

APLIKASI TEORI PRIM DALAM MENENTUKAN JALUR MUDIK

Biyani Satyanegara – NIM : 13508057

Program Studi Teknik Informatika, Sekolah Teknik Elektro Informatika, Institut Teknologi Bandung
Jalan Ganesha 10, Bandung

e-mail: if18057@students.if.itb.ac.id / biyansatya@yahoo.com

ABSTRAK

Makalah ini membahas penggunaan teori dalam struktur diskrit untuk menentukan jalur mudik. Metode yang digunakan dalam makalah ini menggunakan teori prim pada graf berbobot untuk menemukan jalur terpendek dari mulai kota asal hingga kota tujuan dalam perjalanan mudik. Namun penentuan jalur mudik ini hanya didasarkan pada jarak tempuh yang paling minimal.

Kata kunci: Graf, Prim, mudik, jalur, terdekat



Gambar 1. Peta Jalur Mudik

1. PENDAHULUAN

Mudik merupakan fenomena umum yang dilakukan oleh rakyat Indonesia menjelang hari raya Idul Fitri. Pada saat menjelang hari raya Idul Fitri masyarakat dari seluruh kalangan berbondong-bondong melakukan perjalanan dari kota perantauannya ke daerah asalnya masing-masing. Tujuan mereka pulang ke kampung halaman adalah untuk menyambung tali silaturahmi, bertemu sanak saudara ataupun membagi-bagikan rezeki yang telah didapatkan di kota perantauan.

Permasalahan yang sering timbul dan yang sering dialami oleh pemudik adalah pemilihan jalur dari kota asal ke kota tujuan. Para pemudik sulit untuk menentukan jalur mudik tercepat yang akan dilalui. Kali ini penulis akan mencoba menghubungkan antara fenomena mudik yang terjadi pada masyarakat Indonesia. Dengan mata kuliah struktur diskrit yang sudah dipelajari selama satu semester untuk menentukan jalur mudik terdekat dengan menggunakan algoritma prim yang digunakan pada bab pohon dan graf. Peta mudik akan diubah ke dalam bentuk graf berbobot dengan kota sebagai simpul, lintasan sebagai jalan dan bobot sebagai panjang jalan yang harus ditempuh pemudik. Kemudian penentuan jalur terpendek dibuat berdasarkan algoritma prim dengan sedikit modifikasi.

2. METODE

Dalam makalah ini penulis akan sedikit membahas tentang teori yang akan digunakan untuk memecahkan masalah jalur mudik dan juga aplikasinya. Teori yang digunakan digunakan adalah algoritma prim pada pohon merentang

2.1 Algoritma Prim

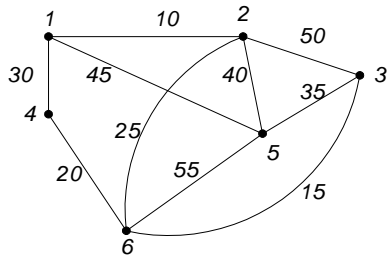
Algoritma Prim adalah algoritma yang digunakan untuk mencari pohon merentang minimum. Pohon merentang minimum adalah graf yang tidak mengandung sirkuit dan mengandung bobot yang paling minimum.

Misalkan T adalah pohon merentang yang sisi-sisinya diambil dari graf G . Algoritma Prim membentuk pohon merentang minimum langkah per langkah. Pada setiap langkah kita mengambil sisi e dari graf G yang mempunyai bobot minimum dan bersisian dengan simpul-simpul di dalam T tetapi e tidak membentuk sirkuit di dalam T . [1]

Berikut ini adalah langkah-langkah algoritma prim :

1. Ambil sisi dari graf G yang berbobot minimum, masukkan ke dalam T
2. Pilih sisi e yang mempunyai bobot minimum dan bersisian dengan simpul di T , tetapi e tidak membentuk sirkuit di T . Masukkan e ke dalam T
3. Ulangi langkah 2 sebanyak $n-2$ kali (n adalah jumlah simpul). [1]

Contoh penggunaan algoritma prim :



Langkah	Sisi	Bobot	Pohon rentang
1	(1,2)	10	
2	(2,6)	25	
3	(3,6)	15	
4	(4,6)	20	
5	(3,5)	35	

3. Aplikasi

Untuk mensimulasikan pencarian jalur terpendek dengan menggunakan algoritma prim, langkah-langkah yang dilakukan sebagai berikut :

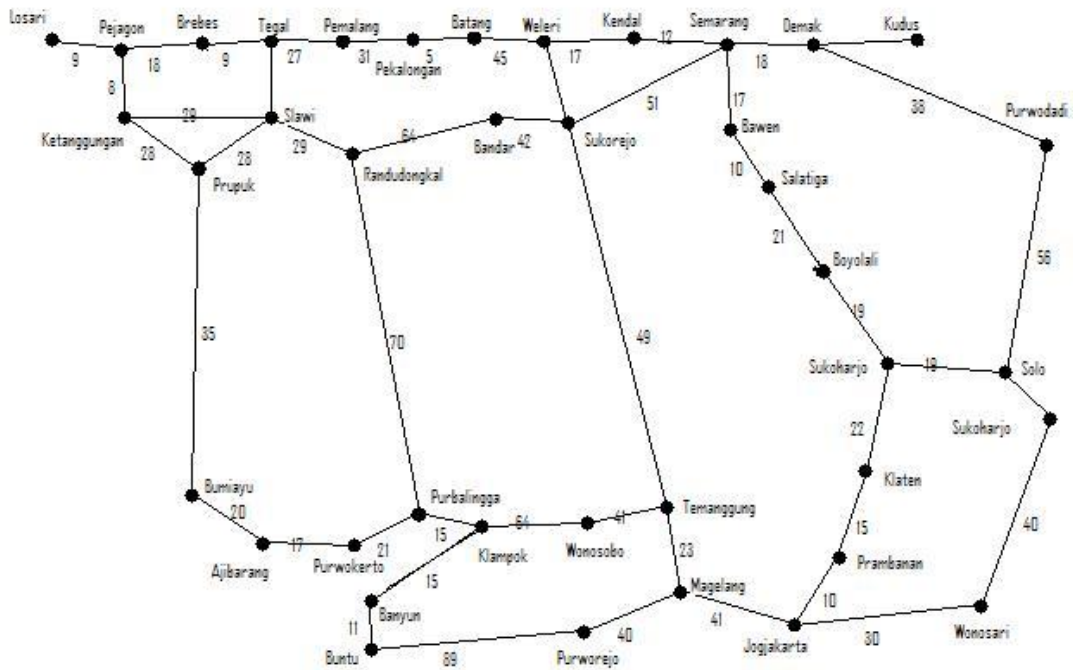
1. Tentukan kota asal dan kota tujuan. Dalam simulasi yang dilakukan oleh penulis, kota asal dimulai dari Kota Losari dan kota tujuan adalah Kota Solo.
2. Ubah dari bentuk peta menjadi bentuk graf berbobot dengan kota sebagai simpul, sisi sebagai jalan yang ditempuh dan bobot besar jarak yang ditempuh. Gambar 2 menunjukkan peta daerah tertentu dalam bentuk graf berbobot
3. Lakukan langkah-langkah algoritma prim pada peta yang telah diubah dalam bentuk graf berbobot

Tabel 1. Tabel Urutan Langkah berdasarkan algoritma Prim

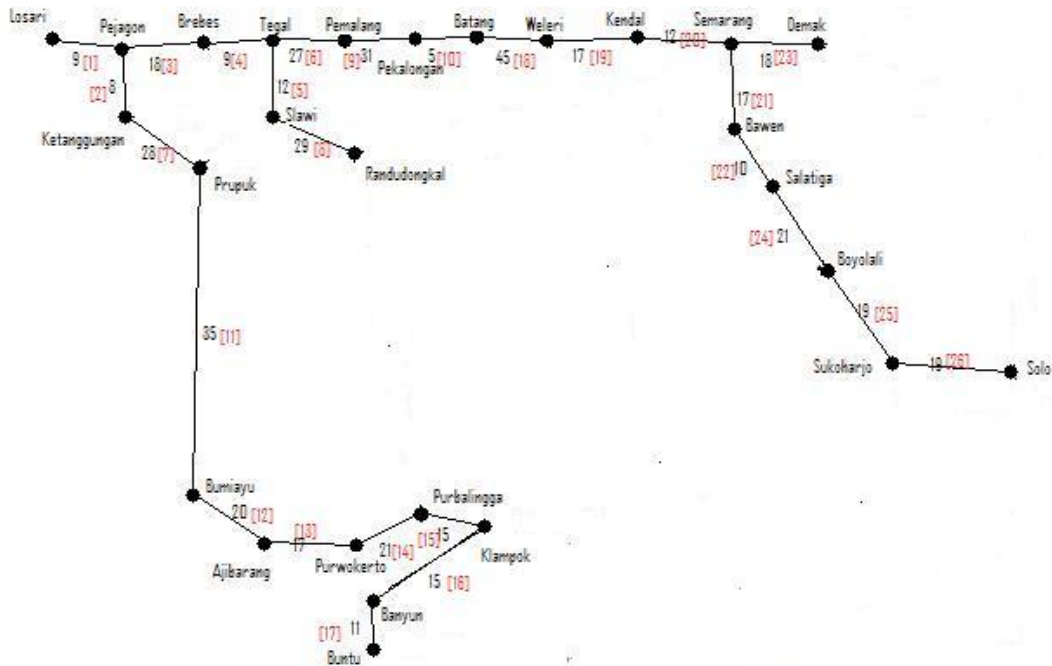
Langkah	Sisi	Bobot(km)
1	Losari – Pejagon	9
2	Pejagon-Ketanggungan	8
3	Pejagon – Brebes	18
4	Brebes – Tegal	9
5	Tegal – Slawi	12
6	Tegal – Pemalang	27
7	Ketanggungan – Prupuk	28
8	Slawi – RanduDongkal	29
9	Pemalang – Pekalongan	31
10	Pekalongan – Batang	5
11	Prupuk – Bumi ayu	35
12	Bumiayu – Ajibarang	20
13	Ajibarang – Purwokerto	17
14	Purwokerto – Purbalingga	21
15	Purbalingga – Klampok	15
16	Klampok – Banyun	15
17	Banyun – Buntu	11
18	Batang – Weleri	45
19	Weleri – Kendal	17
20	Kendal – Semarang	12
21	Semarang – Bawen	17
22	Bawen – Salatiga	10
23	Semarang – Demak	18
24	Salatiga – boyolali	21
25	Boyolali – Kartosuro	19
26	Kartosuro – Solo	19

4. Langkah-langkah algoritma Prim dilakukan hingga terdapat lintasan tunggal yang menghubungkan kota asal ke kota tujuan.

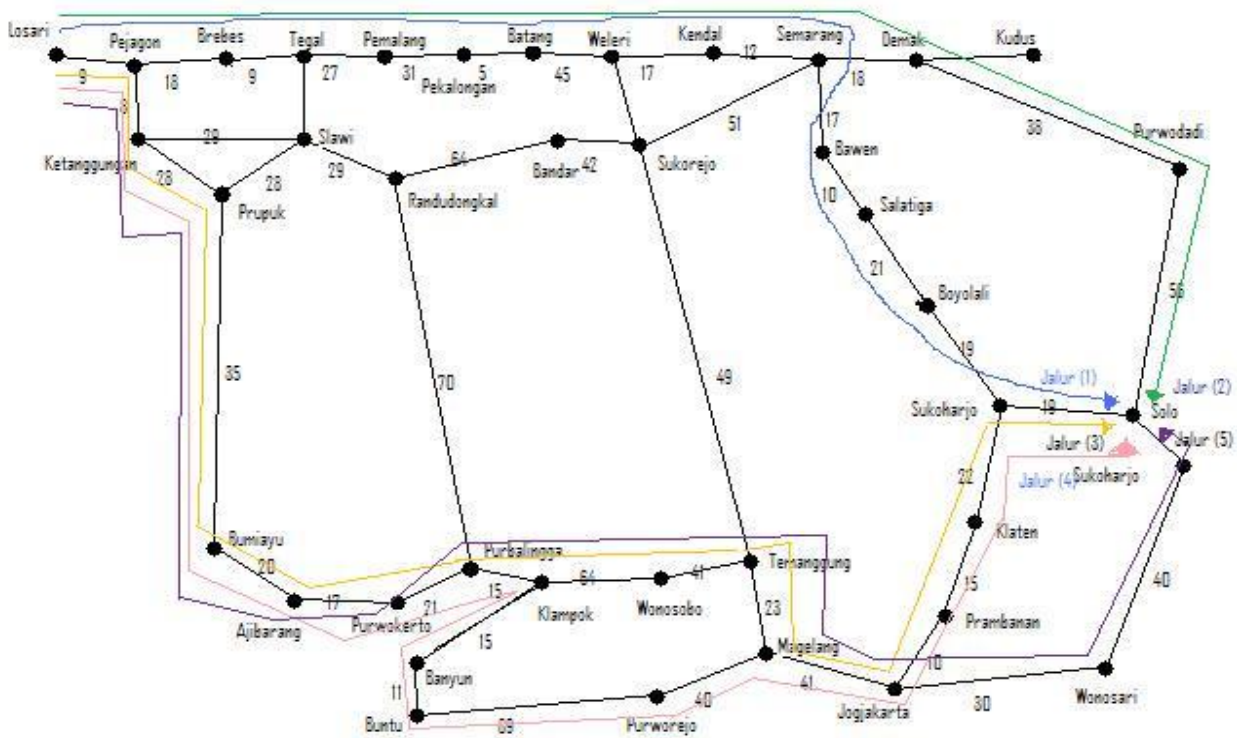
Pada langkah ke 26 terdapat suatu lintasan yang menghubungkan antara kota asal dan tujuan yaitu kota Losari dan Kota Solo.



Gambar 2. Peta Jalur Mudik dalam bentuk Graf Berbobot



**Ket : [x] melambangkan langkah pada algoritma prim
Gambar 3. Jalur yang dihasilkan oleh algoritma prim**



Gambar 4. Gambar beberapa jalur yang dapat ditempuh dari Kota Asal ke Kota Tujuan

Dari tabel 1 didapatkan gambar 3. Pada gambar 3 dapat menunjukkan bahwa jarak terdekat antara Kota Losari dan Solo adalah melalui : Losari – Pejagon – Brebes – Tegal – Pemalang – Pekalongan – Batang – Weleri – Kendal – Semarang – Bawen – Salatiga – Boyolali – Kartosuro – Solo. Total jarak yang ditempuh adalah 259 km.

Mari kita buktikan bahwa jalur yang diberikan oleh algoritma prim adalah jalur yang terdekat antara Kota Losari dan Kota Solo. Terdapat beberapa jalur yang menghubungkan antara Kota Losari dan Kota Solo. Pada gambar 4 dapat dilihat jalur-jalur yang dapat menghubungkan antara Kota Losari dan Kota Solo. Kedua kota tersebut dapat dihubungkan dengan berbagai cara. Untuk membuktikan, setiap jalur harus diketahui total jarak yang akan ditempuh. Total jarak yang dimiliki oleh masing-masing jalur dapat dilihat pada tabel 2 berikut.

Tabel 2. Tabel Jarak yang dimiliki masing-masing jalur

Jalur	Jarak (km)
1	259
2	285
3	388
4	415
5	429

Dari tabel 2, total jarak yang paling sedikit adalah 259 km yaitu jalur 1. Ini menunjukkan bahwa jalur terdekat yang menghubungkan antara Kota Losari dan Kota Solo adalah jalur 1. Setelah itu mari kita bandingkan antara rute jalur 1 yang ada pada gambar 4 dengan gambar 2. Jalur 1 yang berada pada gambar empat sama dengan rute yang ditunjukkan pada gambar 2 yang dihasilkan dari algoritma prim. Dari kedua gambar tersebut maka dapat disimpulkan bahwa jalur yang dibentuk oleh algoritma prim merupakan jalur yang terdekat yang menghubungkan dua Kota yaitu Kota Losari sebagai kota asal dan Kota Solo sebagai kota tujuan akhir.

Pembuktian bahwa algoritma prim dapat diaplikasikan untuk menentukan jarak terdekat antara dua kota dengan menggunakan aplikasi graf berbobot.

IV. KESIMPULAN

Kesimpulan yang dapat diambil dalam pembuatan makalah ini antara lain :

1. Algoritma Prim pada graf berbobot dapat diaplikasikan untuk mencari jalur terdekat dua buah kota
2. Metode pencarian jalur terdekat yang dilakukan oleh penulis hanya terbatas berdasarkan satu faktor yaitu jarak

REFERENSI

- [1] Rinaldi Munir, "Buku Teks Ilmu Komputer Matematika Diskrit", Informatika, 2005.
- [2] Peta Mudik Jawa Bali, 2008.