

METODE *BRANCH AND BOUND* UNTUK MENEMUKAN *SHORTEST PATH*

Mira Muliati – NIM : 13505110

Program Studi Teknik Informatika
Sekolah Teknik Elektro Informatika
Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if 15110@students.if.itb.ac.id

Abstrak

Algoritma *Branch and Bound* (B&B) adalah metode pencarian solusi secara sistematis. Pada metode ini ruang solusi diorganisasikan ke dalam pohon ruang status. Pembentukan pohon ruang status dilakukan berdasarkan skema *BFS* (*Breadth First Search*) dengan pemilihan simpul yang diekspansi berdasarkan nilai ongkos simpul-simpul hidup. Pada pencarian jarak terpendek dari suatu simpul ke simpul yang lain, nilai ongkos menggambarkan nilai taksiran termurah dari simpul tersebut ke simpul tujuan. Jadi, nilai ongkos disini berfungsi sebagai nilai pembatas (*bound*).

Nilai batas didapatkan dari reduksi baris dan kolom matriks M yang merepresentasikan graf berarah. Reduksi dilakukan dengan mengurangi suatu baris atau kolom dengan nilai terkecil baris atau kolom tersebut sedemikian sehingga didapatkan matriks tereduksi M_i dengan sebuah nilai nol pada setiap baris dan setiap kolom. Total nilai pengurang menjadi nilai batas simpul akar. Selanjutnya untuk setiap simpul anak yang dibangkitkan dengan mengunjungi $M(i,j)$, dilakukan reduksi matriks M_i untuk mendapatkan matriks tereduksi $M_{(i,j)}$. Sebelum mereduksi, terlebih dahulu diubah nilai pada baris ke i , kolom ke j dan $M(j, \text{simpul awal})$ diubah menjadi ∞ . Nilai batas simpul anak dihitung dengan rumus :

$$c(n) = c(a) + M(i,j) + r$$

dimana :

- $c(n)$: nilai batas simpul selanjutnya yang sedang dikunjungi
- $c(a)$: nilai batas simpul asal
- $M(i,j)$: nilai matriks M pada baris ke i dan kolom ke j
- r : total nilai pengurang yang digunakan untuk mereduksi matriks pada

Simpul anak beserta nilai batasnya dimasukkan ke dalam sebuah *Queue* untuk dipilih simpul mana yang akan diekspansi. Simpul selanjutnya yang diekspansi adalah simpul dengan nilai batas terkecil. Pada langkah selanjutnya, untuk mendapatkan matriks tereduksi dari simpul anak digunakan matriks tereduksi simpul bapak. Begitu seterusnya sampai dikunjungi $M(i,j)$ yang dituju.

Kata kunci: nilai batas, simpul ekspansi, pohon ruang status, reduksi matriks.

1. Pendahuluan

1.1 Algoritma *Branch and Bound*

1.1.1 *BFS* (*Breadth First Search*)

BFS dikenal sebagai pencarian melebar dalam pohon. Misalkan graf G mempunyai n buah simpul. Traversal di dalam graf dilakukan mulai simpul v . Algoritma *BFS* adalah sebagai berikut : bangkitkan simpul v , kemudian semua simpul yang bertetangga dengan simpul v dibangkitkan terlebih dahulu. Selanjutnya, simpul yang belum dibangkitkan dan bertetangga dengan simpul-simpul tadi dibangkitkan, demikian seterusnya.

Jika graf berbentuk pohon berakar, maka semua simpul pada level d dibangkitkan terlebih dahulu sebelum membangkitkan simpul-simpul pada level $d+1$.^[1]

Algoritma *BFS* menggunakan antrian untuk menyimpan simpul-simpul yang baru dibangkitkan. Simpul-simpul yang baru dibangkitkan ditempatkan di belakang antrian.^[1] Prinsip antrian yang digunakan adalah *FIFO* (*First In First Out*). Dengan skema ini, simpul hidup dimasukkan ke dalam antrian, simpul berikutnya yang akan menjadi simpul ekspansi adalah simpul yang pertama masuk ke dalam antrian.

1.1.2 Pencarian Solusi Algoritma B&B

Pada algoritma B&B ruang solusi diorganisasikan ke dalam pohon ruang status yang dibangun dengan skema BFS. Untuk mempercepat pencarian ke simpul solusi, setiap simpul diberi nilai ongkos $c(i)$ yang merupakan taksiran lintasan termurah dari simpul akar simpul status tujuan. Simpul berikutnya yang akan diekspansi tidak lagi berdasarkan prinsip FIFO pada antrian, melainkan dipilih simpul dengan nilai ongkos terkecil.

Fungsi heuristik untuk menghitung taksiran nilai ongkos dinyatakan secara umum sebagai :

$$c(i) = f(i) + g(i) \quad (1)$$

yang dalam hal ini:

- $c(i)$ = ongkos untuk simpul i
- $f(i)$ = ongkos untuk mencapai simpul i
- $g(i)$ = ongkos untuk mencapai simpul tujuan dari simpul i ^[1]

Algoritma B&B :

1. Masukkan simpul akar ke dalam antrian Q. Jika simpul akar adalah simpul solusi, maka solusi telah ditemukan. Stop.
2. Jika Q kosong, tidak ada solusi. Stop.
3. Jika Q tidak kosong, pilih dari antrian Q simpul i yang mempunyai $c(i)$ terkecil. Jika terdapat beberapa simpul i yang memenuhi, pilih satu secara sembarang.
4. Jika simpul i adalah simpul simpul solusi, berarti simpul telah ditemukan, stop. Jika simpul i bukan simpul solusi maka bangkitkan semua anak-anaknya. Jika i tidak mempunyai anak, kembali ke langkah 2.
5. Untuk setiap anak j dari simpul i , hitung $c(i)$, dan masukkan anak-anak tersebut ke dalam antrian Q.
6. Kembali ke langkah 2.^[1]

1.1.3 Persoalan Lintasan Terpendek (*Shortest Path*)

Persoalan lintasan terpendek adalah menemukan lintasan terpendek dari sebuah simpul asal a ke simpul lain dalam graf G . Graf G merupakan graf berbobot positif.

2. Metode

Untuk menjelaskan dan menguji metode, diambil contoh sebuah matriks yang merepresentasikan sebuah graph berarah G :

$$\begin{bmatrix} \sim & 50 & 10 & 40 & 45 & \sim \\ \sim & \sim & 15 & \sim & 10 & 5 \\ 20 & \sim & \sim & \sim & 15 & \sim \\ \sim & 20 & \sim & \sim & 35 & 3 \\ \sim & 10 & \sim & 30 & \sim & \sim \\ 10 & \sim & 6 & \sim & \sim & \sim \end{bmatrix}$$

Metode yang digunakan adalah metode B&B untuk TSP. Pada TSP, solusi adalah sirkuit Hamilton dengan panjang rute terpendek. Persoalan Shortest Path memiliki tujuan yang sama yaitu menemukan rute terpendek namun bukan untuk sebuah sirkuit Hamilton melainkan rute terpendek dari sebuah simpul ke simpul lainnya. Metode yang digunakan berikut ini berdasarkan metode B&B untuk TSP dari [1].

Akan dicari rute terpendek dari simpul 3 ke simpul 6. Langkah pertama dilakukan reduksi matriks G :

$$\begin{bmatrix} \sim & 50 & 10 & 40 & 45 & \sim \\ \sim & \sim & 15 & \sim & 10 & 5 \\ 20 & \sim & \sim & \sim & 15 & \sim \\ \sim & 20 & \sim & \sim & 35 & 3 \\ \sim & 10 & \sim & 30 & \sim & \sim \\ 10 & \sim & 6 & \sim & \sim & \sim \end{bmatrix} \begin{array}{l} R1 - 10 \\ R2 - 5 \\ R3 - 15 \\ R4 - 3 \\ R5 - 10 \\ R6 - 6 \end{array}$$

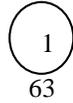
Dihasilkan :

$$\begin{bmatrix} \sim & 40 & 0 & 30 & 35 & \sim \\ \sim & \sim & 10 & \sim & 5 & 0 \\ 5 & \sim & \sim & \sim & 0 & \sim \\ \sim & 17 & \sim & \sim & 32 & 0 \\ \sim & 0 & \sim & 20 & \sim & \sim \\ 4 & \sim & 0 & \sim & \sim & \sim \end{bmatrix} \begin{array}{l} C1 - 4 \\ C4 - 20 \end{array}$$

Karena belum terdapat nilai nol pada kolom satu dan empat, maka masing-masing kolom tersebut dikurangi dengan nilai terkecil dalam kolom. Matriks tereduksi akhir yang didapat adalah :

$$\begin{bmatrix} \sim & 40 & 0 & 30 & 35 & \sim \\ \sim & \sim & 10 & \sim & 5 & 0 \\ 5 & \sim & \sim & \sim & 0 & \sim \\ \sim & 17 & \sim & \sim & 32 & 0 \\ \sim & 0 & \sim & 0 & \sim & \sim \\ 0 & \sim & 0 & \sim & \sim & \sim \end{bmatrix}$$

Total nilai pengurang adalah $(10 + 5 + 15 + 3 + 10 + 6) + (4 + 20) = 63$. Simpul akar ini dimasukkan antrian Q dengan nilai batas 63. Simpul akar ini selanjutnya akan diekspansi. Pohon ruang status yang dibangun :



Langkah untuk membangkitkan simpul selanjutnya adalah :

1. Ubah semua nilai pada matriks bobot tereduksi untuk simpul bapak pada baris ke i dan kolom ke j dengan \sim . Dimana i dan j menunjukkan simpul yang dituju pada graf.
2. Ubah nilai matriks tereduksi pada $(j,3)$ agar simpul tersebut tidak digunakan.
3. Reduksi kembali matriksnya.
4. Hitung nilai batas simpul pada pohon yang sedang dibangun.
5. Masukkan simpul tersebut dalam antrian Q.
6. Bangkitkan simpul anak yang lain seperti pada langkah pertama.
7. Pilih simpul dengan nilai batas terkecil untuk diekspansi.
8. Kembali ke langkah 1.

Selanjutnya dibangun simpul-simpul yang bertetangga dengan simpul 3 yaitu simpul 1 dan 4.

1. Lintasan 3,1

$$\begin{bmatrix} \sim & 40 & 0 & 30 & 35 & \sim \\ \sim & \sim & 10 & \sim & 5 & 0 \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & 17 & \sim & \sim & 32 & 0 \\ \sim & 0 & \sim & 0 & \sim & \sim \\ \sim & \sim & 0 & \sim & \sim & \sim \end{bmatrix} \text{C4} - 5$$

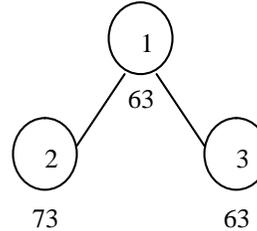
$$c(2) = c(1) + G(3,1) + r = 63 + 5 + 5 = 73$$

2. Lintasan 3,5

$$\begin{bmatrix} \sim & 40 & 0 & 30 & \sim & \sim \\ \sim & \sim & 10 & \sim & \sim & 0 \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & 17 & \sim & \sim & \sim & 0 \\ \sim & 0 & \sim & 0 & \sim & \sim \\ 0 & \sim & 0 & \sim & \sim & \sim \end{bmatrix}$$

$$c(3) = c(1) + G(3,5) + r = 63 + 0 + 0 = 63$$

Simpul hidup dalam antrian Q adalah : 2(73), 3(63). Karena simpul 3 memiliki nilai batas terkecil, simpul selanjutnya yang diekspansi adalah simpul 3. Pohon yang terbentuk :



3. Lintasan 3,5,2

$$\begin{bmatrix} \sim & \sim & 0 & 30 & \sim & \sim \\ \sim & \sim & \sim & \sim & \sim & 0 \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & \sim & \sim & \sim & \sim & 0 \\ \sim & \sim & \sim & \sim & \sim & \sim \\ 0 & \sim & 0 & \sim & \sim & \sim \end{bmatrix}$$

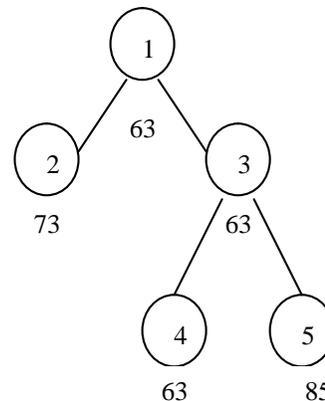
$$c(4) = c(3) + G(5,2) + r = 63 + 0 + 0 = 63$$

4. Lintasan 3,5,4

$$\begin{bmatrix} \sim & 40 & 0 & \sim & \sim & \sim \\ \sim & \sim & 10 & \sim & \sim & 0 \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & 17 & \sim & \sim & \sim & 0 \\ \sim & \sim & \sim & \sim & \sim & \sim \\ 0 & \sim & 0 & \sim & \sim & \sim \end{bmatrix} \text{C2} - 17$$

$$c(5) = c(3) + G(5,4) + r = 68 + 0 + 17 = 85$$

Simpul hidup dalam antrian Q adalah 2(73), 4(63), 5(85). Pohon yang terbentuk :



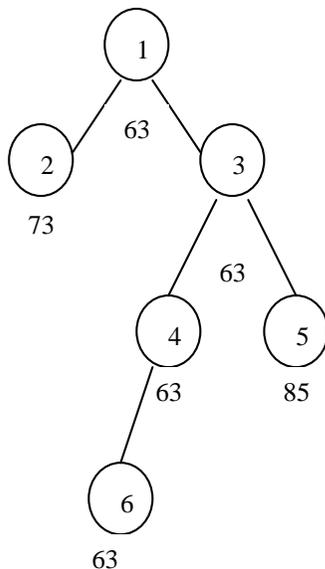
Karena simpul 4 memiliki nilai batas terkecil, selanjutnya akan diekspansi. Pada graf, dari simpul 2 kita bisa ke simpul 3, 5 dan 6. Sampai saat ini jalur yang terbentuk adalah 3-5-2, karena itu dari simpul 2 kita tidak akan mengunjungi simpul 3 dan 5.

5. Lintasan 3,5,2,6

$$\begin{bmatrix} \sim & \sim & 0 & 30 & \sim & \sim \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & \sim & \sim & \sim & \sim & \sim \\ \sim & \sim & \sim & \sim & \sim & \sim \\ 0 & \sim & 0 & \sim & \sim & \sim \end{bmatrix}$$

$$c(6) = c(4) + G(2,6) + r = 63 + 0 + 0 = 63$$

Pohon yang terbentuk :



Sampai di sini telah terbentuk lintasan 3-5-2-6 yang artinya telah ditemukan solusi jadi, pembentukan pohon dapat dihentikan. Solusi ini merupakan solusi terbaik sejauh ini. Simpul-simpul yang lain dapat dibunuh karena dari simpul-simpul tersebut tidak mungkin dihasilkan solusi yang lebih baik.

5. Kesimpulan

Algoritma pencarian jarak terpendek dengan metode B&B :

1. Reduksi matriks yang merepresentasikan graf dengan mengurangi setiap baris atau kolom dengan nilai terkecil baris atau kolom tersebut sehingga didapat matriks dengan minimal satu nilai nol pada setiap baris dan kolom.
2. Masukkan simpul akar dengan nilai batas $c(1) = \text{total pengurang matriks}$ ke dalam antrian Q. Jika simpul akar adalah simpul solusi, maka solusi telah ditemukan. Stop.
3. Jika Q kosong, tidak ada solusi. Stop.
4. Jika Q tidak kosong, pilih dari antrian Q simpul i yang mempunyai $c(i)$ terkecil. Jika terdapat beberapa simpul i yang memenuhi, pilih satu secara sembarang.
5. Jika simpul i adalah simpul solusi, berarti solusi telah ditemukan, stop. Jika simpul i bukan simpul solusi maka bangkitkan semua anak-anaknya. Cara membangkitkan simpul anak :
 1. Ubah semua nilai pada matriks bobot tereduksi milik simpul bapak pada baris ke i dan kolom ke j dengan \sim . Dimana i dan j menunjukkan simpul yang dituju pada graf.
 2. Ubah nilai matriks tereduksi pada (j, simpul asal pada graf) agar simpul tersebut tidak digunakan.
 3. Reduksi kembali matriksnya.

Jika i tidak mempunyai anak, kembali ke langkah 2.
4. Untuk setiap anak j dari simpul i, hitung $c(j)$, dan masukkan anak-anak tersebut ke dalam antrian Q. $c(j)$ didapat dari rumus :
$$c(j) = c(i) + M(k,l) + r$$

dimana :

 - $c(j)$: nilai ongkos simpul anak
 - $c(i)$: nilai ongkos simpul bapak
 - $M(k,l)$: bobot sisi k ke l
 - R : total pengurang pada proses reduksi
5. Kembali ke langkah 2.

DAFTAR PUSTAKA

- [1] Munir, Rinaldi. (2007). Diktat Kuliah IF2251 Strategi Algoritmik. Departemen Teknik Informatika, Institut Teknologi Bandung.

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.