Penghematan BBM pada Bisnis Antar-Jemput dengan Algoritma *Branch and Bound*

Chrestella Stephanie - 13512005
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13512005@std.stei.itb.ac.id

***Abstrak*—Makalah ini membahas penggunaan algoritma *Branch and Bound* yang bertujuan untuk menghemat pemakaian BBM. Penghematan BBM dapat dicapai dengan memilih jarak terpendek total dari titik awal penjemputan sampai kembali ke titik awal tersebut lagi secara keseluruhan. Algoritma *Branch and Bound* dapat digunakan untuk menyelesaikan permasalahan memilih urutan penjemputan untuk memperoleh jarak total yang paling minimal. Dengan memililh jarak yang paling minimal, maka pemakaian BBM menjadi lebih sedikit dan akan berdampak cukup besar bila hal ini dilakukan dalam kurun waktu yang cukup lama.**

Kata Kunci*—*shortest path, Branch and Bound,* BBM, *reduced cost matrix

I. PENDAHULUAN

Saat ini, keadaan jalanan di kota-kota di Indonesia, khususnya kota-kota besar seperti di Jakarta, Bandung, dan sebagainya sudah semakin padat oleh kendaraan, baik oleh kendaraan umum maupun kendaraan pribadi. Bahkan di beberapa ruas jalan, sudah biasa ditemukan antrian kemacetan yang cukup lama dan panjang. Ditambah kondisi harga BBM yang tidak murah, dan orang-orang masa kini yang cenderung menuntut kepraktisan. Maka dari itu, memilih untuk menggunakan jasa antar-jemput merupakan salah satu pilihan yang tepat untuk menghindari hal-hal di atas. Selain menghindari hal-hal yang sudah disebutkan di atas, hal ini dapat membantu pemerintah mewujudkan kota yang rendah polusi dan tidak macet.

Dilihat dari kemungkinan banyaknya penduduk perkotaan yang merasakan permasalahan demikian, maka membuka bisnis antar-jemput merupakan salah satu peluang bisnis yang cukup menjanjikan. Sasaran konsumennya pun cukup banyak, mulai dari anak sekolah sampai pegawai kantor. Namun, sang *owner* harus cerdas saat membuka bisnis ini. Jangan sampai memberikan harga mahal yang sebenarnya tidak perlu untuk biaya BBM saja atau bahkan merugi karena tingginya kebutuhan akan BBM. Salah satu strategi yang dapat dilakukan adalah dengan memilih urutan penjemputan yang memiliki jarak total terpendek. Untuk memperolehnya, kita dapat menggunakan algoritma *branch and bound*. Ada dua cara penyelesaian permasalahan ini dengan algoritma branch and bound. Yang akan digunakan untuk permasalahan ini adalah algoritma *branch and bound* dengan *reduced cost matrix*.

II. DASAR TEORI

A. Definisi Algoritma *Branch and bound*

Algoritma brach and bound (B&B) adalah algoritma atau metode pencarian di dalam ruang solusi secara sistematis. Ruang solusi digambarkan sebagai pohon ruang status. Urutan pembangkitan simpul pada algoritma brach and bound ini sama dengan urutan pembangkitan simpul dengan menggunakan algoritma BFS (Breadth First Search). Hal yang membedakan kedua algoritma tersebut adalah urutan pembangkitan simpulnya. Pada algoritma BFS biasa, simpul dibangkitkan sesuai urutan yang telah disepakati atau ditentukan sebelumnya dengan menggunakan prinsip queue (First In First Out). Pada algoritma brach and bound, urutan pembangkitan simpul ialah berdasarkan *cost* dari masing-masing simpul. Simpul yang memiliki *cost* terkecil (biasanya) dibangkitkan terlebih dahulu.

B. Prinsip Pencarian Solusi

Algoritma brach and bound ialah perbaikan atau pengembangan dari algoritma BFS. Algoritma brach and bound ialah gabungan dari BFS ditambah dengan fungsi least *cost* search, sehingga pada umumnya algoritma brach and bound lebih cepat dalam mencari solusi bila dibandingkan dengan algoritma BFS biasa. Hal ini disebabkan karena pada algoritma brach and bound, urutan pembangkitan simpul dilakukan berdasarkan *cost* dari masing-masing simpulnya. Simpul yang memiliki *cost* termurah atau terkecil yang akan dibangkitkan terlebih dahulu. Demikian seterusnya, sehingga simpul yang memiliki *cost* yang besar tidak perlu dibangkitkan karena kemungkinan besar solusi yang diharapkan tidak ada di cabang tersebut.

Untuk setiap simpul pada pohon, dihitung *cost*nya masing-masing yang menyatakan nilai batas (bound). Nilai batas dapat beragam tergantung kebutuhan permasalahan yang akan diselesaikan dengan algoritma brach and bound. Beberapa contoh nilai batas yang mungkin digunakan adalah :

- Jumlah simpul dalam upapohon X yang perlu dibangkitkan sebelum simpul solusi ditemukan
- Panjang lintasan dari simpul X ke simpul solusi terdekat,
- Dan sebagainya.

Namun, yang menjadi permasalahan dalam menentukan nilai batasnya adalah pada persoalan pada umumnya tidak diketahui letak simpul solusinya. Oleh karena itu, pada prakteknya, nilai batas untuk setiap simpul umumnya hanya berupa taksiran atau perkiraan saja. Fungsi heuristik yang dapat digunakan ialah :

$$\hat{c}(i) = \hat{f}(i) + \hat{g}(i)$$

Keterangan :

- $\hat{c}(i)$: ongkos untuk simpul i
- $\hat{f}(i)$: ongkos untuk mencapai simpul i dari akar
- $\hat{g}(i)$: ongkos mencapai simpul tujuan dari simpul 1

Nilai $\hat{c}(i)$ digunakan untuk mengurutkan pencarian. Simpul yang dibangkitkan adalah simpul yang memiliki $\hat{c}(i)$ terkecil. Berikut adalah algoritma *branch and bound* :

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 $\hat{c}(i)$ terkecil. Jika terdapat beberapa simpul yang memiliki $\hat{c}(i)$-nya minimal, maka pilih satu simpul secara sembarang.
4. Jika simpul i adalah simpul solusi, berarti solusi sudah ditemukan. Stop. Jika simpul i bukan simpul solusi, maka maka bangkitkan semua anak-anaknya. Jika simpul i tidak mempunyai anak, maka kembali ke langkah no 2.
5. Untuk setiap anak j dari simpul i, hitung $\hat{c}(j)$, dan masukkan semua anak-anak tersebut ke dalam antrian Q.
6. Kembali ke langkah 2.

C. Aplikasi dan Penerapan

Algoritma *branch and bound* dapat diaplikasikan untuk berbagai persoalan. Diantaranya adalah untuk menyelesaikan permasalahan permainan 15-Puzzle, persoalan n-ratu, pencarian shortest path (TSP), penjadwalan, dan lain sebagainya. Pada makalah ini, permasalahan yang akan dibahas adalah permasalahan TSP. Ada 2 cara penyelesaian permasalah TSP, yaitu *reduced cost matrix* dan *branch and bound* dengan bobot tur lengkap.

D. *Reduced cost matrix*

Buat sebuah matriks berbobot yang berisikan jarak dari masing-masing tempat ke tempat lainnya. Lalu dari matriks tersebut, cari matriks tereduksinya. Matriks tereduksi ialah matriks yang memiliki minimal 1(satu) nilai 0 pada masing-masing baris dan kolomnya. Caranya dengan melakukan pengurangan dengan angka yang paling kecil pada masing-masing baris atau kolomnya jika di baris atau kolom tersebut tidak ada angka 0. Dan catat seluruh jumlah pengurangan tersebut sebagai *cost* awal matriks tersebut.

Lalu dari matriks akar tersebut, bangkitkan anak-anak simpulnya. Misalkan i titik asal dan j adalah titik tujuan, maka akan dicari matriks tereduksi dengan cara sebagai berikut :

1. Semua elemen matriks pada baris i dan kolom j diganti sehingga bernilai ∞
2. Elemen matriks di (j,1) diubah juga menjadi bernilai ∞
3. Matriks tersebut direduksi kembali

Hitung lagi *cost* masing-masing simpul dengan rumus sebagai berikut :

$$\hat{c}(S) = \hat{c}(R) + A(i,j) + r$$

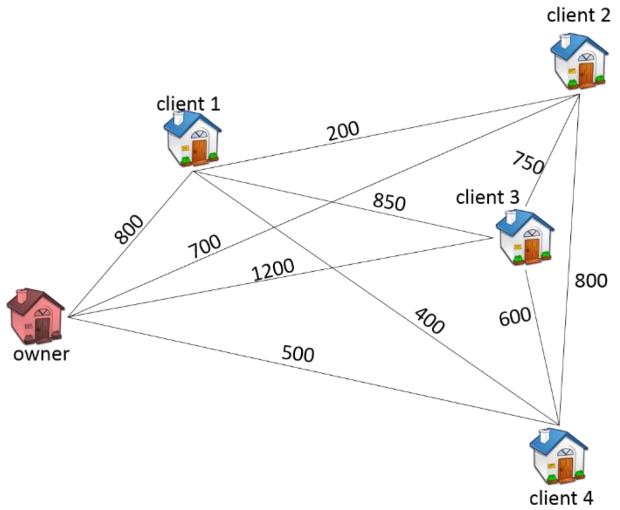
Keterangan :

- $\hat{c}(S)$: bobot perjalanan minimum yang melalui simpul S (simpul di pohon ruang status)
- $\hat{c}(R)$: bobot perjalanan minimum yang melalui simpul R, yang dalam hal ini R adalah orang tua dari S.
- A(i,j) : bobot sisi (i,j) pada graf G yang berkorespondensi dengan sisi (R,S) pada pohon ruang status.
- r : jumlah semua pengurang pada proses memperoleh matriks tereduksi untuk simpul S.

Setelah memperoleh seluruh *cost* dari masing-masing anak simpulnya, maka pilih satu simpul yang memiliki *cost* paling kecil untuk dibangkitkan lagi anak-anak simpulnya jika simpul tersebut bukan simpul solusi. Namun, jika simpul tersebut sudah berupa simpul solusi, maka anak-anak simpul tidak perlu dibangkitkan lagi dan solusi permasalahan berhasil ditemukan.

III. PENJELASAN

Misalkan ada 4 tempat yang harus dikunjungi setiap harinya dengan jarak yang berbeda-beda ke setiap tempatnya, maka dibuatlah sebuah matriks yang merepresentasikan jarak dari satu tempat ke semua tempat lainnya. Simpul *owner* dianggap merupakan simpul paling awal dan paling akhir yang harus dikunjungi.



Gambar 1. Jarak masing-masing tempat ke tempat lain

$$R = \begin{bmatrix} \infty & 800 & 700 & 1200 & 500 \\ 800 & \infty & 200 & 850 & 400 \\ 700 & 200 & \infty & 750 & 800 \\ 1200 & 850 & 750 & \infty & 600 \\ 500 & 400 & 800 & 600 & \infty \end{bmatrix}$$

Dari matriks diatas, dicari matriks tereduksinya, yaitu :

$$A = \begin{bmatrix} \infty & 300 & 200 & 500 & 0 \\ 500 & \infty & 0 & 450 & 200 \\ 400 & 0 & \infty & 350 & 600 \\ 500 & 250 & 150 & \infty & 0 \\ 0 & 0 & 400 & 0 & \infty \end{bmatrix}$$

Dengan r = 500+200+200+600+400+100+200. Jadi, *cost* untuk matriks akar adalah 2200.

Setelah itu, anak-anak simpul dibangkitkan dengan cara yang sudah dijelaskan pada subbab dasar teori.

1. Simpul 2, lintasan 1,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & 450 & 200 \\ 50 & \infty & \infty & 0 & 250 \\ 500 & \infty & 150 & \infty & 0 \\ 0 & \infty & 400 & 0 & \infty \end{bmatrix}$$

$$\hat{c}(2) = 2200 + 300 + 350 = 2850$$

2. Simpul 3, lintasan 1,3

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 300 & \infty & \infty & 250 & 0 \\ \infty & 0 & \infty & 350 & 600 \\ 500 & 250 & \infty & \infty & 0 \\ 0 & 0 & \infty & 0 & \infty \end{bmatrix}$$

$$\hat{c}(3) = 2200 + 200 + 200 = 2600$$

3. Simpul 4, lintasan 1,4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 500 & \infty & 0 & \infty & 200 \\ 400 & 0 & \infty & \infty & 600 \\ \infty & 250 & 150 & \infty & 0 \\ 0 & 0 & 400 & \infty & \infty \end{bmatrix}$$

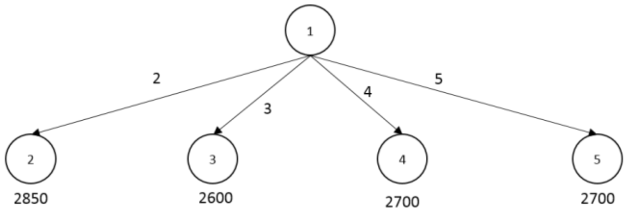
$$\hat{c}(4) = 2200 + 500 + 0 = 2700$$

4. Simpul 5, lintasan 1,5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 150 & \infty & 0 & 450 & \infty \\ 50 & 0 & \infty & 350 & \infty \\ 0 & 100 & 0 & \infty & \infty \\ \infty & 0 & 400 & 0 & \infty \end{bmatrix}$$

$$\hat{c}(5) = 2200 + 0 + 500 = 2700$$

Dari matriks-matriks diatas, jika dibentuk menjadi sebuah pohon maka akan menghasilkan pohon sebagai berikut



Dari seluruh simpul anak-anaknya yang sudah dibangkitkan, dapat kita lihat bahwa yang memiliki *cost* terkecil adalah simpul 3 yang merepresentasikan urutan dari *owner* ke *client* 2. Lalu, karena belum ditemukan solusi, maka simpul 3 yang memiliki *cost* terendah kembali dibangkitkan anak-anak simpulnya.

1. Simpul 2, lintasan 1,3,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 250 & 0 \\ \infty & \infty & \infty & \infty & \infty \\ 500 & \infty & \infty & \infty & 0 \\ 0 & \infty & \infty & 0 & \infty \end{bmatrix}$$

$$\hat{c}(6) = 2600 + 0 + 200 = 2800$$

2. Simpul 4, lintasan 1,3,4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 300 & \infty & \infty & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty \\ \infty & 250 & \infty & \infty & 0 \\ 0 & \infty & \infty & \infty & \infty \end{bmatrix}$$

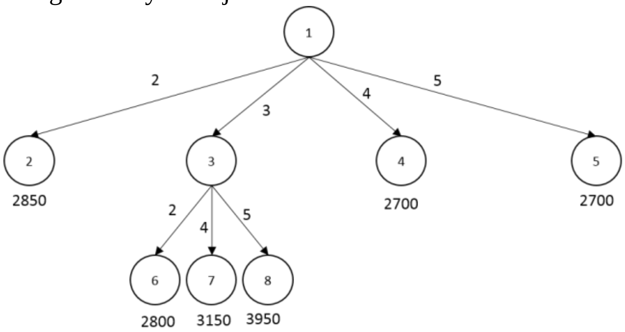
$$\hat{c}(7) = 2600 + 350 + 200 = 3150$$

3. Simpul 5, lintasan 1,3,5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & 0 & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 200 & 0 & \infty & \infty & \infty \\ \infty & 0 & \infty & 0 & \infty \end{bmatrix}$$

$$\hat{c}(8) = 2600 + 600 + 750 = 3950$$

Dari perhitungan di atas, maka bentuk pohon ruang solusinya menjadi :



Karena yang memiliki *cost* paling sedikit adalah simpul 4 dan 5, maka pilih salah satu simpul secara sembarang untuk dibangkitkan anak-anak simpulnya.

1. Simpul 2, lintasan 1,4,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & \infty & 0 \\ 0 & \infty & \infty & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & 400 & \infty & \infty \end{bmatrix}$$

$$\hat{c}(9) = 2700 + 250 + 600 = 3550$$

2. Simpul 3, lintasan 1,4,3

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 300 & \infty & \infty & \infty & 0 \\ \infty & 0 & \infty & \infty & 600 \\ \infty & \infty & \infty & \infty & \infty \\ 0 & 0 & \infty & \infty & \infty \end{bmatrix}$$

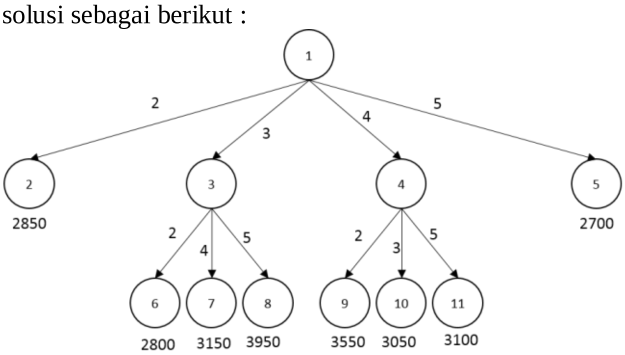
$$\hat{c}(10) = 2700 + 150 + 200 = 3050$$

3. Simpul 5, lintasan 1,4,5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 100 & \infty & 0 & \infty & \infty \\ 0 & 0 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & 0 & 400 & \infty & \infty \end{bmatrix}$$

$$\hat{c}(11) = 2700 + 0 + 400 = 3100$$

Dari perhitungan di atas, diperoleh pohon ruang solusi sebagai berikut :



Karena simpul 5 memiliki *cost* paling kecil, maka simpul 5 dibangkitkan anak-anaknya.

1. Simpul 2, lintasan 1,5,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & 450 & \infty \\ 0 & \infty & \infty & 0 & \infty \\ 300 & \infty & 0 & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

$$\hat{c}(12) = 2700 + 0 + 550 = 3250$$

2. Simpul 3, lintasan 1,5,3

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & 0 & \infty \\ \infty & 0 & \infty & 350 & \infty \\ 200 & 0 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

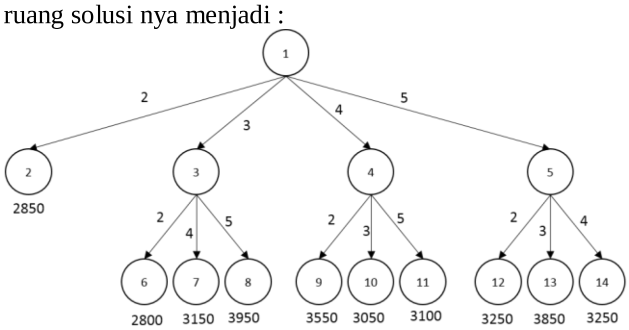
$$\hat{c}(13) = 2700 + 400 + 750 = 3850$$

3. Simpul 4, lintasan 1,5,4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 100 & \infty & 0 & \infty & \infty \\ 0 & 0 & \infty & \infty & \infty \\ \infty & 100 & 0 & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

$$\hat{c}(14) = 2700 + 0 + 550 = 3250$$

Dari perhitungan di atas, maka bentuk pohon ruang solusi nya menjadi :



Karena simpul 6 memiliki *cost* terkecil, maka simpul 6 yang dibangkitkan anak-anak simpulnya.

1. Simpul 4, lintasan 1,3,2,4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 0 \\ 0 & \infty & \infty & \infty & \infty \end{bmatrix}$$

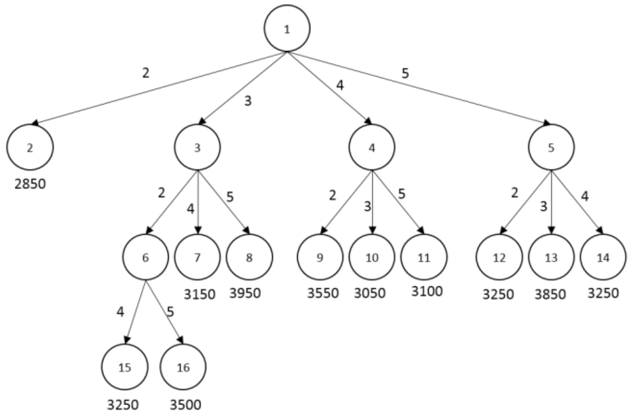
$$\hat{c}(15) = 2800 + 450 + 0 = 3250$$

2. Simpul 5, lintasan 1,3,2,5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & \infty \end{bmatrix}$$

$$\hat{c}(16) = 2800 + 200 + 500 = 3500$$

Dari perhitungan di atas, maka diperoleh pohon ruang solusi sebagai berikut:



Karena simpul 2 memikiki *cost* terkecil, maka simpul 2 dibangkitkan anak-anak simpulnya.

1. Simpul 3, lintasan 1,2,3

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & 250 \\ 500 & \infty & \infty & \infty & 0 \\ 0 & \infty & \infty & 0 & \infty \end{bmatrix}$$

$$\hat{c}(17) = 2850 + 0 + 350 = 3200$$

2. Simpul 4, lintasan 1,2,4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & 200 \\ \infty & \infty & 0 & \infty & 0 \\ 0 & \infty & 250 & \infty & \infty \end{bmatrix}$$

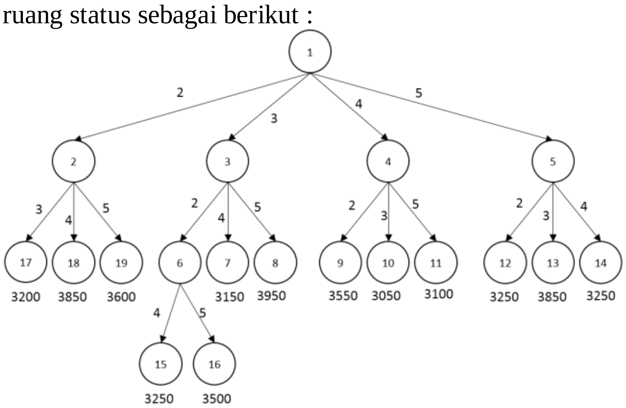
$$\hat{c}(18) = 2850 + 450 + 550 = 3850$$

3. Simpul 5, lintasan 1,2,5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & 0 & \infty \\ 300 & \infty & 0 & \infty & \infty \\ \infty & \infty & 400 & 0 & \infty \end{bmatrix}$$

$$\hat{c}(19) = 2850 + 200 + 550 = 3600$$

Dari perhitungan di atas, maka akan diperoleh pohon ruang status sebagai berikut :



Karena simpul 10 memiliki *cost* terendah, maka simpul 10 dibangkitkan lagi anak-anak simpulnya.

1. Simpul 2, lintasan 1,4,3,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 300 & \infty & \infty & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \end{bmatrix}$$

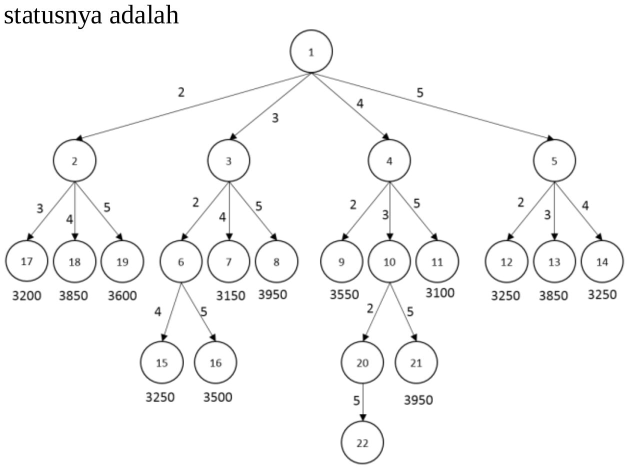
$$\hat{c}(20) = 3050 + 0 + 0 = 3050$$

2. Simpul 5, lintasan 1,4,3,5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & 0 & \infty & \infty & \infty \end{bmatrix}$$

$$\hat{c}(21) = 3050 + 600 + 300 = 3950$$

Karena yang memiliki *cost* terkecil adalah simpul no 20, maka simpul tersebut dibangkitkan anak simpulnya, yaitu simpul 5. Sehingga pohon ruang statusnya adalah



Jadi dapat ditarik sebuah solusi untuk mendapatkan jarak terpendek, rute yang harus ditempuh adalah 1-4-3-2-5, yaitu : *owner* – *client* 3 – *client* 2 – *client* 1 – *client* 4 – sekolah/kantor – *client* 4 – *client* 1-*client* 2 – *client* 3 – *owner* dengan total jarak adalah 2*(3050)+jarak ke tempat tujuan (sekolah/kantor).

Misalkan rute yang dilalui adalah 1-5-3-4-2, maka jarak yang ditempuh menjadi 2*(3700)+jarak ke tempat tujuan. Selisih jarak yang ditempuh jika menggunakan algoritma *branch and bound* dengan yang tidak menggunakan algoritma ialah sebesar 1300 meter atau 1,3 km setiap harinya. Jika dilakukan selama 1 bulan dengan asumsi 25 hari kerja, maka perbedaa jarak tempuh akan sebesar (1,3 * 25) = 32,5 km. Asumsikan bahwa kendaraan yang digunakan mengonsumsi bahan bakar sebanyak 8L/100km, maka penggunaan BBM hemat sebesar 2,6 L setiap bulannya.

IV. KESIMPULAN

- Penghematan penggunaan BBM dapat dicapai dengan melewati rute-rute yang menghasilkan jarak total minimum.
- Pencarian jarak minimum dapat dicari salah satunya dengan menggunakan algoritma *branch and bound* dengan metode *reduced cost matrix*.
- Dari contoh persoalan yang dibahas pada makalah ini,

berhasil dilakukan penghematan sebanyak 2,6 L setiap bulannya. Jika hal ini dilakukan setiap hari selama bertahun-tahun maka penghematan BBM ini akan sangat terasa perbedaannya jika tidak rute-rute yang mengahasilkan jarak terpendek.

DAFTAR PUSTAKA

[1] Munir, Rinaldi. Strategi Algoritma. Bandung, Informatika Bandung, 2009.

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, 18 Mei 2014

Chrestella Stephanie
13512005