



PENERAPAN ALGORTIMA *BRANCH AND BOUND* UNTUK MENENTUKAN RUTE OBJEK WISATA DI KOTA SEMARANG

Fera Marlinda Gurnitowati[✉], Rochmad, Supriyono.

Jurusan Matematika, FMIPA, Universitas Negeri Semarang, Indonesia
Gedung D7 lantai 1 Kampus Sekaran, Gunungpati, Semarang, 50229

Info Artikel

Sejarah Artikel:
Diterima Desember 2013
Disetujui Januari 2014
Dipublikasikan Mei 2014

Keywords :
Algortima *Branch and Bound*
Jalur/Rute
Travelling Salesman Problem
(TSP)

Abstrak

Penelitian ini mengkaji sebuah permasalahan pencarian solusi optimum untuk masalah *Travelling Salesman Problem* (TSP). Tujuan dari penelitian ini adalah meneliti salah satu kasus TSP pada masalah penentuan rute objek wisata di Kota Semarang menggunakan algoritma *Branch and Bound* dan diaplikasikan ke dalam program winQSB. Pengambilan data dilakukan dengan cara melihat peta Kota Semarang yang diperoleh dari Dinas Kebudayaan dan Pariwisata Kota Semarang. Data yang diambil berupa beberapa lokasi wisata, selanjutnya dilakukan pencarian jarak dengan bantuan *Google Maps*. Analisis data dilakukan dengan menggunakan mekanisme algoritma *Branch and Bound* yang kemudian diaplikasikan dalam program winQSB. Dari analisis yang telah dilakukan, dengan cara perhitungan manual maupun dengan bantuan program winQSB rute terpendek yang dapat diperoleh yaitu melalui jalur Masjid Agung Jawa Tengah-Water Blaster-Simpang Lima-Lawang Sewu-Sam Poo Kong-Taman Margasatwa Mangkang-Gereja Blenduk dan kembali ke Masjid Agung Jawa Tengah dengan jarak minimum 53,2 kilometer dalam sekali tempuh. Dari hasil analisis, diharapkan biro perjalanan yang ada di Kota Semarang dapat menerapkan metode perhitungan rute optimal agar dapat mengetahui jalur terpendek dari sejumlah objek wisata.

Abstract

This study examines an optimum solution to the search the problem of Travelling Salesman Problem (TSP). The purpose of this study is to examine one case of TSP on the problem of determining the attractions in the city of Semarang using Branch and Bound algorithm and applied to the program winQSB. Data retrieval is done by looking at a map of Semarang city obtained from the department of culture and tourism Semarang. Data taken in the form of some tourist sites, the search is then performed with the help of Google Maps distance. Data analysis was performed using the mechanism of the Branch and Bound algorithm is then applied in winQSB program. Of analysis has been done, by way of manual calculations and with the help of the program winQSB shortest route that can be obtained is through the Great Mosque of Central Java-Water Blaster-Simpang Lima-Lawang Sewu-Sam Poo Kong-Wildlife Areas Mangkang-Blenduk Church and return to the Great Mosque of Central Java with a minimum distance of 53,2 kilometers in one go. From the analysis, it is expected that the travel agency in the city of Semarang can apply the calculation method in order to determine the optimal shortest path from a number of attractions .

© 2014 Universitas Negeri Semarang

Pendahuluan

Objek wisata merupakan tempat yang sering dikunjungi oleh wisatawan yang ingin rekreasi mencari hiburan atau hanya sekadar melepas kepenatan dari aktifitas sehari-hari. Banyak biro perjalanan yang menyediakan jasa perjalanan objek wisata ke beberapa tempat atau dalam bentuk paket wisata. Dalam pencarian rute terpendek merupakan masalah yang rumit dipandang dari segi komputasinya. Salah satu masalah pencarian rute perjalanan terpendek adalah mencari rute terpendek dari sejumlah objek wisata dan jarak antar objek wisata yang harus dilalui oleh wisatawan bila wisatawan itu berangkat dari sebuah biro perjalanan dan menyinggahi setiap tempat objek wisata tepat satu kali dan kembali lagi ke tempat biro perjalanan. Berdasarkan uraian permasalahan di atas biasa disebut dengan *Travelling Salesman Problem* (TSP). Tujuan penelitian ini adalah untuk meneliti salah satu kasus TSP pada masalah penentuan rute objek wisata di wilayah Kota Semarang menggunakan algoritma *Branch and Bound* dan diaplikasikan dengan menggunakan program winQSB.

Menurut Rosa (2012) TSP adalah masalah optimasi, yaitu mengunjungi setiap tempat dari himpunan tempat-tempat yang ditentukan sekali dan hanya satu kali kemudian kembali ke tempat awal pada akhir dari rute perjalanan dengan jarak, waktu dan biaya yang minimum. Dalam menemukan rute perjalanan terpendek untuk agen travel, terdapat beberapa parameter yang perlu ditentukan sebelumnya untuk menentukan nilai ongkos (*cost*) yaitu jarak antara titik awal ke alamat pengunjung (titik tujuan) dan jarak antara alamat penumpang satu ke penumpang lainnya.

Titik tujuan dari tiap penumpang akan diasosiasikan sebagai simpul dan titik awal sebagai akar. Peta yang menggambarkan lokasi dari titik awal dan titik-titik tujuan penumpang akan direpresentasikan dalam bentuk graf lengkap.

Pemberian nilai ongkos akan dilakukan dengan menggunakan matriks tereduksi dari matriks jarak antar simpul yang dibentuk dari graf G. Matriks tereduksi adalah matriks yang tiap kolom dan tiap barisnya mengandung paling sedikit satu buah angka 0 dan elemen-elemen lainnya bernilai *non-negatif*. Untuk mendapatkan matriks tereduksi, maka tiap baris atau kolom yang belum mengandung angka 0 dikurangi dengan nilai terkecil pada baris atau kolom tersebut.

Semua angka yang digunakan untuk mengurangi tiap baris atau kolom tersebut kemudian dijumlahkan. Hasil dari penjumlahan inilah yang kemudian dijadikan sebagai $\hat{c}(\text{root})$ atau nilai ongkos dari simpul awal atau akar. Hal ini juga berarti bahwa solusi pada persoalan TSP tersebut paling tidak memiliki bobot minimum sebesar nilai $\hat{c}(\text{root})$ yang diperoleh tersebut.

Selanjutnya, misal A adalah matriks tereduksi untuk simpul R. Misalkan S adalah anak dari simpul R sehingga sisi (R,S) pada pohon ruang status berkorespondensi dengan sisi (i,j), maka dilakukan langkah-langkah berikut.

1. Mengubah semua nilai pada baris i dan kolom j menjadi ∞ ,
2. Mengubah $A(i,j)$ menjadi ∞ ,
3. Mereduksi kembali matriks A.

Reduksi matriks A akan menghasilkan matriks lain (misal matriks B) dan fungsi pembatas. Secara umum, persamaan fungsi pembatas adalah seperti persamaan 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 rentang 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

Penelitian ini dilakukan dengan menggunakan algoritma *Branch and Bound*. Menurut Nugraha (2010) Algoritma *Branch and Bound* adalah suatu algoritma pencarian di dalam ruang solusi secara sistematis. Ruang solusi digambarkan ke dalam pohon ruang status. Pohon ruang status tersebut dibangun dengan skema BFS.

Menurut Wijaya (2007) *Breadth First Search* (BFS) adalah algoritma pencarian simpul dalam graf (pohon) secara traversal yang dimulai dari simpul akar dan mengecek semua simpul-simpul tetangganya. Setelah itu, dari tiap simpul tetangganya, algoritma akan terus mengecek semua simpul tetangga yang belum

dicek sedemikian sehingga menemukan simpul tujuan. Untuk mempercepat pencarian solusi maka simpul berikut diekspansi tidak berdasarkan urutan pembangkitnya tetapi berdasarkan cost yang dimiliki oleh tiap simpul-simpul. Cost pada setiap simpul i menyatakan perkiraan ongkos termurah lintasan dari simpul i ke simpul solusi.

Berikut adalah langkah-langkah pencarian solusi dengan *Branch and Bound*.

1. Simpul akar dimasukan ke dalam antrian Q. Jika simpul akar adalah simpul solusi, maka solusi telah ditentukan dan pencarian berhenti.
2. Antrian Q diidentifikasi.
 - a. Jika antrian Q kosong, maka solusi tidak ada dan pencarian berhenti.
 - b. Jika antrian Q tidak kosong, maka dipilih antrian Q simpul i yang mempunyai nilai c paling kecil. Jika terdapat beberapa simpul i yang memenuhi, maka dipilih satu secara sembarang.
 3. Simpul i diidentifikasi.
 - a. Jika simpul i adalah simpul solusi, maka solusi telah ditemukan dan pencarian berhenti.
 - b. Jika simpul i bukan simpul solusi, maka semua anaknya dibangkitkan. Jika simpul tidak memiliki anak, maka kembali ke langkah 2.
 4. Untuk setiap anak j dari simpul i , nilai c dihitung dan semua anak yang sudah dibangkitkan dimasukkan ke dalam antrian Q.
 5. Kembali ke langkah 2.

Kemudian dilakukan penelitian dan pengambilan data sekunder dari Kantor Dinas Kebudayaan dan Pariwisata Kota Semarang serta dengan menggunakan bantuan Google Maps. Menurut Jurnal Sarjana Teknik oleh Mahdia & Fiftin (2013) *Google Maps* adalah sebuah jasa global *virtual* gratis dan *online* yang disediakan oleh perusahaan Google. *Google Maps* menawarkan peta yang dipaparkan dan gambar satelit untuk seluruh dunia, selain itu *Google Maps* juga menawarkan peta jalan, sebuah rute perjalanan untuk perencanaan menggunakan jalan kaki, mobil, atau angkutan umum dan pencari tempat bisnis untuk berbagai negara di seluruh dunia. Menurut salah satu pembuat *Google Maps* (Lars Rasmussen), *Google Maps* adalah salah satu cara untuk mengorganisir informasi dunia secara geografis. Berdasarkan hasil penelitian yang telah

dilakukan, *Google Maps* menghasilkan koordinat yang cukup akurat. Hal itu mengakibatkan hasil pencarian jarak antar lokasi menjadi lebih tepat.

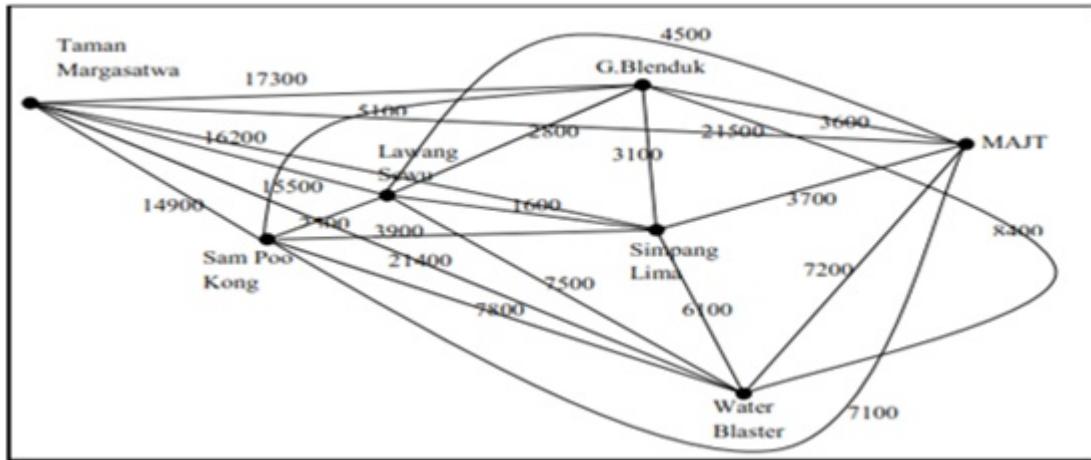
Setelah diperoleh data, data tersebut akan dianalisis kembali dengan menggunakan software WinQSB. Menurut Winarno (2008) Salah satu program komputer yang dirancang untuk memecahkan masalah-masalah kuantitatif di bidang manajemen adalah WinQSB. Salah satu keunggulan program WinQSB adalah merupakan kelanjutan dari program QSB yang pada tahun 1990-an sudah banyak digunakan di kalangan akademisi maupun di kalangan pembuat keputusan. Bedanya QSB masih dijalankan di bawah sistem operasi DOS, sedangkan WinQSB sudah berbasis *Microsoft Windows*. Karena sudah berbasis *Microsoft Windows*, WinQSB sudah memanfaatkan kemampuan grafis komputer. Masing-masing modul yang ada di program ini sudah memiliki ikon sendiri. Kelebihan lain WinQSB adalah adanya 19 model terpisah, sehingga memudahkan pengguna untuk mempelajari modul-modul yang lain.

Metode Penelitian

Metode penelitian yang digunakan adalah metode studi kasus, dimana peneliti melakukan penelitian untuk mengumpulkan data-data yang diperlukan untuk penelitian ini. Subjek dalam penelitian adalah algoritma *branch and bound* untuk menyelesaikan masalah penentuan rute objek wisata. Sumber data yang didapatkan diperoleh dari Dinas Kebudayaan dan Pariwisata Kota Semarang dan dengan bantuan *Google Maps*, selain itu mengumpulkan beberapa literatur yaitu berupa, jurnal, buku, dan literatur ilmiah lainnya yang mendukung penelitian ini. Metode yang digunakan adalah metode analisis hasil penyelesaian masalah penentuan rute objek wisata secara manual menggunakan algoritma *branch and bound* dan diaplikasikan dengan bantuan program winQSB.

Hasil Penelitian dan Pembahasan

Berdasarkan data yang ada, selanjutnya yang dilakukan adalah mencari rute terbaik dengan jarak tempuh terpendek dari satu tempat wisata ke tempat wisata lainnya yang ada di Kota Semarang. Proses pencarian rute menggunakan algoritma *Branch and Bound* akan dijelaskan pada kasus di bawah ini dengan 7 tempat lokasi yang ada di Kota Semarang.



Gambar 1. Graf yang menggambarkan titik-titik tujuan

Berdasarkan graf di atas dapat diubah kedalam bentuk matriks berukuran 7×7 . Matriks 7×7 dimana elemen M_{ij} adalah jarak dari i ke j, sedangkan i dan j adalah simpul pada graf.

$$\begin{bmatrix} \infty & 4500 & 3600 & 7100 & 21500 & 7200 & 3700 \\ 4500 & \infty & 2800 & 2500 & 15500 & 7500 & 1600 \\ 3600 & 2800 & \infty & 5100 & 17300 & 8400 & 3100 \\ 7100 & 2500 & 5100 & \infty & 14900 & 7800 & 3900 \\ 21500 & 15500 & 17300 & 14900 & \infty & 21400 & 16200 \\ 7200 & 7500 & 8400 & 7800 & 21400 & \infty & 6100 \\ 3700 & 1600 & 3100 & 3900 & 16200 & 6100 & \infty \end{bmatrix}$$

Matriks di atas merupakan matriks simetris karena jarak i ke j sama dengan jarak j ke i. Dalam penyerdehaan matriks di atas dilakukan dengan cara mereduksi matriks tersebut.

Reduksi baris :

$$\begin{bmatrix} \infty & 900 & 0 & 3500 & 17900 & 3600 & 100 \\ 2900 & \infty & 1200 & 900 & 13900 & 5900 & 0 \\ 800 & 0 & \infty & 2300 & 14500 & 5600 & 300 \\ 4600 & 0 & 2600 & \infty & 12400 & 5300 & 1400 \\ 6600 & 600 & 2400 & 0 & \infty & 6500 & 1300 \\ 1100 & 1400 & 2300 & 1700 & 15300 & \infty & 0 \\ 2100 & 0 & 1500 & 2300 & 14600 & 4500 & \infty \end{bmatrix}$$

Matriks di atas dihasilkan dari pengurangan tiap baris dengan nilai terkecil pada elemen baris tersebut. Baris ke-1 dikurangi 3600, baris ke-2 dikurangi 1600, baris ke-3 dikurangi 2800, baris ke-4 dikurangi 2500, baris ke-5 dikurangi 14900, baris ke-6 dikurangi 6100 dan baris ke-7 dikurangi 1600.

Reduksi kolom :

$$\begin{bmatrix} \infty & 900 & 0 & 3500 & 5500 & 0 & 100 \\ 2100 & \infty & 1200 & 900 & 1500 & 2300 & 0 \\ 0 & 0 & \infty & 2300 & 2100 & 2000 & 300 \\ 3800 & 0 & 2600 & \infty & 0 & 1700 & 1400 \\ 5800 & 600 & 2400 & 0 & \infty & 2900 & 1300 \\ 300 & 1400 & 2300 & 1700 & 2900 & \infty & 0 \\ 1300 & 0 & 1500 & 2300 & 2200 & 900 & \infty \end{bmatrix}$$

Matriks di atas dihasilkan dari pengurangan seluruh elemen pada kolom 1 dengan 800, kolom 5 dengan 12400, dan kolom 6 dengan 3600. Jadi diperoleh $\hat{c}(\text{root}) = 3600 + 1600 + 2800 + 2500 + 14900 + 6100 + 1600 + 800 + 12400 + 3600 = 49900$. Dengan demikian berarti telah dibangkitkan pohon ruang status yang baru berisi satu buah simpul dengan bobot 49900.



49900

Selanjutnya menghitung simpul-simpul lain pada pohon ruang status dengan mengacu pada persamaan yang telah dijelaskan di atas.

1. Simpul 2; lintasan 1,2

$$\begin{aligned} & \begin{bmatrix} \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 1200 & 900 & 1500 & 2300 \\ 0 & 0 & \infty & 2300 & 2100 & 2000 \\ 3800 & 0 & 2600 & \infty & 0 & 1700 \\ 5800 & 0 & 2400 & 0 & \infty & 2900 \\ 300 & 0 & 2300 & 1700 & 2900 & \infty \\ 1300 & 0 & 1500 & 2300 & 2200 & 900 \end{bmatrix} \\ & = \begin{bmatrix} \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 600 & 900 & 1500 & 2700 \\ 0 & 0 & \infty & 2000 & 1800 & 2400 \\ 3800 & 0 & 2000 & \infty & 0 & 2100 \\ 5800 & 0 & 1800 & 0 & \infty & 3300 \\ 300 & 0 & 1700 & 1700 & 2900 & \infty \\ 0 & \infty & 0 & 1300 & 900 & 0 \end{bmatrix} \end{aligned}$$

Matriks di atas diperoleh dari pengurangan baris ke-7 oleh 900, dan kolom ke-3 oleh 600. Jadi $\hat{e}(2) = 49900 + 900 + 1500 = 52300$.

2. Simpul 3; lintasan 1,3

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan kolom ke-1 oleh 300 dan ke-6 oleh 900. Jadi $\hat{e}(3) = 49900 + 0 + 1200 = 51100$.

3. Simpul 4; lintasan 1,4

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan baris ke-5 oleh 600, kolom ke-3 oleh 1200, kolom ke-6 oleh 900. Jadi $\hat{e}(4) = 49900 + 3500 + 2700 = 56100$.

4. Simpul 5; lintasan 1,5

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan kolom ke-3 oleh 1200, dan kolom ke-6 oleh 900. Jadi $\hat{e}(5) = 49900 + 5500 + 2100 = 57500$.

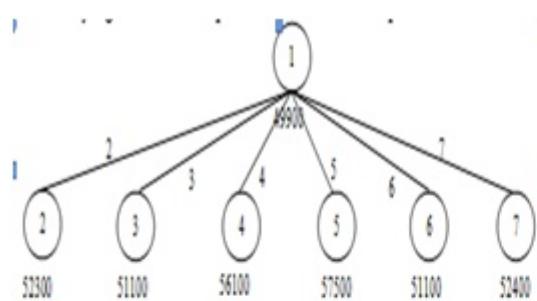
5. Simpul 6; lintasan 1,6

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan kolom ke-3 oleh 1200. Jadi $\hat{e}(6) = 49900 + 0 + 1200 = 51100$.

6. Simpul 7; lintasan 1,7

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan baris ke-2 oleh 900, baris ke-6 oleh 300, kolom ke-3 oleh 300 dan kolom ke-6 oleh 900. Jadi $\hat{e}(7) = 49900 + 100 + 2400 = 52400$.

Pohon status yang terbentuk dari proses reduksi di atas sampai saat ini adalah sebagai berikut.



Gambar 2. Pohon status pertama yang terbentuk

Dengan penomoran simpul sama seperti sebelumnya, yaitu : (1) Masjid Agung Jawa Tengah, (2) Lawang Sewu, (3) Gereja Blenduk, (4) Sam Poo Kong, (5) Taman Margasatwa, (6) Water Blaster, dan (7) Simpang Lima. Terdapat 2 simpul yang memiliki bobot terkecil yaitu simpul 3 dan simpul 6.

7. Simpul 8; lintasan 1,3,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 900 & 1500 & 1400 & 0 & 0 \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 3500 & \infty & \infty & 0 & 800 & 1400 & 0 \\ 5500 & \infty & \infty & 0 & \infty & 2000 & 1300 \\ 0 & \infty & \infty & 1700 & 2900 & \infty & 0 \\ 1000 & \infty & \infty & 2300 & 2200 & 0 & \infty \end{bmatrix}$$

Jadi $\hat{e}(8) = 51100 + 0 + 0 = 51100$.

8. Simpul 9; lintasan 1,3,4

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan baris ke-5 oleh 600. Jadi $\hat{e}(9) = 51100 + 2300 + 600 = 54000$.

9. Simpul 10; lintasan 1,3,5

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya, jadi $\hat{e}(10) = 51100 + 2100 + 0 = 53200$.

10. Simpul 11; lintasan 1,3,6

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan kolom ke-1 oleh 1000. Jadi $\hat{e}(11) = 51100 + 1100 + 1000 = 53200$.

11. Simpul 12; lintasan 1,3,7

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan baris ke-2 oleh 900. Jadi $\hat{e}(12) = 51100 + 300 + 900 = 52300$.

Selanjutnya ekspansi simpul 6.

12. Simpul 13; lintasan 1,6,2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 2100 & \infty & \infty & 900 & 1500 & \infty & 0 \\ \infty & 0 & \infty & 2300 & 2100 & \infty & 300 \\ 3800 & 0 & \infty & \infty & 0 & \infty & 1400 \\ 5800 & 600 & \infty & 0 & \infty & \infty & 1300 \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 1300 & 0 & \infty & 2300 & 2200 & \infty & \infty \end{bmatrix}$$

$$= \begin{bmatrix} \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 800 & \infty & \infty & 900 & 1500 & \infty & 0 \\ \infty & 0 & \infty & 2300 & 2100 & \infty & 300 \\ 2500 & 0 & \infty & \infty & 0 & \infty & 1400 \\ 4500 & 600 & \infty & 0 & \infty & \infty & 1300 \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 0 & 0 & \infty & 2300 & 2200 & \infty & \infty \end{bmatrix}$$

Matriks di atas diperoleh dari pengurangan baris ke-7 oleh 300. Jadi $\hat{c}(13) = 51100 + 1400 + 300 = 52800$.

13. Simpul 14; lintasan 1,6,3

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan kolom ke-1 oleh 1300. Jadi $\hat{c}(14) = 51100 + 1100 + 1300 = 53500$.

14. Simpul 15; lintasan 1,6,4

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya diperoleh dari pengurangan baris ke-5 oleh 600. Jadi $\hat{c}(15) = 51100 + 1700 + 600 = 53400$.

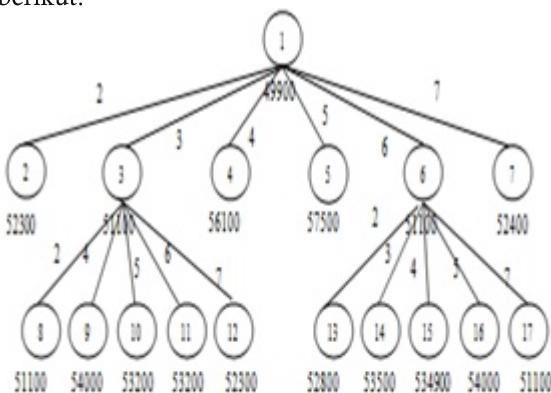
15. Simpul 16; lintasan 1,6,5

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya, jadi $\hat{c}(16) = 51100 + 2900 + 0 = 54000$.

16. Simpul 17; lintasan 1,6,7

Berdasarkan perhitungan matriks yang telah dilakukan sama dengan simpul sebelumnya, jadi $\hat{c}(17) = 51100 + 0 + 0 = 51100$.

Pohon status yang terbentuk dari proses reduksi di atas sampai saat ini adalah sebagai berikut.



Gambar 3. Pohon status kedua yang terbentuk

Sama seperti langkah-langkah yang telah dilakukan di atas hingga sampai didapatkan simpul ke-31, selanjutnya akan diperoleh pohon status baru sebagai berikut.

31. Simpul 32; lintasan 1,6,7,2,4,3

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

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

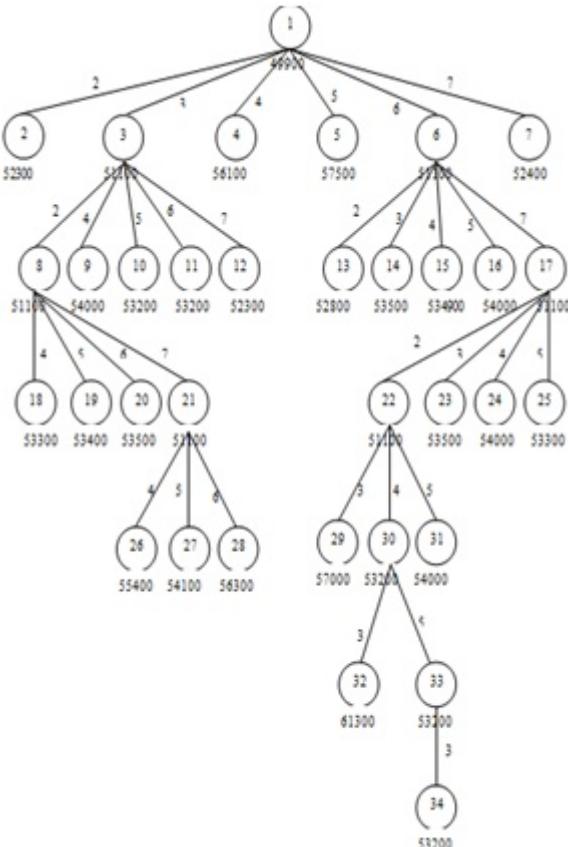
Matriks di atas diperoleh dari pengurangan baris ke-3 oleh 2100 dan kolom ke-1 oleh 5800. Jadi $\hat{c}(32) = 53200 + 200 + 7900 = 61300$.

32. Simpul 33; lintasan 1,6,7,2,4,5

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

Jadi $\hat{c}(33) = 53200 + 0 + 0 = 53200$.

Dari proses-proses reduksi di atas, diperoleh pohon status akhir seperti berikut.



Gambar 4. Pohon status kelima yang terbentuk

Jika dianalisis dengan menggunakan winQSB akan diperoleh hasil sebagai berikut.

11-18-2013	From Node	Connect To	Distance/Cost		From Node	Connect To	Distance/Cost
1	Masjid	Water	7200	5	Sam Poo	Taman	14900
2	Water	Simpang	6100	6	Taman	Gereja	17300
3	Simpang	Lawang	1600	7	Gereja	Masjid	3600
4	Lawang	Sam Poo	2500				
	Total	Minimal	Traveling	Distance	or Cost	=	53200
	(Result	from	Branch	and	Bound	Method)	

Gambar 5. Hasil Akhir program winQSB

Algoritma *Branch and Bound* sangat efektif digunakan dalam menentukan solusi optimum pada masalah TSP ataupun pencarian rute terpendek untuk wilayah Kota Semarang seperti yang telah dijelaskan di atas. Dengan algortima ini, dapat mencari kemungkinan solusi yang masih mungkin dan membuang kemungkinan lain yang tidak akan menghasilkan solusi optimum, sehingga pada akhirnya ditemukan solusi tersebut. Sama halnya dengan pencarian rute kota yang lain juga dapat dilakukan dengan cara yang sama, sehingga dengan demikian perjalanan yang dilakukan dapat efektif dan efisien mengingat jarak yang ditempuh adalah minimum.

Simpulan

Berdasarkan hasil analisis di atas dapat diperoleh jarak minimum dengan rute Masjid Agung Jawa Tengah-Water Blaster-Simpang Lima-Lawang Sewu-Sam Poo Kong-Taman Margasatwa Mangkang-Gereja Blenduk dan kembali ke Masjid Agung Jawa Tengah dengan jarak minimum 53,2 kilometer dalam sekali tempuh. Hasil Sehingga dapat disimpulkan bahwa algoritma *branch and bound* dapat digunakan untuk menyelesaikan persoalan TSP dengan permasalahan pencarian rute objek wisata yang ada di Kota Semarang.

Daftar Pustaka

- Mahdia, F, & N. Fiftin. 2013. *Pemanfaatan Google Maps API untuk Pembangunan Sistem Informasi Manajemen Bantuan Logistik Pasca Bencana Alam Berbasis Mobile WEB (Studi Kasus : Badan Penanggulangan Bencana Daerah Kota Yogyakarta)*. Jurnal Sarjana Teknik Informatika, e-ISSN: 2338-5197 Volume 1 (1):Juni 2013 . Tersedia di <http://jurnal.uad.ac.id/index.php/JSTIF/article/download/1696/1016> [diakses 11-11-2013].

Nugraha, M. P. 2010. *Penerapan Algoritma Branch and Bound dalam Menentukan Rute Objek Terpendek untuk Perjalanan antar Kota Di Jawa Barat*. *Journal of Informatics and Technology*. Tersedia di if17033@student.if.itb.ac.id [diakses 10-2-2013].

Rosa, W. R. 2012. *Penentuan Jalur Terpendek pada Pelayanan Agen Travel Khusus Pengantaran Wilayah Semarang Berbasis SIG dengan Algoritma Branch and Bound*. *Journal of Informatics and Technology*, Vol.1(1):63-71. Tersedia di <http://ejournals-s1.undip.ac.id/index.php/joint> [diakses 13-3-2013].

Wijaya, D. N. 2007. *Perbandingan Algortima BFS dan DFS Dalam Pembuatan Rute Perjalanan Objek Permainan 2 Dimensi*. Makalah Teknik Elektro dan Informatika. Tersedia di if2251@student.if.itb.ac.id [diakses 2-11-2013].

Winarno, W. W. 2008. Analisis Manajemen Kuantitatif dengan WinQSB. Yogyakarta: UPP STIM YKPN.