

Algoritma *Branch & Bound* untuk Optimasi Pengiriman Surat antar Himpunan di ITB

Mohamad Ray Rizaldy - 13505073

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jl. Ganesha 10, Bandung

e-mail: ravrizaldy@gmail.com

ABSTRAK

Seringkali mahasiswa di kampus Institut Teknologi Bandung kewalahan karena harus mengirimkan baik itu surat maupun poster publikasi jika Himpunan Mahasiswa di Program Studinya akan mengadakan acara. Pengambilan rute yang acak dan tanpa pikir panjang adalah salah satu penyebab utama fenomena di atas. Oleh karena itu dalam makalah ini akan dibahas sebuah cara untuk mereka yang akan bertugas sebagai seksi publikasi ataupun divisi eksternal di himpunannya sehingga bisa memudahkan mereka mengerjakan tugasnya secara optimal. Yang akan dipakai sebagai pembuat solusi optimal disini adalah algoritma *Branch and Bound*. Dimana dengan algoritma ini masalah *Travelling Salesman Problem* atau sering disingkat sebagai TSP yang direpresentasikan melalui graf berbobot bisa diselesaikan dengan skema *Breadth First Search* yang telah dimodifikasi melalui sebuah fungsi pembatas. Untuk memudahkan hanya akan digunakan 5 Himpunan Mahasiswa yang ada di Institut Teknologi Bandung ini, dan salah satunya adalah Himpunan Mahasiswa Informatika.

Kata kunci: optimasi, *Travelling Salesman Problem*, *Branch and Bound*, pengiriman dan publikasi.

1. PENDAHULUAN

Setiap himpunan mahasiswa yang ada di Institut Teknologi Bandung ini pasti pernah mengadakan acara-acara yang sifatnya umum. Untuk itu, biasanya himpunan akan mengirimkan surat undangan dan juga poster-poster sebagai publikasi kepada himpunan-himpunan lain.

Yang terjadi adalah mahasiswa yang berperan sebagai bagian publikasi seringkali harus berlelah-lelah mengelilingi kampus untuk berjalan ke setiap himpunan. Hingga saat ini bisa dibayangkan bahwa rute perjalanan itu masih tidak optimal. Oleh karena itu, kali ini makalah ini

akan mencoba membahas bagaimana pemilihan rute agar perjalanan menjadi optimal dan tidak terlalu melelahkan.

Masalah pemilihan rute ini termasuk salah satu contoh dari permasalahan *Travelling Salesman Problem*. Yang dibahas di sini adalah minimasi perjalanan dengan menggunakan Algoritma *Branch And Bound*.

2. ALGORITMA *BRANCH & BOUND*

Algoritma *Branch and Bound* atau sering disingkat menjadi algoritma B&B adalah salah satu cara atau metode untuk pencarian di dalam ruang solusi secara sistematis. Dalam hal ini ruang solusi akan diorganisasikan ke dalam pohon ruang status. Pembentukan pohon ruang status pada algoritma ini dibangun dengan skema pencarian melebar atau *Breadth First Search* yang disingkat menjadi BFS.

Dalam skema BFS, misal kita memiliki sebuah graf G yang memiliki n buah simpul dan kita ingin melakukan proses traversal dalam graf dimulai dari salah satu simpul yang ada dalam G , misalnya simpul v , maka BFS akan mengunjungi simpul v baru kemudian mengunjungi terlebih dahulu simpul-simpul yang berdekatan dengan simpul v . Selanjutnya, simpul yang belum dikunjungi dan bertetangga dengan simpul yang tadi dikunjungi dan demikian seterusnya hingga proses traversal selesai.

Skema BFS biasa jika digunakan akan sangat membutuhkan waktu yang lama. Untuk mempercepat pencarian ke simpul solusi, maka setiap simpul akan diberikan sebuah nilai ongkos (*cost*). Dari nilai ongkos yang didapat, simpul yang sudah dibangkitkan oleh DFS tidak seluruhnya akan diekspansi lagi, tetapi yang akan diekspansi adalah simpul yang memiliki nilai ongkos paling kecil. Nilai ongkos pada tiap simpul diberikan berdasarkan perkiraan atau taksiran ongkos paling kecil dari simpul tersebut ke simpul yang dituju (*goal node*). Dengan kata lain nilai ongkos simpul i (dilambangkan dengan $\hat{c}(i)$) merupakan batas bawah (*lower bound*) dari ongkos pencarian solusi dari status i .

Untuk menghitung ongkos ini digunakan sebuah fungsi pembatas. Fungsi pembatas ini berguna untuk membatasi pembangkitan simpul yang tidak mengarah kepada simpul solusi.

2.1 Prinsip Pencarian Solusi dengan B&B

Sebagai mana dijelaskan sebelumnya bahwa algoritma memakai skema BFS untuk membangun pohon ruang solusinya. Maka algoritma B&B juga menggunakan prinsip antrian FIFO (*First In First Out*) dimana setelah simpul hidup dimasukkan ke dalam antrian, simpul berikutnya yang akan menjadi simpul-E adalah simpul yang pertama masuk ke dalam antrian.

Secara umum algoritma B&B adalah sebagai berikut:

1. Masukkan simpul akar ke dalam antrian Q. Jika simpul akar adalah simpul solusi, maka solusi telah ditemukan dan pencarian dihentikan.
2. Jika Q kosong, tidak ada solusi. Pencarian dihentikan
3. Jika Q tidak kosong, pilih dari antrian Q simpul I yang mempunyai $\hat{c}(i)$ paling kecil. Jika terdapat beberapa simpul i yang memenuhi, pilih salah satu secara sembarang.
4. Jika simpul i adalah simpul solusi, artinya solusi ditemukan dan pencarian dihentikan. Jika bukan, bangkitkan anak-anaknya. Jika simpul tidak memiliki anak maka ulang lagi langkah 2.
5. Untuk setiap anak j dari simpul i hitung $\hat{c}(j)$ dan masukkan semua anaknya ke dalam antrian Q.
6. Kembali kelangkah 2 hingga solusi ditemukan.

2.2 Travelling Salesman Person

Dalam upa-bab ini penyusun akan mencoba membahas penyelesaian masalah *Travelling Salesman Problem* atau sering disingkat menjadi TSP dengan menggunakan menggunakan algoritma *Branch & Bound*.

Sebagai contoh, misalkan diberikan sebuah graf $G=(V,E)$ yang merupakan graf lengkap yang merepresentasikan rute-rute perjalanan yang mungkin akan ditempuh. $|V|$ adalah jumlah simpul dalam G yang masing-masing simpulnya diberi nomor secara berurutan dan identik 1, 2, 3, ..., n. Jarak yang harus ditempuh dari sebuah simpul i ke simpul lainnya misalnya j adalah c_{ij} . Perjalanan akan berawal dari sebuah simpul dan berakhir di simpul itu juga. Maka S adalah ruang solusi, yang dalam hal ini $S=\{(1, \pi, 1)\}$ dimana π adalah permutasi dari simpul-simpul yang ada. Dan banyaknya solusi yang bisa terbentuk dinyatakan dengan $|S|$ adalah sebanyak $(|V|-1)!$ Solusi.

Solusi TSP dinyatakan sebagai vektor $X = (1, x_1, x_2, \dots, x_{n-1}, 1)$. Dengan anggapan bahwa $x_0 = x_n = 1$, karena perjalanan berawal dan berakhir di simpul yang sama.

Selanjutnya untuk pemberian nilai ongkos akan dilakukan dengan menggunakan matriks tereduksi dari matriks jarak antar simpul yang dibentuk dari graf G. Yang dimaksud dengan matriks tereduksi di sini 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 di atas kemudian dijumlahkan. Hasil dari penjumlahan ini kemudian dijadikan sebagai $\hat{c}(root)$ atau nilai ongkos dari simpul awal atau akar. Hal ini juga berarti bahwa untuk persoalan TSP tersebut paling tidak memiliki bobot minimum sebesar nilai $\hat{c}(root)$ yang tadi didapat.

Selanjutnya misalkan A adalah matriks tereduksi untuk simpul R dan misal S adalah anak dari simpul R sedemikian sehingga sisi (R, S) pada pohon ruang status melambangkan sisi (i, j) pada graf G. Jika S bukanlah simpul daun, maka untuk mendapatkan matriks jarak yang tereduksi untuk simpul S adalah dengan cara sebagai berikut:

1. Jadikan semua elemen matriks pada kolom j dan baris I menjadi bernilai ∞ .
2. Lalu ubah elemen matriks di baris j kolom pertama menjadi bernilai ∞ .
3. Setelah itu reduksi kembali matriks dengan cara yang telah dijelaskan sebelumnya.

Jika total dari semua angka pengurang yang dipakai dalam mereduksi matriks diatas disimbolkan sebagai r, maka fungsi pembatas yang juga merupakan nilai ongkos untuk simpul S adalah:

$$\hat{c}(S) = \hat{c}(R) + A(i, j) + r \quad (1)$$

Keterangan :

- $\hat{c}(S)$: adalah ongkos perjalanan minimum yang melalui simpul S
- $\hat{c}(R)$: adalah ongkos perjalanan minimum yang melalui simpul R, dimana R adalah simpul bapak (*parent*) dari S.
- $A(i, j)$: adalah nilai pada baris i dan kolom j pada matriks jarak graf G yang berkoresponden dengan sisi (R, S) pada pohon ruang status.
- r : jumlah angka pengurang pada proses pereduksian matriks untuk simpul S.

Hasil dari reduksi matriks diatas kemudian kita sebut sebagai matriks B.

3. OPTIMALISASI PENGIRIMAN

Sebagaimana judul makalah ini maka bab ini akan membahas tentang cara optimalisasi pengiriman. Untuk itu penyusun memberikan gambaran ringkas peta antar himpunan-himpunan di kampus ITB.



Gambar 1. Peta himpunan-himpunan di kampus ITB, warna kuning menyatakan letak himpunan-himpunannya.

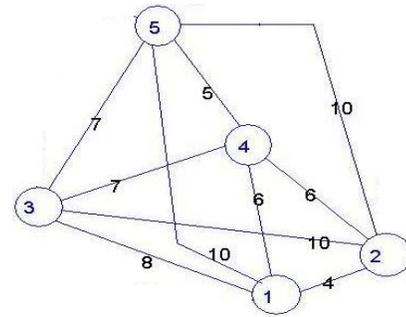
3.1 Penentuan Himpunan sebagai objek TSP

Karena keterbatasan ruang dan waktu yang dimiliki oleh penyusun maka tidak semua himpunan akan diterapkan dalam TSP ini. Penyusun hanya akan mengambil 5 himpunan sebagai sampel kecil dari permasalahan ini.

Adapun kelima himpunan tersebut adalah Himpunan Mahasiswa Informatika (HMIF), Himpunan Mahasiswa Fisika Teknik (HMFT), Himpunan Mahasiswa Teknik Kimia (HIMATEK), Mahasiswa Teknik Industri (MTI), dan Himpunan Mahasiswa Mesin (HMM).

Pemilihan kelima himpunan tersebut dilakukan berdasarkan pembagian region karena himpunan-himpunan tersebut berada pada region yang sama yaitu region Barat Belakang.

Dari peta di atas kita bisa membuat graf yang menghubungkan kelima himpunan mahasiswa tersebut beserta jarak antar simpulnya. Maka graf yang akan terbentuk adalah sebagai berikut:



Gambar 2. Graf tak-berarah 5 himpunan beserta jarak tiap himpunannya.

Keterangan graf:

- simpul 1 : HMIF
- simpul 2 : HMFT
- simpul 3 : HMM
- simpul 4 : HIMATEK
- simpul 5 : MTI

Dari graf tersebut kita bisa membuat matriks jarak antar himpunan yang merepresentasikan jarak antar himpunannya.

$$\begin{bmatrix} \infty & 4 & 8 & 6 & 10 \\ 4 & \infty & 10 & 6 & 10 \\ 8 & 10 & \infty & 7 & 7 \\ 6 & 6 & 7 & \infty & 5 \\ 10 & 10 & 7 & 5 & \infty \end{bmatrix}$$

3.2 Pencarian Nilai Ongkos

Seperti dijelaskan di atas maka matriks jarak tersebut akan direduksi seperti dibawah ini

$$\begin{bmatrix} \infty & 4 & 8 & 6 & 10 \\ 4 & \infty & 10 & 6 & 10 \\ 8 & 10 & \infty & 7 & 7 \\ 6 & 6 & 7 & \infty & 5 \\ 10 & 10 & 7 & 5 & \infty \end{bmatrix} \begin{array}{l} R1 - 4 \\ R2 - 4 \\ R3 - 7 \\ R4 - 5 \\ R5 - 5 \end{array} \begin{bmatrix} \infty & 0 & 4 & 2 & 6 \\ 0 & \infty & 6 & 2 & 6 \\ 1 & 3 & \infty & 0 & 0 \\ 1 & 1 & 2 & \infty & 0 \\ 5 & 5 & 2 & 0 & \infty \end{bmatrix}$$

$$\begin{bmatrix} \infty & 0 & 4 & 2 & 6 \\ 0 & \infty & 6 & 2 & 6 \\ 1 & 3 & \infty & 0 & 0 \\ 1 & 1 & 2 & \infty & 0 \\ 5 & 5 & 2 & 0 & \infty \end{bmatrix} C3 - 2 \begin{bmatrix} \infty & 0 & 2 & 1 & 6 \\ 0 & \infty & 4 & 1 & 6 \\ 2 & 4 & \infty & 0 & 1 \\ 1 & 1 & 0 & \infty & 0 \\ 8 & 8 & 0 & 2 & \infty \end{bmatrix}$$

Matriks terakhir yang kita dapat kita tulis sebagai matriks A. Angka-angka yang dipakai mereduksi matriks tersebut dijumlah dan kita mendapatkan $\hat{c}(\text{root})$ yang juga merupakan nilai $\hat{c}(i)$.

$$\hat{c}(\text{root}) = 4 + 4 + 7 + 5 + 5 + 2 = 27$$

Selanjutnya kita akan ekspansi simpul akar dan mendapatkan simpul anak-anaknya. Dengan catatan bahwa penomoran simpul pada pohon ruang status berbeda dengan penomoran pada graf, maka kita dapat simpul anaknya adalah simpul 2, 3, 4, dan 5. Kemudian dari setiap simpul anaknya akan kita cari nilai *cost* masing-masingnya.

Perhitungannya dikerjakan berdasarkan aturan pada upa-bab 2.1 sebagai berikut:

1. simpul 2: lintasan 1, 2

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 4 & 1 & 6 \\ 2 & \infty & \infty & 0 & 1 \\ 1 & \infty & 0 & \infty & 0 \\ 8 & \infty & 0 & 2 & \infty \end{bmatrix}$$

$r = 2$, sehingga dari persamaan (1) kita dapat $\hat{c}(2) = 27 + 0 + 2 = 29$

2. simpul 3: lintasan 1, 3

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & 1 & 6 \\ \infty & 4 & \infty & 0 & 1 \\ 1 & 1 & \infty & \infty & 0 \\ 8 & 8 & \infty & 2 & \infty \end{bmatrix}$$

$r = 3$, $\hat{c}(3) = 27 + 2 + 3 = 32$

3. simpul 4: lintasan 1, 4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & 4 & \infty & 6 \\ 2 & 4 & \infty & \infty & 1 \\ \infty & 1 & 0 & \infty & 0 \\ 8 & 8 & 0 & \infty & \infty \end{bmatrix}$$

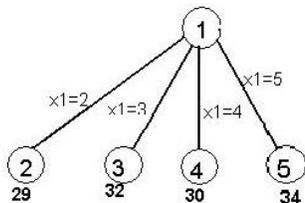
$r = 2$, $\hat{c}(4) = 27 + 1 + 2 = 30$

4. simpul 5: lintasan 1, 5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & 4 & 1 & \infty \\ 2 & 4 & \infty & 0 & \infty \\ 1 & 1 & 0 & \infty & \infty \\ \infty & 8 & 0 & 2 & \infty \end{bmatrix}$$

$r = 1$, $\hat{c}(5) = 27 + 6 + 1 = 34$

Dari nilai-nilai yang kita dapat kita ambil 29 sebagai batasan karena nilai *cost* terkecil dari masing-masing simpul anak adalah 29. Hal ini berarti simpul yang selanjutnya akan diekspansi adalah simpul 2.



Gambar 3. Pohon ruang status hasil ekspansi simpul 1 (root)

Selanjutnya kita akan membangkitkan simpul-simpul anak dari simpul 2. Simpul-simpul yang terbentuk adalah simpul 6, 7, dan simpul 8. Sama seperti sebelumnya kita akan menghitung nilai ongkos untuk masing masing

1. Simpul 6: lintasan 1, 2, 3

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & 1 \\ 0 & \infty & \infty & \infty & 0 \\ 7 & \infty & \infty & 2 & \infty \end{bmatrix}$$

$r=2$, $\hat{c}(6) = 29 + 3 + 2 = 34$

2. Simpul 7: lintasan 1, 2, 4

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 1 & \infty & \infty & \infty & 1 \\ \infty & \infty & 0 & \infty & 0 \\ 7 & \infty & 0 & \infty & \infty \end{bmatrix}$$

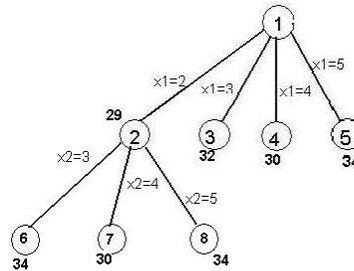
$r=1$, $\hat{c}(7) = 29 + 0 + 1 = 30$

3. Simpul 8: lintasan 1, 2, 5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 1 & \infty & \infty & 0 & \infty \\ 0 & \infty & 0 & \infty & \infty \\ \infty & \infty & 0 & 2 & \infty \end{bmatrix}$$

$r=0$, $\hat{c}(8) = 29 + 5 + 0 = 34$

Pohon ruang status yang sudah terbentuk dari perluasan simpul hingga saat ini di tunjukan oleh gambar 4.



Gambar 4. Pohon ruang status setelah simpul 2 diekspansi

Dari gambar di atas kita bisa langsung memilih simpul 7 sebagai simpul selanjutnya yang akan diekspansi. Alasan pemilihan simpul 7 adalah karena simpul 7 adalah simpul yang memiliki nilai ongkos terkecil dari seluruh simpul hidup saat ini.

Anak anak simpul dari simpul 7 yang terbentuk adalah simpul 9 dan 10 dengan nilai ongkosnya masing-masing adalah sebagai berikut:

1. Simpul 9: lintasan 1, 2, 4, 3

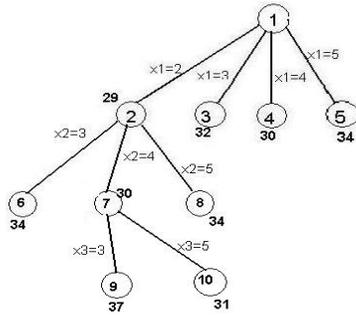
$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 1 \\ \infty & \infty & \infty & \infty & \infty \\ 7 & \infty & \infty & \infty & \infty \end{bmatrix}$$

$r=8$, $\hat{c}(9) = 30 + 0 + 8 = 38$

2. Simpul 10: lintasan 1, 2, 4, 5

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 1 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & \infty & \infty \end{bmatrix}$$

$r=1, \hat{c}(10) = 30 + 0 + 1 = 31$



Gambar 5. Pohon ruang status setelah simpul 7 diekspansi

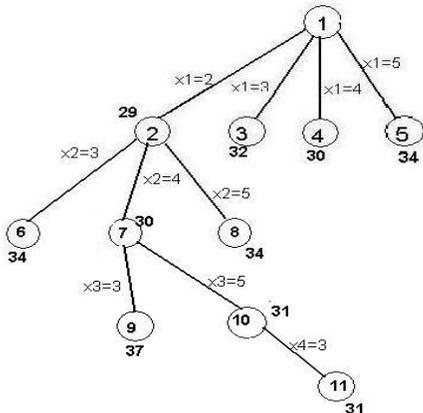
Membandingkan dua nilai ongkos yang ditemukan maka nilai pembatas yang selanjutnya digunakan adalah 30. Akibatnya simpul 10 menjadi simpul selanjutnya yang akan diekspansi dari simpul 10 adalah simpul 11 dengan nilai ongkos sebagai berikut.

Simpul 11: lintasan 1, 2, 4, 5, 3

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

$r=0, \hat{c}(11) = 31 + 0 + 0 = 31$

Hasil perluasan simpul 10 menyebabkan pohon ruang status yang ada sekarang menjadi seperti yang ditunjukkan oleh gambar 6.



Gambar 6. Pohon ruang status yang telah terbentuk saat ini

3.3 Solusi TSP untuk 5 Himpunan

Dari pencarian nilai ongkos di upa-bab sebelumnya dapat kita simpulkan bahwa rute TSP terbaik untuk kelima himpunan jika yang menjadi awal perjalanan yaitu simpul satu dan juga berakhir di simpul yang sama adalah rute dari yang melalui lintasan 1, 2, 4, 5, 3, 1. Dengan kata lain rute optimal yang kita dapat sejauh ini adalah rute HMIF – HMFT – HIMATEK – MTI – HMM – HMIF dengan panjang lintasan 31 satuan perbandingan panjang yang dipakai.

4. KESIMPULAN

1. Algoritma *Branch and Bound* banyak digunakan sebagai metode pencarian solusi. Algoritma ini menggunakan skema *Breadth First Search* sebagai skema traversal dalam grafnya.
2. Salah satu pemecahan untuk *Travelling Salesman Problem* adalah algoritma *Branch and Bound*.
3. Pemecahan solusi TSP pada *Branch and Bound* adalah dengan menggunakan fungsi pembatas yang dirumuskan oleh persamaan (1).
4. Untuk optimalisasi pengiriman antar himpunan dengan Algoritma *Branch and Bound* untuk TSP dalam makalah digunakan 5 himpunan (HMIF, HMFT, HMM, HIMATEK, MTI) sebagai sampel. Rute optimum yang didapat oleh algoritma ini adalah rute HMIF – HMFT – HIMATEK – MTI – HMM – HMIF.

REFERENSI

- [1] Munir, Rinaldi, "Diktat Kuliah Strategi Algoritmik", Bandung, 2007.
- [2] Diestel, Reinhard, "Graph Theory", Springer-Verlag, New York, 2000.
- [2] <http://petakampus.itb.ac.id> diakses 21 Mei 2007, Pukul 16.45.
- [3] <http://www.plano97.net> diakses 21 Mei 2007, Pukul 16.30.