

Pencarian Jalur Terpendek untuk Ambulans di Bandung dengan Algoritma Branch and Bound

Muhammad Farhan/13516093

Teknik Informatika

Sekolah Tinggi Elektro dan Informatika

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

Mfarhan0304@gmail.com

Abstrak—Dalam sebuah kondisi darurat, khususnya membutuhkan pertolongan dan pelarian ke rumah sakit, penjemputan oleh ambulans merupakan hal yang akan dilakukan. Ambulans sendiri sudah mendapatkan prioritas pada jalan dan setiap orang sudah mengerti hal tersebut. Tetapi dengan pertumbuhan penduduk dan banyaknya orang yang mempunyai kendaraan bermotor, terutama mobil, kondisi jalanpun menjadi padat dan tidak mudah membuka jalan untuk ambulans. Di lain sisi, korban yang akan ditangani membutuhkan pertolongan yang cepat agar dapat diselamatkan. Dengan menggunakan pendekatan algoritma *branch and bound*, ambulans langsung dapat diarahkan agar menggunakan jalan yang paling dekat dengan tujuan.

Kata Kunci—ambulans; lintasan; *branch and bound*; minimum; A^* ;

I. PENDAHULUAN

Ambulans merupakan sebuah kendaraan yang dilengkapi peralatan medis untuk mengangkut orang sakit atau korban kecelakaan. Kendaraan ini sangat dibutuhkan ketika sebuah kejadian terjadi, seperti tabrakan, kebakaran, bencana alam, dll. Oleh karena itu dalam pelaksanaannya, ambulans diberi sebuah sirine yang kemudian penggunaannya diatur oleh Undang-Undang untuk menandakan agar pengguna jalan lainnya memberi jalan untuk ambulans lewat. Sirine juga dimaksudkan agar orang yang diangkut dapat ditangani oleh medis secara cepat.

Dengan bertumbuhnya penduduk di kota Bandung dan meningkatnya kesejahteraan masyarakat, membuat kuantitas kendaraan bermotor di jalan semakin tinggi, tetapi lebar ruas jalan tidak bertambah. Hal ini membuat keadaan lalu lintas menjadi lebih padat dan berkurangnya ruang untuk ambulans melewati kemacetan.

Permasalahan ini dapat menimbulkan korban yang seharusnya dapat diselamatkan tetapi gagal sebab tertahannya ambulans pada proses perjalanan. Maka dari itu diperlukan perjalanan dengan rute yang seminimum mungkin untuk meminimalisir melewati jalan panjang dan macet. Algoritma *Branch and Bound* digunakan agar memudahkan pengemudi mendapatkan rute terpendek.

Penyelesaian ini tentu dapat menambah jumlah korban yang dapat diselamatkan sebab semakin singkatnya waktu perjalanan yang harus dilalui.

II. TEORIDASAR

A. Branch and Bound

Algoritma *Branch and Bound* merupakan algoritma yang membagi sebuah permasalahan menjadi permasalahan yang lebih kecil untuk diselesaikan dengan tujuan optimalisasi dan memerhatikan batasan-batasan persoalan. Pencarian dilakukan dengan memanfaatkan bentuk struktur pohon pencarian yang kemudian dilakukan pencabangan (*branching*) agar pencarian mengarah pada pencarian solusi dengan memerhatikan penggunaan pembatasan (*bounding*) dalam pencarian solusi optimal. Batasan dikalkulasi dengan menghitung harga yang dibutuhkan dari akar pohon hingga daun. Algoritma ini pertama kali dikemukakan oleh A. H. Land dan A. G. Doig.

Algoritma *Branch and Bound* merupakan perbaikan dari algoritma *Breadth First Search* yang dipadukan dengan teknik *Least Cost Search*. Pada algoritma BFS, simpul berikutnya yang akan diekspansi sesuai dengan urutan pembangkitan simpulnya atau yang biasa disebut dengan *FIFO*, sedangkan pada algoritma *Branch and Bound*, setiap simpul diberi label harga yang merupakan nilai taksiran lintasan ke simpul status tujuan melalui simpul yang sedang dihitung harganya kemudian simpul dengan harga terkecil yang diekspansi.

Fungsi perhitungan harga dinyatakan sebagai berikut :

$$c(i) = f(i) + g(i)$$

$$c(i) = \text{harga untuk simpul ke } i$$

$$f(i) = \text{harga untuk mencapai simpul } i \text{ dari akar}$$

$$g(i) = \text{harga untuk mencapai daun melalui simpul } i$$

Langkah pengerjaan algoritma *Branch and Bound* yaitu :

1. Masukkan simpul akar ke dalam antrian Q. Jika simpul akar adalah solusi, maka solusi ditemukan. Berhenti.
2. Jika Q kosong, tidak ada solusi. Berhenti.
3. Jika Q tidak kosong, pilih dari antrian Q simpul i yang memiliki nilai harga $c(i)$ paling kecil. Jika terdapat beberapa simpul i yang memenuhi, pilih satu sembarang.
4. Jika simpul i adalah simpul solusi, solusi ditemukan. Berhenti. Jika simpul i bukan simpul solusi, maka bangkitkan anak-anaknya. Jika simpul i tidak mempunyai anak, kembali ke langkah 2.
5. Untuk setiap anak j dari simpul i , hitung $c(j)$ dan masukkan semua anak-anak tersebut ke dalam Q.
6. Kembali ke langkah 2.

B. Graf

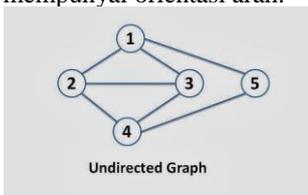
Graf adalah sebuah struktur yang dapat digunakan untuk merepresentasikan hubungan yang terjadi antara suatu objek diskrit yang satu dengan objek diskrit yang lain.

Graf dapat didefinisikan sebagai suatu pasangan himpunan dari himpunan tidak kosong dari simpul- simpul (V/Vertex) dan himpunan sisi yang menghubungkan sepasang simpul (E/Edge), E boleh kosong. Jadi, sebuah graf dimungkinkan tidak mempunyai sisi, tetapi memiliki jumlah titik minimal satu. Graf yang hanya memiliki satu buah titik tanpa sisi dinamakan graf trivial. Penulisan untuk graf G dapat disingkat dengan notasi

$$G = (V,E)$$

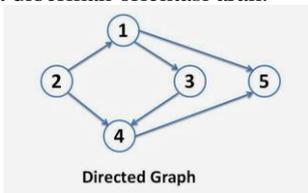
Berdasarkan orientasi arah pada sisi, maka secara umum graf dibedakan atas 2 jenis:

1. **Graf tak-berarah** (undirected graph) Graf yang sisinya tidak mempunyai orientasi arah.



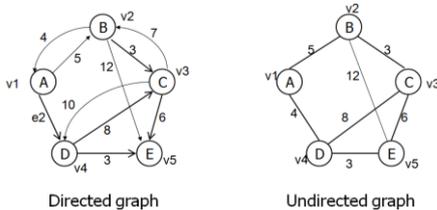
Gambar 1 Graf Tak-Berarah

2. **Graf berarah** (directed graph atau digraph) Graf yang setiap sisinya diberikan orientasi arah.



Gambar 2 Graf Berarah

Seringkali suatu permasalahan dapat dipetakan menjadi graf dengan menggunakan titik serta keterhubungan antar titik. Berikut adalah beberapa contoh dari graf :



Gambar 3 Graf Berbobot

Pada gambar 1,

Directed Graph :

$$V = \{A, B, C, D, E\}$$

$$E = \{(A, B), (A, D), (B, A), (B, C), (B, E), (C, B), (C, D), (C, E), (D, C), (D, E)\}$$

Undirected Graph :

$$V = \{A, B, C, D, E\}$$

$$E = \{(A, B), (A, D), (B, C), (B, E), (C, D), (C, E), (D, E)\}$$

C. A*

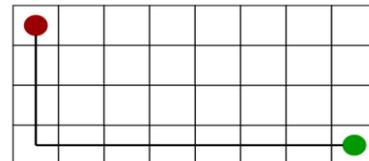
A* atau yang biasa juga disebut dengan algoritma A star adalah salah satu algoritma yang dapat digunakan untuk menentukan total minimum biaya lintasan dan juga saat kondisi yang tepat dapat memberikan solusi yang optimal. Cara kerja dari algoritma ini hampir sama dengan algoritma best first search, akan tetapi di modifikasi dengan fungsi heuristik.

Algoritma A* dapat melakukan prediksi pada setiap simpul-simpul yang dibuat. Langkah ini dilakukan agar memudahkan algoritma tersebut untuk menentukan langkah-langkah selanjutnya yang di harapkan. Cara tersebut dapat disimbolkan dengan fungsi $f'(n)$ sebagai pendekatan terhadap fungsi $f(n)$ dimana fungsi $f(n)$ adalah fungsi evaluasi yang sebenarnya terhadap simpul n .

Pada banyak kasus akan jauh lebih baik jika suatu fungsi di definisikan menjadi suatu kombinasi ataupun jumlah dari dua atau lebih komponen yaitu $g(n)$ dan $h(n)$. Fungsi $g(n)$ adalah ukuran dari biaya yang telah dikeluarkan dari suatu keadaan awal hingga ke simpul n . Hasil dari $g(n)$ tersebut bukan merupakan hasil dari estimasi melainkan jumlah dari biaya penerapan setiap aturan yang dilakukan sepanjang lintasan terbaik yang telah ditentukan dengan fungsi heuristik yang menuju suatu simpul.

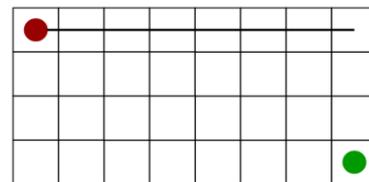
Dalam penentuan fungsi $h'(n)$, terdapat tiga pendekatan umum, yaitu :

1. **Manhattan Distance** – Perhitungan dilakukan dengan cara menghitung jarak sebenarnya.



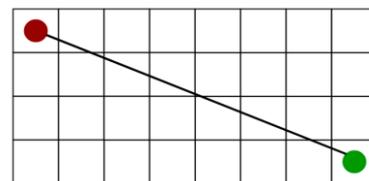
Gambar 4 Manhattan Distance

2. **Diagonal Distance** – Perhitungan jarak hanya menggunakan data pada sumbu x atau jika pada peta, hanya menggunakan garis lintang.



Gambar 5 Diagonal Distance

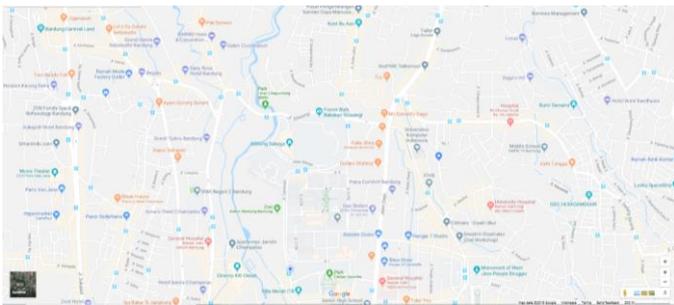
3. **Euclidian Distance** – Perhitungan menggunakan garis lurus dari suatu posisi ke posisi lainnya tanpa memerhatikan proses.



Gambar 6 Euclidian Distance

III. PEMBAHASAN

A. Persoalan

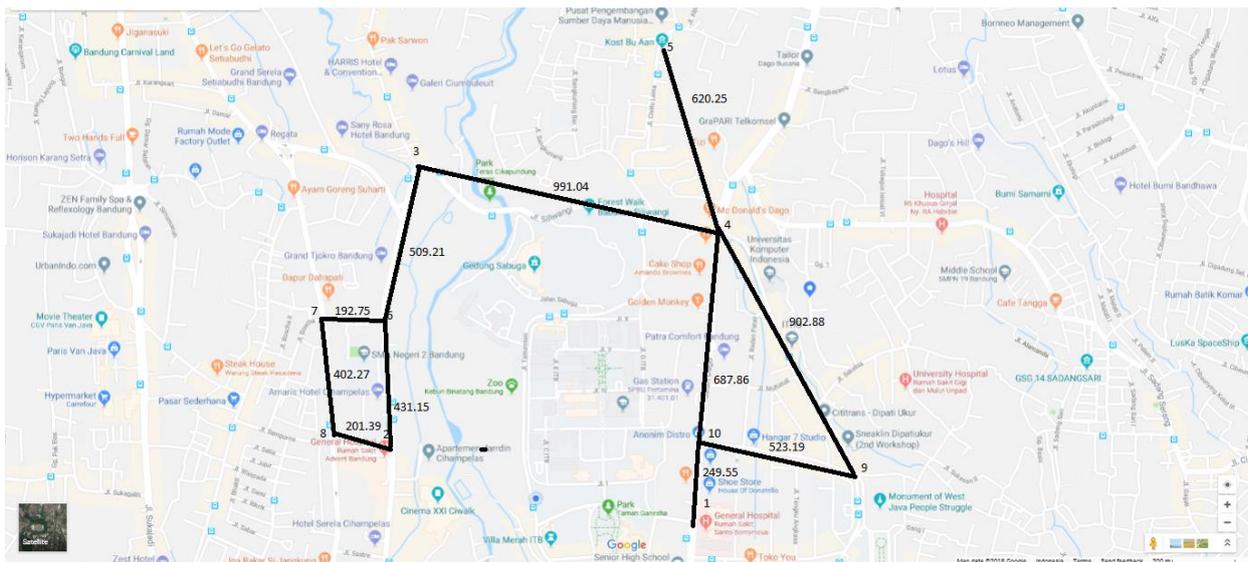


Gambar 7 Peta Wilayah Bandung Utara

RS. Santo Borromeus dan RS. Advent Bandung merupakan dua rumah sakit yang ramai dikunjungi oleh masyarakat dan menyediakan ambulans untuk beberapa keadaan tertentu. Daerah pada peta notabene merupakan tempat padat kendaraan, terutama pada jam berangkat dan pulang kerja. Hal ini dikhawatirkan dapat memperlambat ambulans dalam melakukan proses penjemputan atau pengantaran pasien untuk mendapat pertolongan yang cepat. Terutama jika pengemudi ambulans tidak mengambil jalan yang cepat untuk mencapai tujuan.

Pada makalah kali ini, penulis mencoba untuk menggunakan permintaan ambulans dari Kos Bu Aan dan kembali lagi ke Rumah Sakit yang berkaitan, kemudian menggambarkan lintasan terpendek untuk pengemudi beserta jarak yang harus ditempuh.

B. Penarikan Informasi



Gambar 8 Pelengkapan Informasi Peta

Langkah yang kemudian dilakukan adalah mencari jarak antar titik bertetangga. Hal tersebut dapat dilakukan dengan menggunakan bantuan *google maps* dalam perhitungan jarak antar titik. Perhitungan dilakukan dengan menggunakan garis lurus sebagai jarak dari satu titik ke titik lainnya atau yang biasa disebut dengan *euclidian distance*.

Hasil perhitungan jarak kemudian ditabulasikan. Tabel tersebut kemudian dapat digunakan untuk melakukan komputasi dengan algoritma *Branch and Bound*. Kita dapat melihat dan memilih mana jarak yang terpendek dari suatu titik ke titik lainnya, kemudian menjumlahkan dengan fungsi perhitungan harga sesuai yang telah dituliskan.

2	996.68	1	996.68
3	1457	3	944.83
4	946.05	4	1278
5	1541	5	1581
6	1205	6	455.52
7	1379	7	503.23
8	1192	8	199.1
9	555.55	9	1503
10	251.35	10	989
RS Santo Borromeus		RS Advent Bandung	

Tabel 1 Jarak Kos Bu Aan – Rumah Sakit

C. Komputasi

Persoalan dimulai dari simpul kos Bu Aan, yaitu simpul nomor 5. Persoalan dibagi menjadi dua, yaitu kearah RS Santo Borromeus dan RS Advent Bandung. Pada persoalan kali ini c_1 merupakan harga menuju RS Santo Borromeus sedangkan c_2 merupakan harga menuju RS Advent Bandung.

$$c_1(1) = 1541 \quad c_2(1) = 1581$$

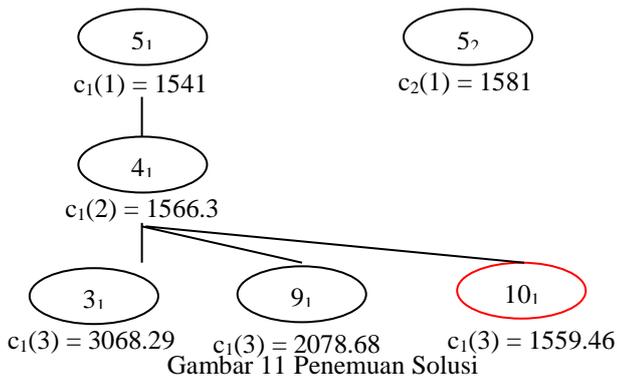
Gambar 9 Akar

Perhitungan harga, $c_1(1) = 1541$ dan $c_2 = 1581$. Hal ini disebabkan kos Bu Aan adalah tempat pertama perjalanan, maka yang kita ekspansi adalah c_1 sebab memiliki nilai yang lebih kecil dibandingkan dengan c_2 .



Gambar 10 Ekspansi Akar

$c_1(2)$ dihitung dengan menjumlahkan 620.25 yang merupakan jarak titik 5 menuju titik 4 dengan jarak titik 4 menuju RS. Santo Borromeus, yaitu titik 1 dengan jarak 946.05. Dikarenakan jumlah $c_1(2)$ lebih kecil dari daun $c_2(1)$, maka $c_1(2)$ dipilih untuk diekspansi.



Gambar 11 Penemuan Solusi

$c_1(2)$ jika diekspansi akan memiliki tiga daun sebab pada peta, terdapat tiga percabangan. Hal ini menimbulkan adanya $c_1(3)$ pertama yang dikalkulasi dengan 620.25 + 991.04 sebagai $g(n)$ dan 1457 sebagai $h'(n)$, $c_1(3)$ kedua yang dikalkulasi dengan 620.25 + 902.88 sebagai $g(n)$ dan 555.55 sebagai $h'(n)$, serta $c_1(3)$ ketiga yang dikalkulasi dengan 620.25 + 687.86 sebagai $g(n)$ dan 251.35 sebagai $h'(n)$.

Setelah dilakukan kalkulasi seperti diatas, didapat bahwa $c_1(3)$ ketiga memiliki nilai yang lebih kecil dari seluruh daun lainnya. Hal ini menyebabkan $c_1(3)$ ketiga dipilih untuk dilanjutkan sebagai ekspansi selanjutnya, ternyata $c_1(3)$ ketiga merupakan simpul tujuan.

Dari seluruh komputasi diatas, dapat disimpulkan bahwa RS. Santo Borromeus merupakan rumah sakit terdekat dengan jalur 5 - 4 - 10 - 1. Algoritma ini kemudian dapat dikembangkan dan membantu seluruh rumah sakit yang ada di Indonesia untuk lebih cepat dalam menggapai pasien yang membutuhkan.

IV. KESIMPULAN

Dalam permasalahan penentuan lintasan penentuan rumah sakit terdekat, dapat digunakan algoritma *Branch and Bound*. Namun, dalam dunia nyata, tidak akan sesimpel itu, dikarenakan terdapat berbagai faktor lain yang harus dipertimbangkan. Terkadang faktor-faktor tersebut begitu dinamis dan tidak bisa semudah itu ditampilkan dalam bentuk angka sehingga akan memberikan tantangan tersendiri untuk mengolahnya menjadi solusi yang dapat diterima dan optimal. Maka dalam kasus yang lebih akurat dan kompleks, solusi ini tidak sepenuhnya tepat.

Algoritma pencarian pencarian jalur ambulans terdekat dilakukan secara bertahap, tahap yang pertama yaitu mencari rumah sakit dengan menggunakan algoritma *Branch and Bound(A*)* diatas. Pertama yaitu dengan menentukan semua kemungkinan rute yang ada dan harus mengetahui jarak setiap titiknya. Kemudian tabulasi data yang diperoleh menjadi data jarak titik kepada tujuannya.

Meski begitu, dengan segala asumsi dan keterbatasan yang ada, dan dengan memberikan heuristik yang sesuai, algoritma *Branch and Bound* akan memberikan hasil yang optimal. Dan seiring dengan peningkatan kemampuan ekstraksi informasi, algoritma ini akan terus berkembang dan semakin luas digunakan terutama dalam rangka memberikan layanan penentuan lintasan dan sistem pengiriman yang lebih baik.

REFERENSI

- [1] <http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2017-2018/stima17-18.htm>, diakses pada 10 Mei 2018.
- [2] <https://www.thecrazyprogrammer.com/2014/01/graphs-introduction-and-terminology.html>, diakses pada 11 Mei 2018.
- [3] <http://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html>, diakses pada 11 Mei 2018.

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, 10 Mei 2018

Muhammad Farhan - 13516093