan F, rute terpendek adalah langsung melewati jalan yang menghubungkan A dan C, C dan D, D dan E, lalu E dan F. Sayangnya rute itu memiliki bobot yang besar, jadi mungkin saja waktu yang dibutuhkan menjadi lama. Rute alternative yang bisa kita gunakan adalah melewati jalan yang menghubungkan A dan B, B dan E, lalu E dan F. Jalan ini memang lebih panjang, tapi karena relatif lebih lancar, jadi kemungkinan besar kita bisa sampai tujuan lebih cepat.

Untuk pencarian rute tercepat, kita bisa menggunakan Algoritma Dijkstra (diberi nama sesuai penemunya, Edsger Wybe Dijkstra). Sebenarnya algoritma ini umumnya digunakan untuk mencari lintasan terpendek dari sebuah graf berbobot dengan simpul menyatakan suatu tempat, sisi menyatakan jalan, dan bobot menyatakan panjang lintasan. Tapi dengan menganalogikan panjang lintasan dengan daya dukung jalan, kita bisa menggunakan algoritma ini untuk mencari rute tercepat pada pemodelan sistem lalu lintas ini. Algoritma Dijkstra mencari lintasan terpendek dengan sejumlah langkah. Algoritma ini menggunakan prinsip *Greedy*. Prinsip *Greedy* pada algoritma Dijkstra menyatakan bahwa pada setiap langkah Kita memilih sisi yang berbobot minimum dan memasukkannya ke dalam himpunan solusi.

Berikut ini adalah contoh algoritma Dijkstra dalam *Pseudo Code*

procedure Dijkstra (input m: matriks, a : simpul awal)
{Mencari lintasan terpendek dari simpul awal a ke semua simpul lainnya,}

Deklarasi

$S_1, S_2, \dots, S_n : \text{integer} \{ \text{larik integer} \}$
 $d_1, d_2, \dots, d_n : \text{integer} \{ \text{larik integer} \}$
 $i : \text{integer}$

Algoritma

{Langkah 0 (inialisasi :)
for i \leftarrow 1 to n do
 $S_i \leftarrow 0$
 $d_i \leftarrow m_{ai}$
endfor

(Langkah 1)

$S_a \leftarrow 1$ (karena simpul a adalah simpul asal lintasan terpendek, jadi simpul a sudah pasti terpilih dalam lintasan terpendek)
 $d_a \leftarrow \infty$ (tidak ada lintasan terpendek dari simpul a ke a)

(Langkah 2, 3, ..., n-1 :)

for i \leftarrow 2 to n-1 do
 Cari j sedemikian sehingga $S_j = 0$
 dan $d_j = \min \{d_1, d_2, \dots, d_n\}$
 $S_j \leftarrow 1$ { simpul j sudah terpilih ke dalam lintasan terpendek }
 Perbarui d_i untuk $i = 1, 2, 3, \dots, n$
 dengan : d_i (baru) = $\min \{d_i$ (lama), $d_j + m_{ji}\}$
endfor

4. HASIL DAN PEMBAHASAN

Dengan menggunakan pemodelan sistem lalu lintas dengan graf ganda berarah berbobot dan algoritma Dijkstra, kita bisa mendapatkan rute tercepat ke suatu tempat. Rute yang didapatkan bisa saja bukan jalan terpendek ke suatu tempat tapi rute yang didapatkan adalah rute yang bisa membawa Kita sampai lebih cepat ke suatu tempat.

Hal ini bisa dibuktikan dengan contoh nyata. Misal saja kita berada dari Simpang Dago, ingin pergi Ke Gramedia yang terdapat di Jalan Merdeka. Rute terpendek yang bisa Kita lalui adalah tinggal berjalan lurus saja ke arah Jalan Merdeka. Tapi Jalan yang dilalui itu seringkali macet, terutama bila hari libur. Masalahnya bukan terletak pada keadaan jalan yang rusak, ataupun badan jalan yang sempit, tapi masalahnya adalah karena volume kendaraan yang melewati jalan itu sangatlah banyak. Terutama pada hari libur.

Dengan menggunakan pemodelan ini dan algoritma Dijkstra, kita bisa mendapatkan rute yang lebih baik, yaitu melewati Jalan Taman Sari. Ini memang bukan jalan terpendek, tapi ini adalah jalan yang dapat membawa kita lebih cepat ke Gramedia.

5. KESIMPULAN

Pemodelan Sistem Lalu Lintas dengan graf ganda berarah berbobot dapat digunakan untuk memodelkan sistem lalu lintas dengan lebih baik daripada menggunakan graf berarah biasa. Terutama untuk memodelkan sistem lalu lintas di perkotaan, di mana orang lebih mengutamakan waktu. Dengan menambahkan bobot sebagai representasi daya dukung jalan, pencarian rute menjadi lebih baik.

DAFTAR REFERENSI

- [1] Munir, Rinaldi. (2001). Diktat Kuliah IF2153 Matematika Diskrit. Departemen Teknik Informatika, Institut Teknologi Bandung.
- [2] http://en.wikipedia.org/wiki/Graph_theory
Tanggal akses 1 Januari 2008, pukul 19.00
- [3] http://en.wikipedia.org/wiki/List_of_graph_theory
Tanggal akses 1 Januari 2008, pukul 20.00
- [4] <http://portal.acm.org/citation.cfm?id=771516.771525>
Tanggal akses 2 Januari 2008, pukul 09.00