

## IMPLEMENTASI ALGORITMA *BEST PATH PLANNING* UNTUK PENCARIAN RUTE TRANS JOGJA

Anik Budiati<sup>1)</sup>, P. Insap Santosa<sup>2)</sup>, Wing Wahyu W.<sup>3)</sup>

<sup>1,2)</sup>Jurusan Teknik Elektro dan Teknologi Informasi Universitas Gadjah Mada Yogyakarta  
Gedung Teknik Elektro dan Teknologi Informasi Fakultas Teknik UGM, Jl. Grafika No. 2 Yogyakarta  
e-mail : [anik.cio.8a@mail.ugm.ac.id](mailto:anik.cio.8a@mail.ugm.ac.id), [insap@jtet.gadjahmada.edu](mailto:insap@jtet.gadjahmada.edu), [wing@mti.ugm.ac.id](mailto:wing@mti.ugm.ac.id)

### Abstrak

Untuk membantu mengatasi kemacetan di Yogyakarta, Pemda DIY menyediakan bus TransJogja sebagai sarana transportasi publik. Akan tetapi, masih banyak masyarakat yang memilih menggunakan kendaraan pribadi untuk bepergian. Salah satu hal yang menyebabkan masyarakat malas menggunakan transportasi publik adalah ketika mereka harus pergi ke suatu tempat, tetapi mereka tidak tahu bus jalur berapa yang harus dinaiki untuk sampai ke lokasi tersebut. Penelitian ini bertujuan membuat suatu aplikasi untuk merencanakan rute perjalanan dari suatu halte asal ke halte tujuan. Algoritma yang digunakan untuk penentuan rute adalah algoritma *Best Path Planning*. Pada algoritma ini, rute terpilih adalah rute dengan jumlah transfer paling sedikit dan jarak tempuh terpendek. Penelitian dimulai dari proses mengumpulkan data, melakukan perancangan aplikasi, kemudian mengimplementasikan rancangan tersebut menjadi sebuah aplikasi. Dari hasil uji coba, sistem berhasil menemukan rute terpendek dengan jumlah transfer seminimal mungkin.

**Kata Kunci** : *Best Path Planning*, *TransJogja*, transportasi publik

### 1. PENDAHULUAN

DIY merupakan salah satu kota tujuan wisata di Indonesia, selain itu perannya sebagai kota pendidikan membuat tingkat mobilitas baik penduduk maupun pendatang cukup tinggi. Akan tetapi penambahan jumlah kendaraan di DIY tidak diikuti dengan penambahan ruas jalan yang sesuai dengan kebutuhan. Hal ini dapat dilihat dari jumlah kendaraan yang melewati ruas jalan hampir melebihi kapasitas jalan yang disediakan sehingga menyebabkan kemacetan di beberapa ruas jalan. Untuk mengurangi kemacetan tersebut sudah saatnya masyarakat beralih ke transportasi publik sebagai sarana transportasi mereka sehari-hari. Pemda DIY bekerja sama dengan penyedia jasa menyelenggarakan pelayanan transportasi publik yang dikenal sebagai bus TransJogja. Sayangnya, banyak orang lebih memilih menggunakan kendaraan pribadi daripada transportasi publik. Salah satu alasannya adalah kurangnya informasi rute bus menuju tempat tujuan mereka. Untuk menarik minat masyarakat agar bersedia beralih menggunakan transportasi publik, salah satu cara yang dapat dilakukan adalah dengan menyediakan informasi rute bus untuk membantu penumpang merencanakan perjalanan mereka (Chao dkk, 2001).

Pencarian rute terpendek merupakan suatu masalah yang paling banyak dibahas dan dipelajari sejak akhir tahun 1950 (Purwananto, 2005). Salah satu aplikasi pencarian rute terpendek adalah pada masalah transportasi publik. Masalah pencarian rute terpendek pada sistem transportasi publik tidak bisa disamakan dengan pencarian rute terpendek pada area tertentu. Proses pencarian rute terpendek pada sistem transportasi publik perlu memperhatikan bahwa kendaraan umum hanya dapat melewati rute-rute tertentu yang telah ditentukan sebelumnya, dan penumpang tidak dapat meminta pengemudi untuk mengganti rutenya (Chao dkk, 2001). Banyak masalah perencanaan rute yang dapat diselesaikan dengan menghitung jarak terpendek menggunakan pemodelan yang cocok, graf berbobot yang merepresentasikan jaringan transportasi. Pada transportasi publik, perlu diperhitungkan waktu yang diperlukan untuk melakukan transfer antar rute (Geisberger, 2011). Semakin banyak transfer antar rute yang dilakukan, maka semakin banyak waktu dan biaya yang harus dikeluarkan oleh penumpang (Chao, 2002). Untuk menghindari hal tersebut, aplikasi ini menerapkan algoritma *Best Path Planning* untuk mencari rute terpendek Bus Trans Jogja yang beroperasi di Daerah Istimewa Yogyakarta.

### 2. TINJAUAN PUSTAKA

Masalah pencarian rute terpendek pada sistem transportasi publik tidak bisa disamakan dengan pencarian rute terpendek pada area tertentu. Proses pencarian rute terpendek pada sistem transportasi publik perlu memperhatikan bahwa kendaraan umum hanya dapat melewati rute-rute tertentu yang telah ditentukan sebelumnya, dan penumpang tidak dapat meminta pengemudi untuk mengganti rutenya (Chao dkk, 2001). Algoritma-algoritma yang dapat digunakan untuk menyelesaikan masalah perencanaan rute terbaik pada transportasi publik pernah diteliti oleh Hsiao dkk, Geisberger, Liu dkk, dan Chao-Lin Liu.

Hsiao dkk menawarkan algoritma solusi berdasarkan teknik optimasi koloni semut (*ant colony optimization*) untuk mencari rute terpendek dari suatu lokasi asal ke lokasi tujuan pada peta dengan mempertimbangkan kondisi lalu lintas (Hsiao, 2004). Banyak masalah perencanaan rute yang dapat diselesaikan dengan menghitung jarak terpendek menggunakan pemodelan yang cocok, graf berbobot yang merepresentasikan jaringan transportasi. Pada transportasi publik, perlu diperhitungkan waktu yang diperlukan untuk melakukan transfer antar

rute (Geisberger, 2011). Liu dkk menggunakan matrik konektivitas (*connectivity matrices*) yang merupakan pengembangan dari matrik kedekatan (*adjacency matrices*) untuk menyelesaikan masalah batasan jalur pada transportasi publik. Strategi lain yang ditawarkan adalah menggunakan *hubs* (Chao dkk, 2001). Dengan mempertimbangkan bahwa transfer antar rute pada transportasi publik memerlukan lebih banyak waktu dan biaya, digunakan matrik transisi (*transition matrices*) untuk memperhitungkan jumlah transfer yang dibutuhkan oleh penumpang (Chao, 2002).

Pencarian rute terpendek dan rute alternatif menggunakan algoritma Dijkstra pernah dilakukan oleh Haryanto dengan mengambil studi kasus Kota Surakarta. Aplikasi yang dibuat berbasis WAP menggunakan PHP dan MySQL. Penggunaan teknologi WAP memungkinkan aplikasi ini dapat diakses melalui telepon seluler yang mempunyai fasilitas WAP. Penelitian ini cocok untuk orang yang bepergian menggunakan kendaraan pribadi (Haryanto, 2008).

### 3. METODE PENELITIAN

#### 3.1. Pengumpulan Data

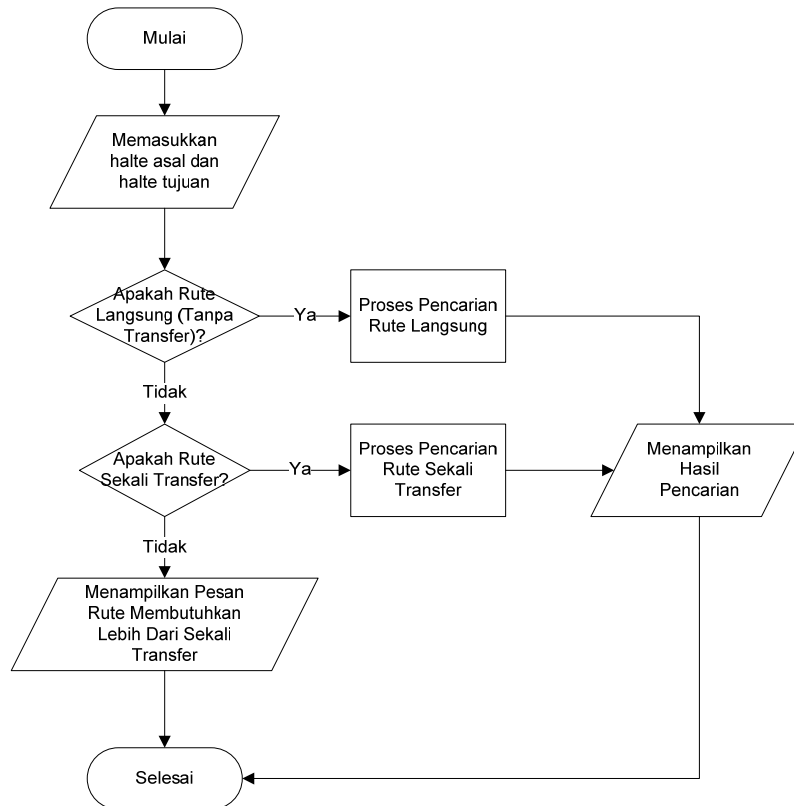
Data yang dibutuhkan untuk pembuatan aplikasi ini yaitu :

1. Data Halte, berisi nama halte (shelter) dimana penumpang dapat naik/turun dari Bus TransJogja. Saat ini terdapat 98 halte yang tersebar di wilayah DIY.
2. Data Jalur, saat ini terdapat 8 jalur Bus TransJogja yang aktif beroperasi yaitu jalur 1A, 1B, 2A, 2B, 3A, 3B, 4A, dan 4B
3. Data Rute, berisi urutan shelter yang dilewati oleh masing-masing jalur Bus TransJogja

Data-data di atas didapatkan dari Dinas Perhubungan, Komunikasi dan Informatika DIY yang bertanggung jawab atas operasional Bus TransJogja di DIY.

#### 3.2. Perancangan Aplikasi

Proses pencarian rute dimulai dari proses input halte asal dan halte tujuan. Kemudian dilakukan pengecekan apakah dari halte asal, halte tujuan dapat dicapai menggunakan satu jalur bus TransJogja, perlu satu kali transfer, atau perlu lebih dari satu kali transfer. Kemudian dilakukan proses pencarian rute-rute yang mungkin. Rute yang terpilih adalah rute dengan jumlah transfer paling sedikit dan jarak tempuh minimal. Jika rute ditemukan, maka rute akan ditampilkan. Apabila untuk mencapai halte tujuan diperlukan lebih dari sekali transfer, maka ditampilkan pesan bahwa rute yang dicari membutuhkan lebih dari satu kali transfer. *Flowchart* aplikasi dapat dilihat pada Gambar 1.



Gambar 1. *Flowchart* Aplikasi Pencarian Rute TransJogja

#### 3.3. Implementasi

Proses implementasi adalah proses menterjemahkan perancangan aplikasi ke dalam bahasa pemrograman. Aplikasi ini dibangun menggunakan IDE NetBeans 7.1.1 dan JDK 1.6.0\_31. Bahasa pemrograman yang digunakan adalah bahasa Java.

### 3.4. Pengujian

Pengujian yang dilakukan adalah uji fungsionalitas aplikasi. Pengujian ini bertujuan untuk melihat apakah fungsi-fungsi dasar dari aplikasi ini bisa berjalan.

## 4. ALGORITMA BEST PATH PLANNING

Algoritma *Best Path Planning* merupakan algoritma yang diusulkan oleh Chao Lin Liu untuk mengatasi masalah pencarian rute pada transportasi publik dengan meminimalkan jumlah transfer antar rute untuk efisiensi waktu dan biaya (Chao, 2002). Algoritmanya adalah sebagai berikut :

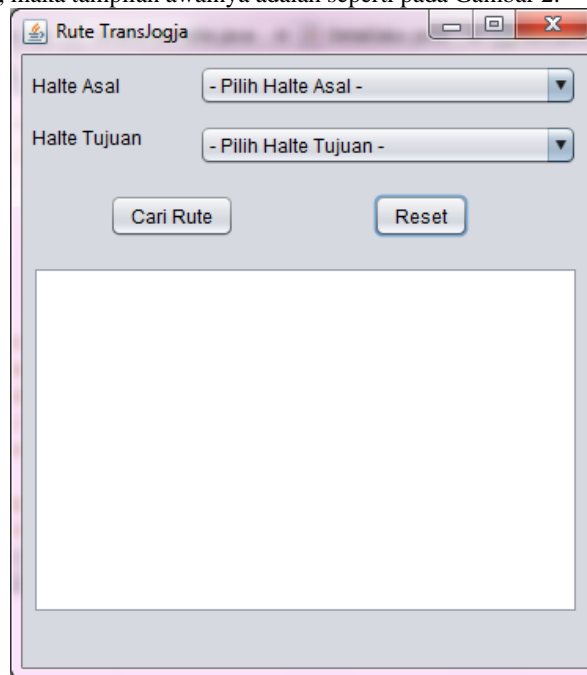
Misalkan  $o$  dan  $d$  merepresentasikan lokasi asal (*origin*) dan tujuan (*destination*), maka :

1. Kasus sederhana : jika  $o=d$ , tampilkan pesan bahwa lokasi asal sama dengan lokasi tujuan, tidak perlu rute transportasi publik.
2. Tanpa Transfer : jika  $Dservice(o,d)$  tidak kosong, tampilkan rute-rute yang berada pada himpunan  $Dservice(o,d)$ .
3. Satu Transfer : jika  $\exists i \in SR(O)$ ,  $\exists j \in SR(D)$  dan  $\exists s \in CS(i,j)$  sehingga baik  $Dservice(O,s)$  dan  $Dservice(s,D)$  tidak kosong, maka terdapat solusi satu transfer.
4. Kasus Non-trivial : Temukan rencana perjalanan melalui *hubs*.

$Dservice(o,d)$  merupakan himpunan rute layanan langsung dimana kita dapat melakukan perjalanan dari  $o$  ke  $d$  tanpa perlu transfer bus.  $SR(s)$  merupakan himpunan rute yang melewati lokasi  $s$ .  $CS(i,j)$  merupakan himpunan lokasi pemberhentian (halte) yang sama dari rute  $i$  dan  $j$ . *Hubs* adalah lokasi-lokasi dimana beberapa layanan terkonsentrasi sehingga memberikan kesempatan pada penumpang untuk melakukan transfer dari satu rute ke rute lain (Chao, 2002). Rute yang terpilih adalah rute dengan jumlah transfer paling sedikit dan jarak tempuh paling pendek. Jarak tempuh merupakan total jarak dari halte asal ke halte tujuan.

## 5. HASIL DAN PEMBAHASAN

Ketika aplikasi dijalankan, maka tampilan awalnya adalah seperti pada Gambar 2.



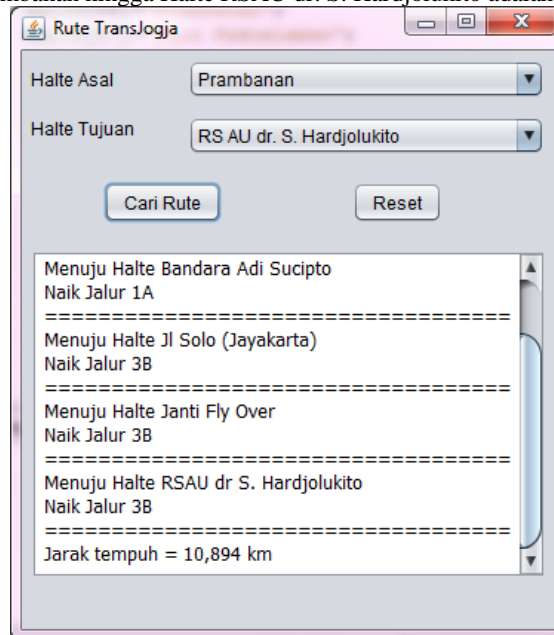
Gambar 2. Tampilan awal aplikasi

Pada pengujian dengan rute langsung (tanpa transfer), hasilnya seperti pada Gambar 3. Penumpang dari Halte Jl Solo (KR 1) dapat pergi ke Halte Jl Solo (Jogja Bisnis) menggunakan Jalur 1A dengan jarak tempuh 5,498 km.



Gambar 3. Pengujian Rute Langsung

Hasil pengujian yang membutuhkan satu kali transfer dapat dilihat pada Gambar 4. Untuk menuju Halte RSAU dr. S. Hardjolutito dari Halte Prambanan, penumpang dapat menaiki Jalur 1A hingga Halte Bandara Adi Sucipto kemudian transit di Halte Bandara Adi Sucipto dan menaiki Jalur 3B sampai Halte RSAU dr. S. Hardjolutito. Jarak tempuh dari Halte Prambanan hingga Halte RSAU dr. S. Hardjolutito adalah 10,894 km.



Gambar 4. Pengujian Satu Kali Transfer

## 6. KESIMPULAN

Dari hasil pengujian fungsionalitas, aplikasi berhasil menemukan rute terpendek dengan jumlah transfer minimal (tanpa transfer atau satu kali transfer). Untuk setiap dua jalur yang berbeda, selalu terdapat setidaknya satu halte yang dapat digunakan untuk transit penumpang (transfer/pindah jalur), sehingga aplikasi akan selalu menemukan solusi rute terpendek tanpa transfer atau melalui satu kali transfer.

**DAFTAR PUSTAKA**

- Chao Lin Liu, T. W. (2001). Path Planning Algorithms for Public Transportation System. *The Fourth International IEEE Conference on Intelligent Transportation Systems*. Oakland, California, USA.
- Geisberger, R. (2011). *Advanced Route Planning in Public Transportation*. Munchen: Karlsruhe Institut für Technologie.
- Haryanto. (2008). *Implementasi WAP pada Telepon Seluler untuk Pencarian Rute Jalan Terpendek Menggunakan Algoritma Dijkstra (Studi Kasus : Kota Surakarta)*. Yogyakarta: Universitas Gadjah Mada.
- Liu, C. L. (2002). Best Path Planning for Public Transportation System. *The IEEE 5th International Conference on Intelligent Transportation Systems*.
- Ying-Tung Hsiao, C.-L. C.-C. (2004). Ant Colony Optimization for Best Path Planning. *International Symposium on Communications and Information Technologies 2004 (ISCIT 2004)*. Sapporo, Japan.
- Yudhi Purwananto, D. P. (2005). Implementasi dan Analisis Algoritma Pencarian Rute Terpendek di Kota Surabaya. *Jurnal Penelitian dan Pengembangan Telekomunikasi*, 10 (2), 94-101.