

Penggunaan Algoritma Greedy dalam Penentuan Jalur Angkot di Bandung

Shirley - 13508094

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

1f18094@itb.ac.id

Abstract—Saat ini, di kota metropolitan khususnya, kebanyakan kota telah menyediakan sarana transportasi bagi umum. Kita sebagai pengguna tentu saja selektif dalam memilih jalur yang pendek dan efisien, tidak berliku-liku, menghemat waktu, dan menghemat ongkos tentunya. Algoritma Greedy ini merupakan algoritma pencarian yang dapat digunakan untuk mencari solusi yang dapat mendekati permasalahan tersebut, apalagi dengan banyaknya rute angkot yang berbeda-beda setiap jurusan angkot. Akan banyak sekali alternatif jalan dari tempat asal (initial state) menuju tempat tujuan (goal state). Ditambah dengan relatifnya ongkos pergi yang ditawarkan masing-masing-masing angkot perjurusannya. Algoritma greedy yang akan digunakan juga menggunakan metode heuristik sehingga dalam pengerjaannya, metode ini menggunakan suatu fungsi yang dapat menghitung biaya perkiraan dari suatu simpul tertentu menuju simpul yang lain. Dalam bahasan ini yang dijadikan variabel-variabel acuan diantaranya adalah jarak antar kota $g(n)$ dan estimasi biaya $h(n)$. Oleh karena itu dalam proses pengerjaannya, analisis secara umum akan diterapkan mulai dari skema umum algoritma, ilustrasi gambar dan mapping rute jalan kota yang akan dibahas, tepatnya yaitu masing-masing daerah yang mewakili kota Bandung Timur sampai wilayah Utara tepatnya dari Ujungberung sampai ITB.

Index Terms—Greedy, heuristic, lintasan terpendek, transportasi, angkot.

I. PENGENALAN

Bandung merupakan sebuah kota dan sekaligus ibukota dari provinsi Jawa Barat, Indonesia. Kota ini merupakan kota terbesar ketiga di Indonesia setelah Jakarta dan Surabaya. Selain itu, kota ini juga dikenal dengan sebutan kota kembang dan *Paris Van Java*.

Sampai pada tahun 2004, kondisi transportasi jalan di kota Bandung masih buruk dengan tingginya tingkat kemacetan serta ruas jalan yang tidak memadai, termasuk masalah parkir dan tingginya polusi udara. Permasalahan ini muncul karena beberapa faktor diantaranya pengelolaan transportasi oleh pemerintah setempat yang tidak maksimal seperti rendahnya

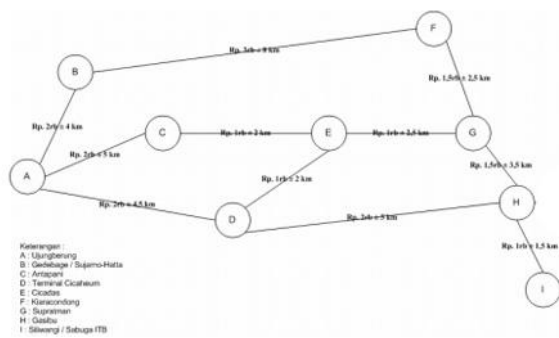
koordinasi antara instansi yang terkait, ketidakjelasan wewenang setiap instansi, dan kurangnya sumber daya manusia, serta ditambah tidak lengkapnya peraturan pendukung.

Sampai tahun 2000 panjang jalan di kota Bandung secara keseluruhan baru mencapai 4.9 % dari total luas wilayahnya dengan posisi idealnya mesti berada pada kisaran 15-20 %^[1]. Pembangunan jalan baru, peningkatan kapasitas jalan dan penataan kawasan mesti menjadi perhatian bagi pemerintah kota untuk menjadikan kota ini menjadi kota terkemuka. Pada 25 Juni 2005, jembatan Pasupati resmi dibuka^[2], untuk mengurangi kemacetan di pusat kota, dan menjadi *landmark* baru bagi kota ini. Jembatan dengan panjangnya 2.8 km ini dibangun pada kawasan lembah serta melintasi Ci Kapundung dan dapat menghubungkan poros barat ke timur di wilayah utara kota Bandung.

Untuk transportasi di dalam kota, masyarakat Bandung biasanya menggunakan angkutan kota atau yang lebih akrab disebut angkot. Selain itu, bus kota dan taksi juga menjadi alat transportasi di kota ini. Sedangkan sebagai terminal busantarkota dan provinsi di kota ini adalah terminal Leuwipanjang untuk rute barat dan terminal Cicaheum untuk rute timur.

Untuk ukuran sebuah kota besar, pengaturan jalan-jalan di kota Bandung cukup rumit, ditambah banyaknya jalan satu arah, sehingga banyak orang luar Bandung yang bertandang ke Bandung mengeluh karena sulitnya mencari jalan di kota ini. Hal itu ditambah buruk dengan angkot yang jumlah rutenya banyak dan warnanya mirip-mirip dan ongkosnya tergantung dari seberapa jauh kita naik angkot tersebut. Permasalahannya adalah bagaimana caranya kita mengetahui angkot mana yang harus kita naiki untuk mencari tujuan dengan ongkos minimal dan jarak minimal pula.

Untuk lebih jelasnya, marilah kita melihat contoh kasus di bawah ini :



Gambar 1 Rute jalan dari Ujungberung ke ITB

Keterangan gambar :

- A : Ujungberung
- B : Gedebage / Soekarno-Hatta
- C : Antapani
- D : Terminal Cicaheum
- E : Cicadas
- F : Kiancondong
- G : Supratman
- H : Gasibu
- I : Siliwangi / Sabuga ITB

II. PEMBAHASAN

Pencarian merupakan kegiatan mendefinisikan ruang masalah untuk masalah yang dihadapi. Ruang masalah ini dapat digambarkan sebagai himpunan keadaan (state) atau bisa juga sebagai himpunan rute dari keadaan awal (initial state) menuju keadaan tujuan (goal state). Langkah kedua adalah mendefinisikan aturan produksi yang digunakan untuk mengubah suatu state ke state lainnya. Langkah terakhir adalah memilih metode pencarian yang tepat sehingga dapat menemukan solusi terbaik dengan usaha yang minimal. Pada pembahasan ini, kita akan menggunakan algoritma greedy sebagai cara pencarian rute.

Algoritma greedy merupakan salah satu algoritma paling populer yang banyak digunakan untuk memecahkan persoalan optimasi. Algoritma ini sederhana dan lempang. Secara harfiah, greedy berarti tamak, sehingga algoritma ini bisa juga diartikan sebagai algoritma yang tamak. Algoritma ini akan memilih jalan yang paling menguntungkan tanpa memikirkan bagaimana ke depannya. Prinsipnya adalah "wake what you can get now!"^[3]. Ada dua metode Greedy yang bisa dipakai, yaitu :

1. Blind atau un-informed search (pencarian buta atau tidak berbekal informasi)
2. Heuristic atau informed search (pencarian dengan berbekal informasi).

Metode yang akan diambil dalam bahasan kali ini adalah greedy menggunakan heuristic karena dalam bahasan kali ini kita akan mendapatkan panduan lain

yang bisa diperkirakan yaitu biaya estimasi tarif angkot.

Kata Heuristic berasal dari sebuah kata kerja Yunani, heuriskein, yang berarti 'mencari' atau 'menemukan'. Dalam dunia pemrograman, sebagian orang menggunakan kata heuristic sebagai lawan kata dari algoritmik, dimana kata heuristic ini diartikan sebagai 'suatu proses yang mungkin dapat menyelesaikan suatu masalah tetapi tidak ada jaminan bahwa solusi yang dicari selalu dapat ditemukan'. Di dalam mempelajari metode-metode pencarian ini, kata heuristic diartikan sebagai suatu fungsi yang memberikan suatu nilai berupa biaya perkiraan (estimasi) dari suatu solusi.

III. METODOLOGI

Pada pembahasannya, algoritma greedy akan membangkitkan simpul berikutnya dari sebuah simpul (yang sejauh ini terbaik diantara semua leaf nodes) yang pernah dibangkitkan. Penentuan simpul terbaik dapat dilakukan dengan menggunakan informasi berupa biaya perkiraan dari suatu simpul menuju ke goal atau gabungan antara biaya sebenarnya dan biaya perkiraan tersebut. Biaya perkiraan tersebut dapat diperoleh dengan menggunakan suatu fungsi yang disebut fungsi heuristic.

Dalam bahasan jalur angkot ini, penulis akan menggunakan fungsi yaitu $f(n)$ dengan biaya perkiraan adalah $h(n)$ sebagai tarif angkot dari satu simpul ke simpul lainnyadan biaya sebenarnya adalah jarak dari stu simpul ke simpul lainnya $g(n)$ seperti yang telah dituliskan dalam (gambar 1.).

Secara umum berikut adalah pseudocode algoritma Greedy^[4] :

1. OPEN berisi initial state dan CLOSED masih kosong.
2. Ulang sampai goal ditemukan atau sampai tidak ada nodes di dalam OPEN
 - a. Ambil simpul terbaik yang ada di OPEN
 - b. Jika simpul tersebut sama dengan goal, maka sukses
 - c. Jika tidak, masukkan simpul tersebut ke dalam CLOSED
 - d. Bangkitkan semua suksesor dari simpul tersebut
 - e. Untuk setiap suksesor kerjakan :
 - i. Jika suksesor tersebut belum pernah dibangkitkan, evaluasi sukseso tersebut, tambahkan ke OPEN, dan catat parent atau orang tuanya.
 - ii. Jika suksesor tersebut sudah pernah dibangkitkan, ubah parentnya jika jalur melalui parent ini lebih baik daripada jalur melalui parent yang sebelumnya. Selanjutnya, perbarui biaya untuk suksesor tersebut dan nodes lain yang

berada di level bawahnya.

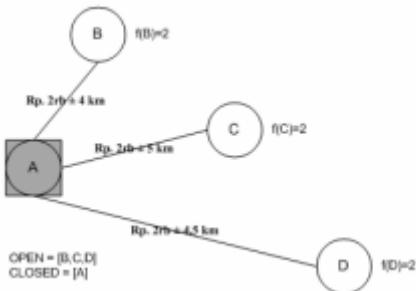
Pada algoritma tersebut diatas, OPEN adalah senarai (list) yang digunakan untuk menyimpan simpul-simpul yang pernah dibangkitkan dan nilai heuristiknya telah dihitung tapi belum terpilih sebagai simpul terbaik (best node). Dengan kata lain, OPEN berisi simpul-simpul yang masih memiliki peluang (peluangnya masih terbuka) untuk terpilih sebagai simpul terbaik. Sedangkan CLOSED adalah senarai untuk menyimpan simpul-simpul yang sudah pernah dibangkitkan dan sudah pernah terpilih sebagai simpul terbaik. Artinya, CLOSED berisi simpul-simpul tidak mungkin terpilih sebagai simpul terbaik (peluang untuk terpilih sudah tertutup).

Ada beberapa cara penyelesaian dalam algoritma greedy, tergantung berdasarkan apa kita melihatnya. Pada pembahasan ini, kita akan membahas greedy terhadap harga dan greedy terhadap jarak.

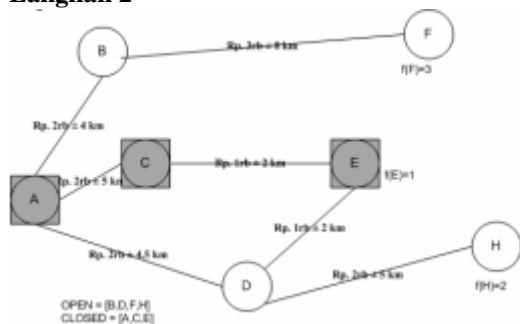
A. Greedy Terhadap Harga

Berikut adalah langkah-langkahnya dalam menyelesaikan masalah jalur angkot yang terdapat pada gambar 1

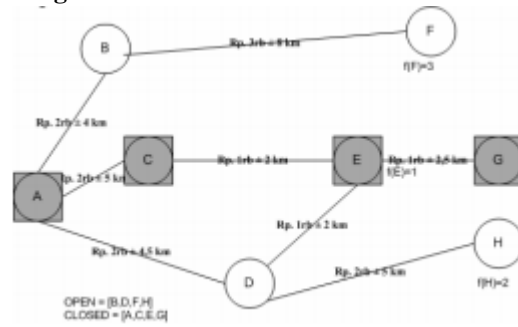
Langkah 1



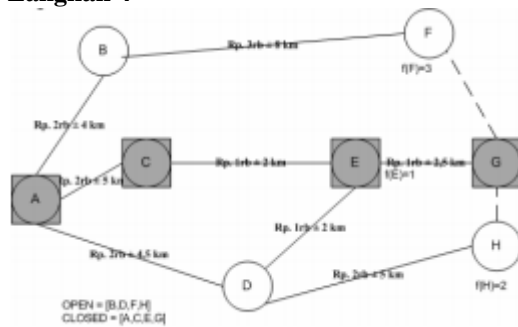
Langkah 2



Langkah 3



Langkah 4



Pada graph di atas simpul hitam merupakan simpul yang telah berada di CLOSED.

Pada langkah ke-1, ada 3 simpul yang bisa dipilih dari simpul awal A, yaitu B, C, dan D. Pada saat menemui persimpangan seperti itu, algoritma greedy akan mengecek simpul manakah yang paling menguntungkan dari kedua simpul tersebut. Ternyata, simpul yang memiliki nilai $h(n)$ yang paling kecil adalah B, C dan D. Dalam kasus ini anggap lah algoritma akan memilih C terlebih dahulu yang akan masuk ke dalam CLOSED.

Pada langkah ke-2, setelah mengambil simpul C, maka kemudian algoritma akan memeriksa, simpul apakah yang ada di C. Ternyata simpul yang terhubung dengan simpul C hanyalah simpul E, maka simpul E pun masuk ke dalam CLOSED.

Pada langkah ke-3, dicek lagi, simpul apakah yang dimiliki oleh simpul terakhir, yaitu simpul E, dan ditemukan bahwa E hanya terhubung dengan simpul G. Kemudian simpul G pun dimasukkan ke dalam CLOSED.

Pada langkah ke-4, sama seperti sebelumnya, dilakukan pengecekan. Ternyata simpul G terhubung ke dua buah simpul, yaitu simpul F dan simpul H. Tetapi karena simpul F tidak mengarah ke solusi, maka simpul H lah yang dipilih dan dimasukkan ke CLOSED. Setelah

simpul H dipilih, maka kemudian diteruskan ke simpul I dan sampailah pada tujuan.

Jadi dengan menggunakan algoritma greedy terhadap harga didapatkan jalur A-C-E-G-H-I. Jalur yang ditempuh adalah Ujungberung → Antapani → Cicadas → Supratman → Gasibu → Siliwangi.

Estimasi biaya dan jarak angkotnya adalah :

$$\begin{aligned} \text{Biaya} &= AC + CE + EG + GH + HI \\ &= 2000 + 1000 + 1000 + 1500 + 1000 \\ &= \text{Rp } 6.500,00 \\ \text{Jarak} &= AC + CE + EG + GH + HI \\ &= 5 + 2 + 2,5 + 3,5 + 1,5 \\ &= 14,5 \text{ km} \end{aligned}$$

Walaupun demikian, sebenarnya ada beberapa rute lain yang bisa ditempuh jika menggunakan algoritma ini karena pada langkah pertama, semua simpul memiliki nilai $h(n)$ yang sama.

- A → B → F → G → H → J / I → K → L
- A → C → E → G → H → J / I → K → L
- A → D → E → G → H → J / I → K → L
- A → D → H → J / I → K → L

B. Greedy Terhadap Jarak

Langkah-langkah pada greedy terhadap jarak ini tidak berbeda jauh dengan greedy terhadap harga.

Pada langkah ke-1, ada 3 simpul yang bisa dipilih dari simpul awal A, yaitu B, C, dan D. Pada saat menemui persimpangan seperti itu, algoritma greedy akan mengecek simpul manakah yang paling menguntungkan dari kedua simpul tersebut. Ternyata, simpul yang memiliki nilai $g(n)$ yang paling kecil adalah B, maka B terlebih dahulu yang akan masuk ke dalam CLOSED.

Pada langkah ke-2, setelah mengambil simpul B, maka kemudian algoritma akan memeriksa, simpul apakah yang ada di B. Ternyata simpul yang terhubung dengan simpul B hanyalah simpul F, maka simpul F pun masuk ke dalam CLOSED.

Pada langkah ke-3, dicek lagi, simpul apakah yang dimiliki oleh simpul terakhir, yaitu simpul F, dan ditemukan bahwa F hanya terhubung dengan simpul G. Kemudian simpul G pun dimasukkan ke dalam CLOSED.

Pada langkah ke-4, sama seperti sebelumnya, dilakukan pengecekan. Ternyata simpul G terhubung ke simpul H, maka simpul H lah yang dipilih dan dimasukkan ke CLOSED. Setelah simpul H dipilih, maka kemudian diteruskan ke simpul I dan sampailah pada tujuan.

Jadi dengan menggunakan algoritma greedy terhadap harga didapatkan jalur A-B-F-G-H-I. Jalur yang ditempuh adalah Ujungberung → Gedebage / Soekarno-Hatta → Kiaracondong → Supratman → Gasibu → Siliwangi.

Estimasi biaya dan jarak angkotnya adalah :

$$\begin{aligned} \text{Biaya} &= AB + BF + FG + GH + HI \\ &= 2000 + 3000 + 1500 + 1500 + 1000 \\ &= \text{Rp } 9.000,00 \\ \text{Jarak} &= AB + BF + FG + GH + HI \\ &= 4 + 8 + 2,5 + 3,5 + 1,5 \\ &= 19,5 \text{ km} \end{aligned}$$

IV. KESIMPULAN DAN SARAN

Untuk mengukur performansi metode pencarian, terdapat empat kriteria yang dapat digunakan, yaitu :

- **Completeness** : Apakah metode tersebut menjamin penemuan solusi jika solusinya memang ada?
- **Time Complexity** : Berapa lama waktu yang diperlukan?
- **Space complexity** : Berapa banyak memori yang diperlukan?
- **Optimality** : Apakah metode tersebut menjamin menemukan solusi yang terbaik jika terdapat beberapa solusi berbeda?

Berdasarkan completeness, algoritma greedy selalu menjamin adanya solusi untuk permasalahannya. Oleh sebab itu, baik algoritma greedy terhadap jarak maupun greedy terhadap harga, masing-masing pasti menjamin penemuan solusinya jika solusinya memang ada.

Untuk ukuran time complexity dan space complexity, waktu dan memori yang diperlukan untuk menganalisisnya hampir sama, tergantung dari rute yang ditempuh. Semakin panjang rute yang ditempuhnya, maka akan semakin lama waktu yang diperlukan untuk menjalankan algoritmanya. Untuk contoh kasus di atas, karena jumlah simpul solusinya sama, maka dapat dipastikan bahwa waktu dan memori yang dibutuhkan kedua algoritmanya juga sama.

Untuk pengukuran optimaliti, algoritma greedy sendiri tidak menjamin hasil yang optimal dari solusi yang diberikan. Untuk contoh kasus ini, greedy terhadap jarak tidak memberikan waktu yang optimal, sedangkan greedy terhadap harga juga tidak mendapatkan hasil yang optimal. Sebenarnya pada greedy terhadap harga, solusi bisa saja didapatkan yang optimal, tapi itu memiliki kemungkinan hanya 20% (1 kemungkinan dari 5 kemungkinan yang ada).

Sebagai saran, untuk ke depannya, jika ingin menghitung jalur optimal seperti ini, sebaiknya ditambahkan nilai heuristik lainnya untuk bahan perhitungan. Misalnya pada kasus ini, bisa ditambahkan nilai heuristik kepadatan jalan misalnya, sehingga nilai optimal yang didapat tidak hanya optimal dalam hal harga, tetapi juga optimal dalam hal waktu tempuh. Dengan menghindari jalan-jalan yang padat dan macet, walaupun mengambil jalan lebih memutar tetapi pengguna bisa sampai di tujuan jauh lebih cepat. Kiranya hal itu juga akan berguna dibandingkan kita hanya memikirkan tentang harga murah saja.

Selain itu, untuk greedy terhadap harga, mungkin bisa ditambahkan algoritma lainnya yang memungkinkan ditemukannya solusi optimal dengan kemungkinan 100%, tidak lagi 20% seperti itu.

REFERENCES

- [1] K. Hermawan, "Attracting tourists, traders, investors: strategi memasarkan daerah di era otonomi," Gramedia Pustaka Utama, 2005.
- [2] <http://diskimrum.jabarprov.go.id>
- [3] R. Munir, "Diktat IF3051 Strategi Algoritma", Program Studi Teknik Informatika ITB, 2009.
- [4] Y. Kurniasari, C. Rahmadhani, F. T. Yuliana, "Penerapan Algoritma Greedy." *Jurusan Teknik Informatika, STT Telkom*

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, 29 April 2010

Shirley - 13508094