

PERANCANGAN DAN IMPLEMENTASI ALGORITMA A* PADA APLIKASI ANGKOT-FINDER DI KOTA BANDUNG UNTUK SMARTPHONE BERBASIS ANDROID

DESIGN AND IMPLEMENTATION OF A* ALGORITHM IN BANDUNG ANGKOT-FINDER APPLICATION FOR ANDROID SMARTPHONE

Didik Setiawan¹, Astri Novianty², Surya Michrandi Nasution³

^{1,2,3} Prodi S1 Sistem Komputer, Universitas Telkom Bandung

¹didiks_1@yahoo.com, ²astri_nov@yahoo.com, ³surya.michrandi@gmail.com

Abstrak

Sebagian besar orang Bandung masih menggunakan angkutan umum sebagai alat transportasi, salah satunya adalah angkot (Angkutan Kota). Karena kota Bandung adalah kota terbesar di Propinsi Jawa Barat, maka terdapat banyak jalan yang menyebabkan banyaknya jalur angkot. Sekarang permasalahannya adalah bagaimana cara orang-orang yang ada di kota Bandung dapat mengetahui jalur angkot tersebut. Hal ini dikarenakan kota Bandung merupakan kota yang memiliki banyak pendatang baru, seperti para mahasiswa yang merantau yang masih awam dengan kota Bandung.

Oleh karena itu, diperlukan sebuah aplikasi *mobile* yang dapat membantu untuk menemukan angkot yang digunakan. Aplikasi ini membantu pengguna untuk mengetahui angkot mana yang harus ditumpangi sesuai dengan jalur yang terbaik. Aplikasi ini dirancang untuk berjalan pada sistem operasi *android*. Metode yang digunakan pada saat pencarian jalur adalah algoritma A*. Algoritma A* ini mencari jalur angkot dari posisi awal menuju tujuan yang diinginkan.

Aplikasi ini mampu memberikan rekomendasi angkot dan jalur yang digunakan untuk sampai ke tujuan berdasarkan tarif dan jarak.

Kata Kunci : Bandung, Angkutan Kota, Angkot, Algoritma A*, *Android*.

Abstract

Most people in Bandung still using public transport as a transportation, one of them is angkot (Angkutan Kota/Public Transportation). Because the city is the largest city in West Java, there are many streets that lead to many angkot route. Now the problem is how the people in the city can find out the angkot route. Because the city of Bandung is a city that has a lot of newcomers, such as the students who go abroad are still unfamiliar with the city of Bandung.

Therefore, we need a mobile application that can help to find out the angkot that will be used. This application helps the user to determine which ones should be rode in accordance with the best track. This application is designed to run on Android operating system. The method used when searching a path is A * Algorithm. A * algorithm will search angkot path from the initial position to the desired destination.

This application is able to provide recommendations of angkot and path that will be used to reach a destination based on the costs and distances.

Keywords: Bandung, Public Transportation, Angkot, Algorithm A *, *Android*.

1. Pendahuluan

Kota Bandung merupakan kota yang memiliki wilayah yang cukup luas sekaligus merupakan kota terbesar di Propinsi Jawa Barat. Untuk mencapai suatu tempat yang diinginkan, masyarakat membutuhkan alat transportasi. Namun tidak semua orang memiliki alat transportasi pribadi sehingga mereka menggunakan angkutan umum. Angkutan umum yang sering digunakan adalah angkot (Angkutan Kota).

Banyaknya jumlah angkot yang ada di kota Bandung, tentu saja membuat masyarakat membutuhkan informasi dan petunjuk mengenai jalur angkot. Masyarakat sering kali mengalami kesulitan pada saat mencari angkot yang harus ditumpangi dan jalur mana saja yang harus dilalui untuk mencapai tujuan mereka.

Oleh karena itu, diperlukan sebuah aplikasi *mobile* yang bisa menunjukkan jalur angkot sehingga bisa mencapai tujuan yang diinginkan. Aplikasi ini akan membantu pengguna untuk mengetahui angkot mana yang harus ditumpangi sesuai dengan jalur yang terbaik. Dengan mengimplementasikan *Artificial Intelligence* pada aplikasi ini, aplikasi ini dapat memilih jalur secara optimal, yaitu berdasarkan jarak terpendek dan tarif yang murah. Metode yang digunakan dalam pencarian jalurnya adalah algoritma A*. Algoritma A* ini akan mencari jalur angkot dari posisi awal menuju tujuan yang diinginkan.

Harapan yang ingin dicapai dari pembuatan aplikasi ini adalah dapat membantu masyarakat kota Bandung dalam mencari angkutan kota yang akan ditumpangi sehingga sampai ke tujuan yang diinginkan.

2. Material dan Metodologi

2.1 Angkutan Kota

Angkutan kota merupakan salah satu angkutan umum. Angkutan kota adalah alat transportasi kota yang berupa kendaraan umum dengan rute yang sudah ditentukan. Angkot digunakan sebagai metode transportasi untuk jarak pendek dan menengah terutama di daerah pinggiran kota[1]. Tujuannya adalah untuk membantu seseorang menjangkau tempat yang diinginkan. Angkutan kota dapat menaikkan dan menurunkan penumpang dimana saja asalkan masih berada dalam rute angkot itu sendiri.

Tarif angkutan kota biasanya ditetapkan oleh pemerintah daerah setempat, namun orang yang menumpang dengan jarak yang pendek atau anak sekolah biasanya dikenai tarif yang lebih murah. Hal ini tidak dirumuskan dalam peraturan tertulis, namun menjadi praktik umum yang digunakan oleh sopir-sopir angkutan kota.

Umumnya sebuah angkutan kota memiliki peraturan berkaitan dengan jumlah maksimal penumpang dan jumlah maksimal berat barang yang harus dibawa. Tetapi tidak jarang hal ini dilanggar oleh para pemilik angkutan kota. Terkadang juga sebuah angkutan kota sering menepi dalam waktu yang lama untuk menunggu penumpang agar bisa terisi penuh. Jalur operasi sebuah angkutan kota dapat diketahui melalui warna atau kode berupa huruf atau angka yang ada di badannya.

2.2 Algoritma A*

Algoritma A* merupakan algoritma *Best First Search* yang menggabungkan *Uniform Cost Search* dengan *Greedy Best-First Search*[2]. Beberapa terminologi dasar yang terdapat pada algoritma a* adalah :

- Simpul (*node*) adalah petak-petak kecil sebagai representasi dari area pencarian (*pathfinding*).
- Simpul asal/mulai (*source node*) adalah sebuah terminologi untuk posisi awal sebuah benda.
- Simpul akhir/tujuan (*destination node*) adalah tempat tujuan yang ingin dicapai pada pencarian.
- Simpul sekarang (*current node*) adalah simpul terbaik sebelumnya yang dipilih dan menjadi titik acuan untuk membangkitkan simpul tetangganya (*adjacent*).
- Simpul tetangga (*neighbor node*) adalah simpul-simpul yang bertetangga dengan *current node*.
- *Open list* adalah tempat menyimpan data simpul yang diakses dari simpul asal (*source node*) atau dari tetangga simpul sekarang (*current node*) yang belum pernah berada di *open list* maupun *closed list*.
- *Closed list* adalah tempat menyimpan simpul yang pernah menjadi simpul sekarang (*current node*).
- *Came from* adalah tempat yang menyimpan data ketetanggaan dari suatu simpul, misalnya *y came from x* artinya *neighbor node y* dari *current node x*
- *Walkability* adalah sebuah atribut yang menyatakan apakah sebuah simpul dapat atau tidak dapat dilalui oleh *current node*

$$f(x) = g(x) + h(x) \quad (1)$$

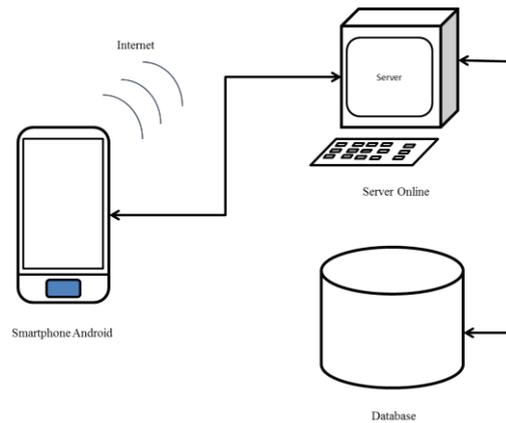
Dalam persamaan 1, $g(x)$ adalah biaya sebenarnya (*actual cost*), biasanya berupa jarak yang pernah ditempuh. Tapi dalam kasus tertentu, $g(x)$ dapat berupa penjumlahan antara jarak dengan biaya medan (*terrain cost*). Sedangkan $h(x)$ adalah biaya heuristiknya (*heuristic cost*), biasanya berupa perkiraan jarak dari simpul sekarang (*current node*) ke simpul tujuan (*destination node*). Dan $f(x)$ adalah total biaya, jumlah dari biaya sebenarnya dan biaya heuristik.

Seperti yang ditunjukkan dalam persamaan 1, algoritma ini menggunakan fungsi biaya heuristik simpul x untuk menentukan urutan yang akan melintasi dalam prioritas antrian[3]. Fungsi biaya dapat dibagi menjadi dua bagian, yaitu fungsi biaya jalan terakhir dan fungsi biaya jalan masa depan[3].

3. Pembahasan

3.1 Gambaran Umum Sistem

Aplikasi yang akan dibuat diberi nama *angkot-Finder*. *Angkot-Finder* merupakan aplikasi *mobile* berbasis *android* yang berguna untuk memberikan informasi kepada pengguna mengenai jalur maupun informasi angkutan kota atau yang sering disebut “angkot”. Dengan mengimplementasikan *Artificial Intelligence* pada aplikasi ini, aplikasi ini diharapkan dapat memberikan hasil yang optimal berdasarkan jarak terpendek dan harga termurah. Metode yang digunakan pada saat pencarian jalur adalah algoritma A*.

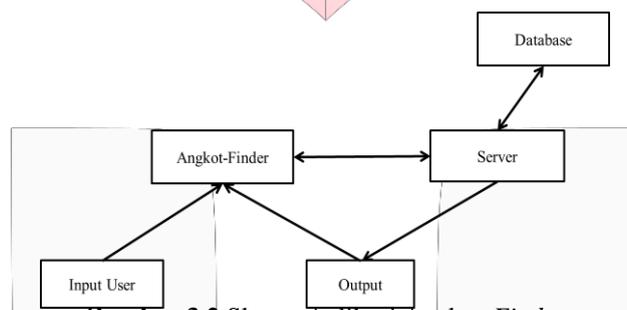


Gambar 3.1 Gambaran Umum Sistem

Pada gambar 3.1, terlihat bagaimana cara kerja aplikasi. Pada saat pengguna ingin menggunakan aplikasi angkot-*Finder*, Pengguna harus mengaktifkan internet terlebih dahulu. Setelah itu, pengguna bisa memilih jalan atau tempat yang diinginkan sebagai titik awal dan titik tujuan yang ingin dicapai. Aplikasi ini kemudian akan mengirimkan data yang dimasukkan oleh pengguna dari *Smartphone android* ke server. Server akan mengolah data yang dikirimkan dan hasilnya akan dikirim kembali ke *Smartphone android* pengguna.

3.2 Skema Aplikasi

Skema umum aplikasi yang akan dibangun dapat dilihat pada gambar di bawah ini :



Gambar 3.2 Skema Aplikasi Angkot-*Finder*

Beberapa penjelasan mengenai skema di atas adalah :

1. Pengguna memilih nama jalan atau tempat yang diinginkan sebagai titik awal dan titik tujuan yang ingin dicapai.
2. Masukan yang diterima akan dikirim oleh aplikasi ke server.
3. Server mengolah data yang dikirim oleh aplikasi dengan mengimplementasikan data-data yang ada pada *database*.
4. Hasil dari pengolahan data di server akan dikirim kembali dan ditampilkan pada *Smartphone* dalam bentuk string yang berupa informasi mengenai angkot yang harus digunakan.

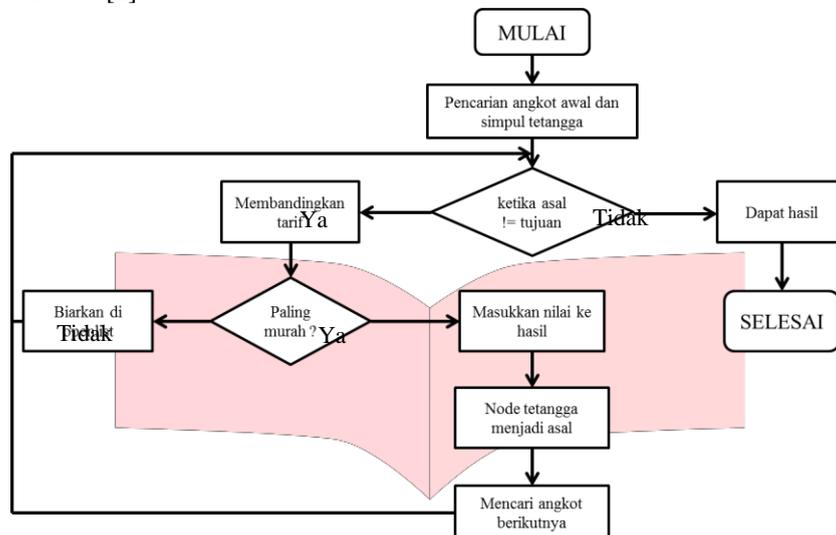
3.3 State space

- Pada pencarian, yang menjadi state adalah simpul/*node* pada graf yang mempresentasikan jalan yang dilalui angkot.
- Karena alternatif *next-state* yang mungkin dipilih itu dihasilkan oleh pemilihan angkot, maka dalam permasalahan ini angkot dapat berperan seperti operator penghasil alternatif *next-state*.
- Karena target yang ingin dicapai adalah tarif perjalanan semurah mungkin, maka dapat dirancang fungsi evaluasi sebagai berikut.
 - Biaya sebenarnya (*actual cost*) = $g(n)$ = nilai tarif ketika menggunakan angkot tertentu.
 - Biaya perkiraan (*estimation cost*) = $h(n)$ = estimasi tarif pada sisa perjalanan yang masih ada.
 - Jika target *point/node* ada pada rute angkot yang telah dipilih, $h(n) = 0$.
 - Jika tidak, $h(n)$ = tarif angkot yang terakhir dinaiki + (sisa jarak / total jarak tempuh angkot) x tarif angkot.
 - Biaya total = $f(n) = g(n) + h(n)$.
- Untuk mendapatkan target yang ingin dicapai adalah jarak perjalanan yang sedekat mungkin, maka dapat dirancang fungsi evaluasi sebagai berikut.
 - Biaya sebenarnya (*actual cost*) = $g(n)$ = jarak yang telah dicapai hingga *node/point* tertentu.

- Biaya perkiraan (*estimation cost*) = $h(n)$ = estimasi jarak dari *node/point* yang terakhir hingga ke tujuan.
- Biaya total = $f(n) = g(n) + h(n)$.

3.4 Algoritma Pencarian Jalur

Pada aplikasi yang dibuat, algoritma pencarian jalur menggunakan algoritma A*. Algoritma A* adalah algoritma *Best First Search* yang menggabungkan *Uniform Cost Search* dengan *Greedy Best-First Search*[2].



Gambar 3.4 Alur pencarian jalur menggunakan algoritma A*

Beberapa penjelasan mengenai alur pencarian pada gambar 3.4 :

1. Mencari angkot awal dan simpul tetangga dari *node* awal yang dimasukkan oleh pengguna.
2. Melakukan pengulangan dengan syarat awal tidak sama dengan tujuan maka akan membandingkan tarif dari angkot yang didapat dari proses sebelumnya, jika awal sama dengan tujuan maka proses perulangan selesai dan akan mendapatkan hasil.
3. Setelah membandingkan tarif, angkot yang memiliki tarif yang paling murah, akan dimasukkan ke dalam hasil, *node* tetangganya akan menjadi asal untuk pencarian angkot berikutnya.
4. Angkot awal yang belum masuk ke hasil di masukkan ke dalam *open list* untuk dibandingkan dengan angkot yang di dapat dalam pencarian berikutnya.

Tabel 3.1 Pengujian *white box* pencarian jalur berdasarkan harga dan jarak

Kasus dan Hasil Uji (Data Normal)				
Data Masukan	Yang diharapkan		Pengamatan	Kesimpulan
	Berdasarkan Harga	Berdasarkan Jarak		
Bumi Asri Residence – Jl. Aksan	20. Ciroyom – Bumi Asri Tarif : Rp 3.500	20. Ciroyom – Bumi Asri Tarif : Rp 3.500	Sesuai dengan yang diharapkan	Diterima
	04. Abdul Muis – Elang : 3.500	04. Abdul Muis – Elang Tarif : 3.500		
	Total Tarif : Rp 7.000	Total Tarif : Rp 7.000		
Bumi Asri Residence – Jl. Babakan Ciparay	20. Ciroyom – Bumi Asri Tarif : Rp 3.500	20. Ciroyom – Bumi Asri Tarif : Rp 3.500	Sesuai dengan yang diharapkan	Diterima
	34. Sadang Serang – Caringin Tarif : Rp 4.500	34. Sadang Serang – Caringin Tarif : Rp 4.500		
	Total Tarif : Rp 8.000	Total Tarif : Rp 8.000		
Bumi Asri Residence – Jl. BKR	20. Ciroyom – Bumi Asri Tarif : Rp 3.500	20. Ciroyom – Bumi Asri Tarif : Rp 3.500	Sesuai dengan yang diharapkan	Diterima
	21. Ciroyom – Cikudapateuh Tarif : Rp 4.000	27. Cijerah – Ciwastra – Derwati Tarif : Rp 4.500		
	Total Tarif : Rp 7.500	35. Cibaduyut – Karang Setra Tarif : Rp 4.500 Total Tarif : Rp 12.500		
Bumi Asri	20. Ciroyom – Bumi Asri	20. Ciroyom – Bumi Asri	Sesuai	Diterima

Residence – Jl. Sudirman	Tarif : Rp 3.500	Tarif : Rp 3.500	dengan yang diharapkan
	Total Tarif : Rp 3.500	23. Sederhana – Cijerah Tarif : Rp 3.500	
		Total Tarif : Rp 7.000	

Data masukkan dari tabel 3.1 di masukkan ke dalam tabel 3.2 di bawah untuk dilihat optimasi pencarian jalur angkot berdasarkan jarak dan harga.

Tabel 3.2 Pengujian terhadap optimasi pencarian jalur angkot

Pengujian	Menampilkan informasi jalur angkot	Informasi jalur angkot yang direkomendasikan		Informasi Berdasarkan jarak dan harga	
		Benar	Salah	Optimal	Tidak
1	√	√		√	
2	√	√		√	
3	√	√			√
4	√	√			√

Keterangan untuk tabel 4.4 :

- Pengujian 1 : Bumi Asri Residence – Jl. Aksan
- Pengujian 2 : Bumi Asri Residence – Jl. Babakan Ciparay
- Pengujian 3 : Bumi Asri Residence – Jl. BKR
- Pengujian 4 : Bumi Asri Residence – Jl. Sudirman

4. Kesimpulan

Berdasarkan dari hasil pengujian dan analisis pada bab iv dapat ditarik beberapa kesimpulan sebagai berikut :

1. Aplikasi angkot-*finder* yang dibangun sudah berhasil menampilkan info mengenai jalur dan angkot yang harus digunakan.
2. Pencarian jalur dengan mengimplementasikan algoritma A* berjalan cukup baik.
3. Aplikasi mampu memberikan rekomendasi jalur dan angkot yang harus digunakan untuk sampai ke tujuan berdasarkan harga.
4. Aplikasi mampu memberikan rekomendasi jalur dan angkot yang harus digunakan untuk sampai ke tujuan berdasarkan jarak meskipun dari segi harga masih belum optimal. Dari hasil pengujian yang dilakukan, performa aplikasi bergantung pada trafik jaringan provider yang digunakan oleh *smartphone* dan trafik jaringan pada server. Semakin padat trafik jaringan maka semakin menurun performa aplikasi, begitu pula sebaliknya

DAFTAR PUSTAKA

- [1] Kuntara, I, Ridha, M, Sabran, K.U. (2014). "Best price algorithm in finding routes based on unconventional public transportation : Indonesian suburban regions." *Cyber and IT Service Management (CITSM)*, 75-77.
- [2] Suyanto, (2007) "Artificial Intelligence Searching, Reasoning, Planning dan Learning," Bandung : INFORMATIKA.
- [3] S. Rabin, (2000) "A* Speed Optimizations." In *Game Programming Gems*, edited by Mark Deloura. Hingham, M A : Charles River Media.