

Penerapan Algoritma A* sebagai salah satu algoritma Branch & Bound Pada Aplikasi Global Positioning System (GPS)

Humasak Simanjuntak¹, Marojahan Sigiro²

Laboratorium Ilmu dan Rekayasa Komputasi

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

E-mail : humasak@students.if.itb.ac.id¹, marojahan@students.if.itb.ac.id²

Abstrak

Perkembangan teknologi yang sangat cepat harus didukung dengan perkembangan dan implementasi algoritma yang mangkus, sehingga teknologi tersebut dapat memberikan hasil yang baik, efektif dan efisien. Algoritma A* merupakan salah satu algoritma Branch & Bound yang menggunakan informasi tambahan (fungsi heuristik) dalam melakukan pencarian solusi. Algoritma ini akan memberikan solusi yang optimal terhadap suatu status pencarian yang dilakukan. Pada saat ini, algoritma A* atau algoritma Branch & Bound banyak dikembangkan pada aplikasi *path finding* maupun aplikasi-aplikasi game. Salah satu teknologi yang berkembang pada saat ini adalah teknologi Global Positioning System (GPS). Pada makalah ini, akan diusulkan penerapan algoritma A* dengan suatu fungsi heuristik untuk teknologi *Global Positioning System*. Penerapan ini diharapkan memberikan hasil yang lebih optimal dalam penentuan sebuah posisi maupun jalur yang akan dilalui oleh pengguna aplikasi tersebut.

Kata kunci: algoritma A*, algoritma Branch & Bound, Global Positioning System, Fungsi Heuristik.

1. Pendahuluan

Kebutuhan manusia akan sebuah teknologi terus meningkat. Teknologi tersebut diharapkan tidak hanya memberikan solusi yang baik, tetapi juga harus memberikan solusi yang optimal dan efisien.

Pada saat ini, banyak algoritma yang berkembang untuk mencapai solusi dari sebuah permasalahan. Namun, algoritma tersebut belum tentu memberikan solusi yang optimal terhadap masalah yang ada. Solusi yang optimal dapat dihasilkan dengan menambahkan sebuah informasi tambahan pada algoritma tersebut yang akan digunakan oleh algoritma sebagai dasar dalam pencarian solusi. Algoritma yang menggunakan informasi tambahan atau sering disebut fungsi heuristik dalam pencarian solusi optimal adalah algoritma Branch & Bound. Salah satu algoritma Branch & Bound yang banyak digunakan adalah algoritma A*.

Pada makalah ini, akan diterapkan penggunaan algoritma A* pada teknologi GPS. Dimana, teknologi GPS yang dimaksud, tidak hanya memberikan posisi atau letak dari seseorang, tetapi mampu bekerja sebagai sebuah aplikasi *intelligent* yang dapat memberikan jalur teroptimal bagi pengguna jika ditentukan daerah asal dan daerah tujuan yang ingin dicapai. Dalam hal ini, akan diberikan suatu informasi tambahan yang akan digunakan sebagai dasar dalam pencarian solusi tersebut.

2. Algoritma A*

2.1 Pengertian Algoritma A*

Algoritma A* adalah algoritma yang dikemukakan oleh Hart, Nilsson, dan Raphael pada tahun 1968. Algoritma A* merupakan salah satu algoritma Branch & Bound atau disebut juga sebagai sebuah algoritma untuk melakukan pencarian solusi dengan menggunakan informasi tambahan (heuristik) dalam menghasilkan solusi yang optimal.

2.2 Penggunaan Fungsi Heuristik

Pada algoritma A*, untuk mempercepat pencarian simpul solusi, maka setiap simpul diberi sebuah nilai ongkos (*cost*). Dimana, simpul berikutnya yang akan diekspansi tidak lagi berdasarkan urutan pembangkitan (seperti BFS), tetapi simpul yang memiliki ongkos paling kecil (*least cost search*).

Nilai ongkos pada setiap simpul n menyatakan taksiran ongkos termurah lintasan dari simpul n ke simpul solusi (*goal node*), yaitu:

$F(n)$ = nilai taksiran lintasan termurah dari simpul status n ke status tujuan

Dengan kata lain, $F(n)$ menyatakan batas bawah (*lower bound*) dari ongkos pencarian solusi dari status n . Fungsi heuristik yang terdapat pada algoritma A* untuk menghitung taksiran nilai dari suatu simpul dengan simpul yang telah dilalui adalah:

$$F(n) = g(n) + h(n)$$

Dimana :

$F(n)$ = ongkos untuk simpul n

$g(n)$ = ongkos mencapai simpul n dari akar

$h(n)$ = ongkos mencapai simpul tujuan dari simpul n

2.3 Algoritma

Algoritma A* merupakan salah satu algoritma Branch & bound. Oleh karena itu, algoritma yang digunakan pada algoritma Branch & Bound sama dengan algoritma yang digunakan pada algoritma A*. Adapun algoritma yang digunakan pada algoritma A* adalah:

1. Masukkan simpul akar ke dalam antrian Q . Jika simpul akar adalah simpul solusi (*goal node*), maka solusi telah ditemukan. Stop.
2. Jika Q kosong, tidak ada solusi. Stop
3. Jika Q tidak kosong, pilih dari antrian Q simpul n yang mempunyai $F(n)$ paling kecil. Jika terdapat beberapa simpul n yang memenuhi, pilih satu secara sembarang.
4. Jika simpul n adalah simpul solusi, berarti solusi sudah ditemukan, stop. Jika simpul n bukan simpul solusi, maka bangkitkan semua anak-anaknya. Jika n tidak mempunyai anak, kembali ke langkah 2.
5. Untuk setiap anak j dari simpul n , hitung $F(n)$, dan masukkan semua anak-anak tersebut ke dalam antrian Q .
6. Kembali ke langkah 2

3. Global Positioning System (GPS)

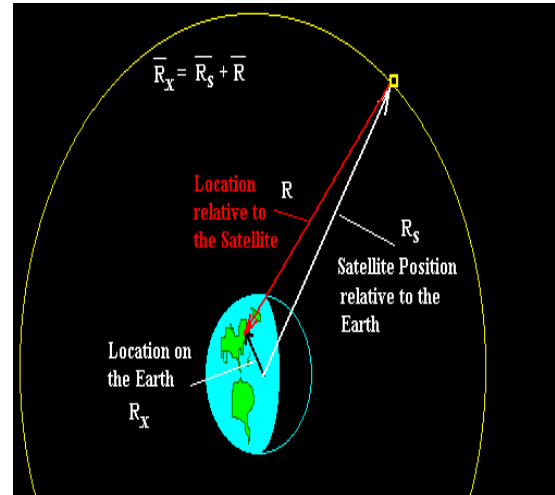
3.1 Pengertian

GPS adalah suatu teknologi *space based* atau sistem navigasi radio yang mencakup seluruh dunia, cocok untuk segala cuaca, posisi tiga dimensi, percepatan, navigasi pelayaran, dan data waktu untuk pengguna baik warganegara biasa maupun militer. Penggunaan potensial dari GPS dalam suatu komunitas maupun lingkungan yang berbeda (seperti: jalan raya) adalah menyediakan informasi perjalanan atau navigasi yang dipetakan dengan *Geographic Information System* (GIS).

3.2 Cara Kerja

GPS dapat menyediakan infrastruktur peta jalan raya digital yang sangat akurat. Teknologi GPS bekerja berdasarkan prinsip segitiga, dimana jika jarak dari *observer* ke tiga titik dapat diukur, maka posisi dari *observer* dapat diketahui. Sistem ini menggunakan paling sedikit 24 satelit dalam orbit 19.320 km (12.000 mil) diatas permukaan bumi. Satelit ini

secara bertahap akan mengirimkan (*broadcast*) posisi (lokasi), waktu dan informasi lain yang diperlukan. Berikut ini adalah konsep kerja dari GPS:



Gambar 1 Perhitungan jarak pada GPS

Dari gambar diatas, dapat diketahui bahwa:

1. Posisi satelit relatif terhadap bumi
2. Lokasi pada bumi dilokasikan relatif terhadap satelit
3. Kemudian, posisi lokasi pada bumi dapat dihitung dari penjumlahan vektor dua besaran yang lain. Semua besaran harus dilakukan pada sebuah presisi dimana lokasi pada bumi diketahui berada pada jarak 15 m
4. Jarak dari satelit dihitung dengan waktu yang diperlukan sebuah signal radio untuk mencapai lokasi dari satelit.

$$\text{distance} = (\text{speed of light}) \times (\text{time of flight})$$

4. Eksperimen

Setelah mengetahui cara kerja dari algoritma A* dan GPS, maka melalui makalah ini akan diusulkan penggunaan algoritma A* dalam menentukan jalur yang paling optimal jika diketahui daerah asal dan daerah tujuan. Eksperimen *path finding* ini dipilih karena GPS merupakan sebuah aplikasi atau sistem yang dapat digunakan untuk pencarian posisi maupun jalur yang akan dilalui ke suatu tempat, sehingga akan ditemukan kesesuaian penggunaan algoritma A* dalam pencarian jalur optimal pada aplikasi GPS. Melalui hasil eksperimen ini juga akan diusulkan kombinasi dari algoritma A* dengan aplikasi GPS yang telah ada saat ini sehingga aplikasi tersebut dapat memberikan hasil yang lebih baik.

4.1 Deskripsi Persoalan

Persoalan yang diambil dalam eksperimen ini adalah metode pencarian jalan (*path finding*) dari Pluit menuju Tebet melalui beberapa tempat di Jakarta. Beberapa tempat di Jakarta yang dilalui adalah Pluit, Kapuk, Ancol, Grogol, Palmerah, Kebayoran Lama, Kebayoran Baru, Gandaria, Kemang, Menteng, Jatinegara, Cempaka Putih, Kemayoran, Tanjung Priok, Palmerah, Tebet. Pengambilan beberapa tempat ini didasarkan pada tempat-tempat umum yang sering menjadi daerah tujuan di Jakarta. Dalam hal ini, setiap tempat akan digambarkan sebagai sebuah simpul yang dinyatakan dengan sebuah persimpangan yang ada pada daerah tersebut. Nilai sisi antara dua simpul menyatakan jarak antara kedua simpul bersebelahan (bertetangga).

Tujuan dari pencarian jalan ini adalah untuk mendapatkan jalur dengan nilai (jarak) terkecil dan teroptimal dari simpul awal (Pluit) menuju simpul tujuan (Tebet) dengan memperhitungkan setiap persimpangan yang ada dalam mencapai simpul tujuan.

4.2 Batasan

Beberapa batasan dalam eksperimen ini adalah:

1. Simpul awal dan simpul akhir diketahui (Pluit menuju ke Tebet)
2. Jarak antar tiap simpul diberikan dan terdapat nilai heuristik pada tiap simpul tersebut. Nilai heuristik setiap simpul dinyatakan dengan banyaknya persimpangan yang akan dilalui dari simpul tersebut dikalikan dengan konstanta yang diperoleh dari hasil analisa.

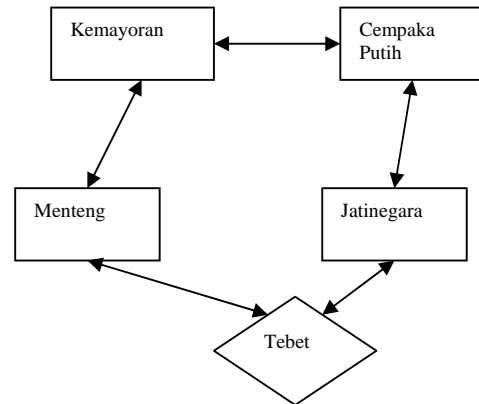
Pada eksperimen ini, diasumsikan bahwa jalan yang akan dilalui adalah jalan yang telah didefinisikan dalam graf. Selain itu, kemacetan yang ada pada setiap jalan tidak diperhitungkan dan setiap keberadaan lampu lalu lintas pada setiap persimpangan dihitung dalam fungsi heuristik.

4.3 Penentuan Nilai heuristik

Pemecahan persoalan *path finding* dengan menggunakan algoritma A* sangat bergantung pada fungsi heuristik. Oleh karena itu, fungsi heuristik perlu ditentukan dengan cermat. Pada persoalan di ini, penentuan nilai heuristik setiap simpul menggunakan pendekatan yang melibatkan perhitungan jumlah perempatan (*traffic light*) dari suatu simpul terhadap simpul tujuan (dalam kasus ini simpul "Tebet"). Simpul tujuan sendiri memiliki nilai heuristik nol.

Untuk mempermudah penentuan nilai heuristik, sebelumnya dilakukan perhitungan jumlah

perempatan antara 2 buah simpul yang dihubungkan oleh sebuah sisi (disebut I). Sebagai contoh, simpul "Kapuk" dengan simpul "Palmerah" memiliki nilai $I=8$ atau simpul "Menteng" dengan "Tebet" memiliki nilai $I=7$. Asumsi yang digunakan adalah jalan beserta perempatan yang dilalui selalu sama. Selanjutnya, lakukan perhitungan nilai heuristik sementara (disebut T). Untuk simpul yang bersisian langsung dengan simpul tujuan, nilai T-nya adalah nilai I antara simpul tersebut dan simpul tujuan. Sedangkan untuk simpul yang berhubungan dengan simpul tujuan melalui simpul atau simpul-simpul lain, misalnya simpul X dan Y. Nilai T-nya adalah minimum dari nilai I_X ditambah dengan nilai T_X dan nilai I_Y ditambah dengan nilai T_Y . I_X adalah jumlah perempatan dari simpul tereksansi dengan simpul X, I_Y adalah jumlah perempatan dari simpul tereksansi dengan simpul Y. Sebagai contoh:



Nilai T_{Tebet} adalah 0, karena "Tebet" merupakan simpul tujuan. Nilai $T_{Menteng}$ adalah jumlah perempatan antara "Menteng" dan "Tebet", yaitu 7. Begitu juga halnya dengan nilai $T_{Jatinegara}$ adalah jumlah perempatan antara "Jatinegara" dan "Tebet", yaitu 6. Nilai $T_{CempakaPutih}$ adalah minimum dari $I_{Jatinegara} + T_{Jatinegara}$ dan $I_{Kemayoran} + T_{Kemayoran}$. Karena nilai $T_{Kemayoran}$ belum diketahui dan nilai $T_{Kemayoran}$ jika dihitung melalui "Menteng" ($T_{Menteng} + I_{Menteng}$) lebih besar daripada $I_{Jatinegara} + T_{Jatinegara}$, maka nilai $T_{CempakaPutih} = I_{Jatinegara} + T_{Jatinegara} = 16$. Selanjutnya, nilai $T_{Kemayoran}$ diperoleh dari minimum nilai $I_{CempakaPutih} + T_{CempakaPutih}$ dan $T_{Menteng} + I_{Menteng}$, yaitu $T_{Menteng} + I_{Menteng} = 17$.

Untuk mendapatkan nilai heuristik, kalikan nilai heuristik sementara dengan suatu konstanta, yaitu 0.833. Konstanta ini diperoleh atas dasar pemikiran sebagai berikut:

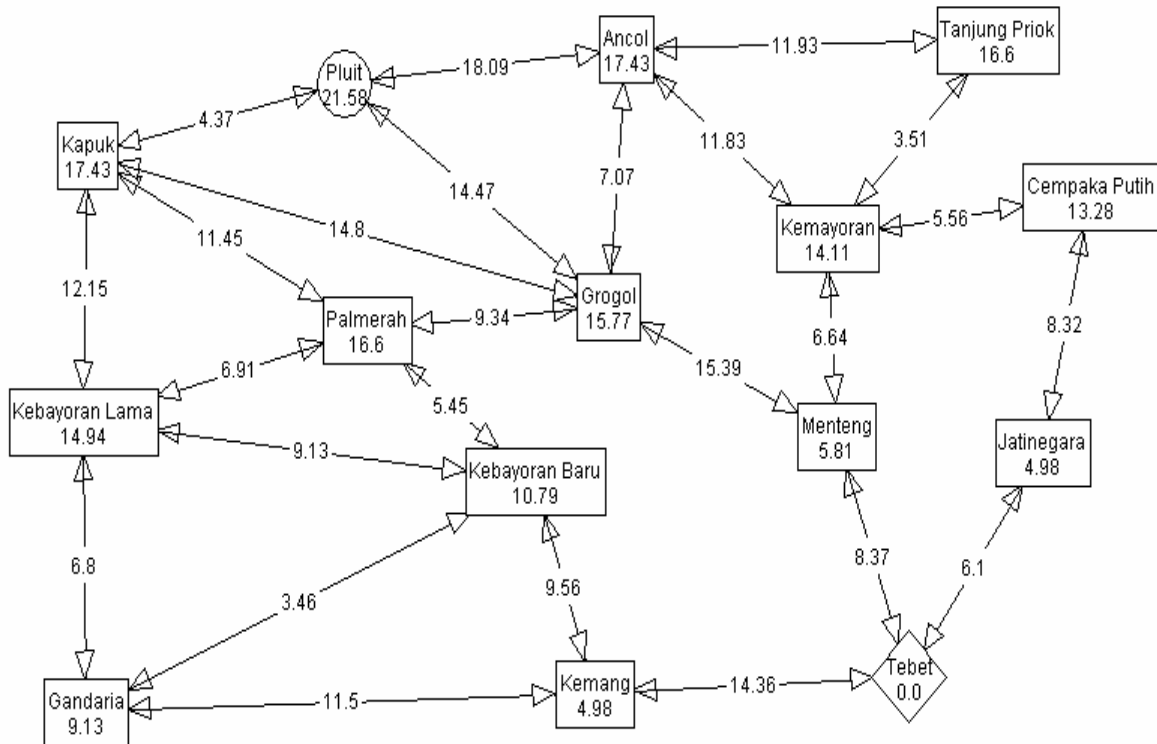
Diasumsikan, lama waktu lampu merah rata-rata di Jakarta adalah satu menit dan kecepatan rata-rata mobil-mobil di Jakarta adalah 50 km/jam. Untuk setiap satu menit yang terbuang, sebuah mobil dengan kecepatan 50 km/jam dapat menempuh jarak sebanyak 0.833 kilometer. Andaikan sebuah mobil, yang melaju di jalan sepanjang X kilometer, terhenti selama satu menit di depan lampu merah. Hal

tersebut dapat disamakan dengan sebuah mobil yang menempuh $X+0.833$ kilometer. Begitu juga halnya dengan kasus mobil yang terhenti di depan n lampu merah. Nilai X disini tidak lain adalah nilai sisi antara dua buah simpul.

Jadi dirumuskan:

$I(x,y)$ = Jumlah perempatan antara 2 buah simpul
 T_n = nilai heuristik sementara untuk simpul n
 $T_y = \text{Min } (I(x,y) + T_x)$, x adalah semua tetangga y
 $T_y = I(x,y)$, bila x = tujuan, dimana $T_x = 0$.
 $T_{\text{final}} = T_y * c$, $c=0.833$.

Dengan perumusan tersebut, tiap simpul yang melambangkan tiap daerah akan memiliki nilai heuristik yang merupakan perkiraan untuk mencapai tujuan. Selain itu, tiap sisi yang terdapat dari satu simpul ke simpul lain akan memiliki nilai/cost. Oleh karena itu, persoalan *path finding* beberapa daerah di Jakarta digambarkan dengan graph sebagai berikut:



Gambar 2 Graph beberapa lokasi di Jakarta dengan nilai heuristik tiap simpul

4.4 Hasil

Pada eksperimen ini, digunakan sebuah aplikasi (*tool*) yang disebut **CIspace applet** yang dapat mengimplementasikan algoritma A*. Oleh karena itu, data yang diperlukan oleh *tool* ini dalam pencapaian sebuah solusi adalah graph dari daerah yang akan ditelusuri beserta dengan nilai heuristik untuk tiap simpul yang telah dihitung dengan

menggunakan fungsi heuristik yang telah ditentukan sebelumnya serta jarak antar simpul.

Jalur optimal yang dihasilkan oleh algoritma A* melalui aplikasi **CIspace applet** dengan titik awal = Pluit dan titik akhir = Tebet adalah :

Pluit --> Grogol --> Menteng --> Tebet

dengan total *cost* : $51.46 = 42.33 + 9.13$. Jalur ini dihasilkan dengan beberapa urutan :

Path:Pluit-->Kapuk-->KebayoranLama-->

Kebayoran Baru --> Gandaria

Path:Pluit-->Kapuk-->Palmerah-->KebayoranLama
--> Gandaria

Path: Pluit --> Grogol --> Ancol

Path: Pluit --> Grogol --> Menteng --> Kemayoran

Path:Pluit --> Kapuk --> Palmerah --> Kebayoran
Baru --> Kemang --> Gandaria

Path:Pluit --> Grogol --> Menteng --> Tebet (Goal)

4.5 Kombinasi dengan GPS

Dengan memperhatikan solusi dari persoalan *path finding* dari Pluit – Tebet, maka sebuah aplikasi GPS juga dapat dikembangkan agar menghasilkan sebuah solusi yang optimal. Dimana, setiap posisi atau jarak yang dihasilkan oleh aplikasi GPS dikombinasikan dengan sebuah nilai heuristik yang didasarkan pada beberapa fenomena yang terjadi di alam, seperti: kemacetan lalu lintas.

Melalui makalah ini, diusulkan penggunaan algoritma A* dalam pencarian solusi optimal untuk suatu masalah pencarian jalur dengan menggunakan GPS. Dimana, aplikasi GPS tidak hanya memperhitungkan jarak yang ada antara satu posisi dengan posisi yang lain, tetapi menambahkan fungsi heuristik (seperti: perkiraan kemacetan, jumlah persimpangan dan lampu lalu lintas) sehingga hasil yang didapat lebih optimal.

Pada masalah kali ini, fungsi heuristik yang diusulkan diterapkan pada aplikasi GPS adalah memperhitungkan jumlah perempatan dan lampu lalu lintas yang ada dalam mencapai suatu posisi. Hal ini menjadi usulan karena beberapa daerah di Indonesia (seperti: Jakarta) mempunyai tingkat kemacetan dan kepadatan lalu lintas yang tinggi. Oleh karena itu, penggunaan aplikasi GPS dengan algoritma A* ini diharapkan membantu masyarakat dalam mencari jalur yang optimal untuk menghindari kemacetan dan mempercepat waktu tempuh yang dilakukan.

5. Analisis

Hasil eksperimen yang dilakukan terhadap penerapan algoritma A* dalam aplikasi GPS akan menghasilkan beberapa analisa, sebagai berikut:

1. Algoritma A* akan menghasilkan solusi yang optimal sesuai dengan fungsi heuristik yang diterapkan.
2. Penggunaan fungsi heuristik pada algoritma A* menyebabkan algoritma A* membutuhkan

space memori yang besar, karena pada setiap simpul akan terjadi perhitungan *cost* yang telah dilalui dan perkiraan *cost* dari simpul tersebut ke simpul tujuan.

3. Penerapan algoritma A* pada Aplikasi GPS akan meningkatkan nilai guna dan keoptimalan dari aplikasi tersebut.

6. Kesimpulan

Algoritma A* merupakan salah satu algoritma Branch & Bound yang selalu memberikan solusi yang optimal. Algoritma A* menggunakan informasi tambahan (nilai heuristik) dalam pencarian solusi. Oleh karena itu, solusi yang optimal sangat tergantung kepada fungsi heuristik yang diterapkan. Penerapan algoritma A* dalam penentuan suatu jalur untuk aplikasi GPS akan memberikan hasil yang mendekati kenyataannya, karena jarak tiap simpul (posisi) diketahui dan terdapat perkiraan dari suatu posisi tertentu ke posisi akhir.

7. Daftar Pustaka

- [1] Bogan Larry. 1998."GPS How it works", <http://ailab2.et.ntust.edu.tw/~ysy/search.htm#s> diakses tanggal 16 Mei 2006
- [2] Dana, Peter, H.1999."Global Positioning System Overview", <http://www.colorado.edu/geography/gcraft/notes/gps/gps.html>, diakses tanggal 16 Mei 2006.
- [3] Leslie, Tung dkk., "Cispace Applet", <http://www.cs.ubc.ca/labs/lci/CIspace>, Laboratory for Computational Intelligence at the University of British Columbia
- [4] Munir Rinaldi. 2006. "Diktat Kuliah Strategi Algoritmik"
- [5] Russell Stuart and Norvig Peter.2003."Artificial Intelligence", Pearson Educational International.