

PERBANDINGAN RUTE OPTIMUM HASIL PERHITUNGAN ALGORITMA *DIJKSTRA* DAN *A-STAR* UNTUK SIRKULASI PAKET JASA EKSPEDISI JNE DI D.I. YOGYAKARTA

Latief Zainulhayat
latief.zainulhayat@gmail.com

Taufik Hery Purwanto
taufik_hp@yahoo.com

Abstract

The development of Geographic Information Systems (GIS) and the determination of optimum route algorithm can generate alternative solution in various fields related to traffic phenomenon. This study aims to know about the ability of Dijkstra's algorithm and the A (A-star) in determination of JNE package services circulation's optimum route as well as comparing the process and the result. The study was conducted in the province of Yogyakarta where it is the area of D.I JNE Yogyakarta branch. Materials used in this study are road network data from the Department of Transportation D.I. Yogyakarta, RBI Map and also plan of travel time as the impedance. Some of JNE's agents were used as a point of origin meanwhile JNE's outbound warehouse were used as destination. The optimum route calculation was done using Dijkstra's algorithm and the A-star who became an innate function of PostGIS in PostgreSQL database. The results of this study indicate that the optimum route calculation using Dijkstra's algorithm and the A-star produce the same route although the parameters used in the lines of code are different. Broadly speaking, the process carried out by A-star algorithm is 0-13% faster than Dijkstra's algorithm.*

Key words : optimum route, Dijkstra, A-star, JNE

Abstrak

Perkembangan Sistem Informasi Geografis (SIG) dan algoritma penentuan rute optimum dapat menghasilkan alternatif solusi masalah di berbagai bidang yang berkaitan dengan fenomena lalu lintas. Penelitian ini bertujuan untuk mengetahui kemampuan algoritma *Dijkstra* dan *A** (*A-star*) untuk menentukan rute optimum sirkulasi paket jasa ekspedisi JNE serta membandingkan proses dan hasilnya. Penelitian dilakukan di provinsi D.I Yogyakarta sebagai area layanan JNE cabang Yogyakarta. Bahan yang digunakan meliputi data jaringan jalan dari Dinas Perhubungan D.I. Yogyakarta dan Peta Rupa Bumi Indonesia dengan waktu tempuh rencana sebagai impedansinya. Lokasi yang digunakan sebagai titik asal dan tujuan rute merupakan beberapa agen JNE dan gudang *outbond* JNE. Perhitungan rute optimum dilakukan menggunakan algoritma *Dijkstra* dan *A-star* yang menjadi fungsi bawaan *PostGIS* dalam basisdata *PostgreSQL*. Hasil dari penelitian ini menunjukkan bahwa perhitungan rute optimum menggunakan algoritma *Dijkstra* dan *A-star* menghasilkan rute yang sama meskipun parameter yang digunakan dalam baris kodenya berbeda. Secara garis besar proses yang dilakukan algoritma *A-star* bisa 13% lebih cepat dibanding algoritma *Dijkstra*.

Kata kunci: rute optimum, *Dijkstra*, *A-star*, JNE

PENDAHULUAN

Aktivitas pertukaran barang dan jasa yang terus meningkat menuntut pergerakan dan pengiriman untuk memenuhi kebutuhan antar daerah. Transportasi menjadi elemen utama dalam berlangsungnya jasa pengiriman. Menurut Hoel (2011), tujuan adanya transportasi adalah menciptakan mekanisme pertukaran barang, manusia maupun informasi yang pada akhirnya akan mendukung perkembangan ekonomi di satu wilayah. Transportasi berarti pemindahan barang dan/atau manusia dari tempat asal ke tempat tujuan. Salah satu penyedia jasa pengiriman paket terbesar di Indonesia adalah JNE telah mampu menangani hingga 16 juta pengiriman dalam 1 bulan dan 60% - 70% dari total tersebut adalah pengiriman *e-commerce* (jne.co.id). Memperpanjang rentang waktu pelayanan memang dapat dijadikan langkah untuk meningkatkan layanan, cara lain yang dapat dilakukan adalah meningkatkan efisiensi rute perjalanan dalam pengantaran paket.

Kondisi lalu lintas D.I Yogyakarta yang dewasa ini semakin padat menjadi tantangan tersendiri bagi jasa ekspedisi/pengiriman paket untuk menjaga kualitas layanannya dari segi ketepatan waktu pengiriman, khususnya untuk layanan YES (Yakin Esok Sampai) yang menjanjikan waktu pengiriman hanya sebatas 1-2 hari. Penentuan rute optimum merupakan solusi untuk mengatasi kondisi tersebut diatas. Rute optimum sendiri merupakan rangkaian kombinasi ruas jalan yang diambil pengendara dari titik awal menuju titik tujuan secara optimal dari segi waktu, jarak maupun pertimbangan lain yang dijadikan sebagai bobot hitung pada setiap segmen jalan. Kemampuan SIG dalam menyampaikan informasi secara visual baik dari deskripsi objek, fenomena maupun distribusi spasialnya mempermudah manusia untuk melakukan pengambilan keputusan dengan kondisi dan pilihan tertentu. Gabungan data spasial dan non-spasial memodelkan keadaan di lapangan (Maantay, 2006). Penentuan rute merupakan salah satu fenomena yang dapat disimulasikan menggunakan Sistem Informasi Geografi. Sistem informasi geografi merupakan cabang ilmu dalam geografi yang berbasis pada perangkat lunak komputer untuk membangun bentuk/model dan analisis digital dari permukaan bumi untuk menghasilkan informasi yang memiliki referensi spasial yang tepat dan akurat

(Suryantoro, 2013). Informasi yang dihasilkan dari oleh sistem informasi geografi merupakan output dari serangkaian proses yang dilakukan oleh sistem komputer, di mana proses tersebut meliputi menangkap, menyimpan, query, analisis dan pada akhirnya menampilkan informasi yang memiliki referensi keuangan/spasial (Chang, 2012).

Proses dan hasil dari penentuan rute dipengaruhi oleh algoritma yang bekerja dibelakangnya. Algoritma adalah metode khusus yang tepat dan terdiri atas serangkaian langkah terstruktur dan dituliskan secara sistematis, yang akan dikerjakan untuk menyelesaikan suatu masalah dengan menggunakan bantuan komputer. Penelitian ini mengambil dua algoritma penentuan rute optimal untuk dibandingkan, algoritma tersebut yaitu algoritma *Dijkstra* dan *A** (baca; A-star) yang mewakili algoritma dari masa yang berbeda. *Dijkstra* merupakan algoritma tertua dengan *greedy-search* dimana perhitungan dilakukan ke semua segmen pada simpul yang sedang dihitung. Sedangkan algoritma *A** dilakukan berdasarkan fungsi heuristik dimana diperkirakan arah dari titik awal menuju titik tujuan terlebih dulu sebagai pertimbangan dalam memilih segmen dan simpul yang diikutsertakan dalam perhitungan. Menurut Merrifield (2010) algoritma *A** bekerja lebih optimal dibandingkan algoritma *Dijkstra* dengan mengurangi ruang perhitungan dalam *greedy-search*. Algoritma *A** berdasar fungsi heuristik dimana diperhitungkan garis lurus penghubung asal-tujuan. Pemilihan algoritma terbaik menjadi permasalahan dalam penentuan rute optimum untuk kasus distribusi paket, dimana masing-masing algoritma memiliki logika dan proses perhitungan yang berbeda. Perbedaan bisa dilihat dari proses yang berupa kestabilan sistem dengan berhasil atau tidaknya dan juga waktu proses berlangsung. Perbedaan selanjutnya dilihat dari hasil algoritma, mencakup total beban dari rute yang dipilih dan jumlah total segmen jalan yang ada dalam rangkaian rute.

Penelitian ini bertujuan untuk (1) mengetahui kemampuan algoritma *Dijkstra* dan *A** (A-star) untuk menentukan rute optimum sirkulasi paket jasa ekspedisi JNE di D.I Yogyakarta dan (2) membandingkan proses dan hasil rute optimum sirkulasi paket dari algoritma *Dijkstra* dan *A** (A-star).

METODE PENELITIAN

Penentuan rute optimum secara mendasar membutuhkan data masukan berupa jaringan jalan beserta impedansi tertentu yang digunakan untuk mewakili fenomena hambatan lalu lintas di lapangan untuk setiap ruas jalannya beserta koordinat lokasi asal dan tujuan rute. Data jaringan jalan yang digunakan pada penelitian ini merupakan gabungan dan pelengkapan dari data jaringan jalan dari Peta Rupa Bumi Indonesia dan data jaringan jalan dari Dinas Perhubungan Provinsi D.I. Yogyakarta. Lokasi yang digunakan sebagai titik awal rute merupakan sebaran agen-agen JNE sedangkan titik akhirnya merupakan gudang *outbond* JNE, namun dari sekian banyak jumlah agen yang ada data tersebut diseleksi menurut pembagian area dan jadwal pengambilan berdasarkan data sistem kerja dan sirkulasi paket yang dilakukan JNE, khususnya untuk layanan YES (Yakin Esok Sampai). Data posisi kantor JNE dikumpulkan dengan melakukan koleksi posisi di lapangan atau plotting yang dilakukan menggunakan GPS untuk memperoleh posisi absolut dari lokasi di lapangan.

Data jaringan jalan yang dapat diproses dalam analisa jaringan mutlak membutuhkan topologi ruas jalan yang sudah terkoreksi. Topologi dalam jaringan jalan secara mutlak harus memiliki perpotongan yang jelas sehingga tiap ruas jalan saling terhubung. Setiap persimpangan jalan dalam basis data jaringan dianggap sebagai node dan ruas jalan dianggap sebagai segment. Setiap segment harus berpotongan pada pertemuannya di node. Node merupakan pertemuan dua atau lebih ruas jalan yang bersinggungan. Pada node inilah terjadi pemilihan ruas untuk meneruskan dan membentuk rute. Data spasial memiliki dua bagian penting yang membuatnya berbeda dari data lain, yaitu informasi spasial dan deskriptif/atribut (Yousman, 2004). Data atribut yang digunakan sebagai penentu dalam rute optimum disebut dengan impedansi. Impedansi diasumsikan sebagai harga yang mempengaruhi dan mewakili kondisi ruas jalan tersebut. Data jaringan jalan yang sudah dikoreksi topologinya dan diisi impedansi untuk setiap ruasnya selanjutnya dimasukkan ke dalam basisdata spasial berformat SQL (*structured query language*). Data jaringan jalan pada tahap awal basisdata spasial tersebut akan berisi atribut id, fungsi jalan dan panjang ruas jalan. Basis data spasial menyimpan representasi dari fenomena

geografis dari dunia nyata untuk digunakan dalam SIG yang menjadi tempat penyimpanan yang memuat banyak data dan berfungsi tidak hanya sebatas menyimpan data, namun juga berbagi, pembaruan, manipulasi dan eksekusi data dengan cara yang efisien (Rolf A. Ed By; 2000). Tiga data atribut tersebut yang selanjutnya diturunkan lagi menjadi impedansi di tahap selanjutnya tentu saja dilengkapi sesuai dengan kebutuhan perangkat lunak yang digunakan.

Perangkat lunak yang digunakan adalah *PostgreSQL* untuk membangun basisdata dan terintegrasi dengan ekstensi *PostGIS* dan *pgrouting* sebagai fungsi spasialnya. *Pgrouting* cukup praktis dalam melakukan analisa jaringan karena algoritma-algoritma yang kerap diaplikasikan sudah berupa fungsi tanpa harus menuliskan kode pemrograman dengan bahasa tertentu sehingga tidak terlalu rumit dalam segi teknisnya. Adapun kode dasar atau yang digunakan untuk mengeksekusi perintah penentuan rute optimum adalah sebagai berikut;

```
SELECT FROM [algoritma](
  'SELECT gid AS id,
    source, target,
    [impedansi] AS cost,
    [impedansi_r] AS reverse_cost
  FROM [DBspasial]',
  [idA], [idB]);
```

Algoritma penentuan rute yang sudah menjadi fungsi bawaan dari *PostGIS* dipilih dan ditentukan pada [algoritma] pada baris kode di atas. 'Impedansi' dan 'impedansi_r' merupakan kolom yang berisi beban hitung yang disematkan pada tiap ruas jalan, yang membedakan dua kolom tersebut adalah arah perhitungannya. Perbedaan arah ini bisa digunakan sebagai larangan atau *restriction* untuk ruas jalan tertentu. Basisdata spasial yang digunakan sebagai wadah perhitungan dideklarasikan pada 'DBspasial' sedangkan id titik yang mewakili lokasi asal dan tujuan rute diwakilkan oleh 'idA' dan 'idB'. Baris kode tersebut di atas akan disesuaikan kembali dengan kebutuhan dari penelitian.

Algoritma yang berbeda menyelesaikan perhitungan dengan proses dan pertimbangan yang sedikit-banyak terdapat perbedaan. Rute dari satu asal ke titik tujuan yang didapat dari penentuan dan perhitungan dengan menggunakan dua algoritma yang berbeda bisa terdapat perbedaan. Perbedaan tersebut dapat meliputi perbedaan waktu yang dibutuhkan untuk menyelesaikan proses, total impedansi yang

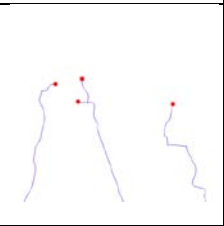
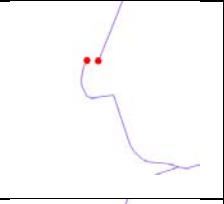

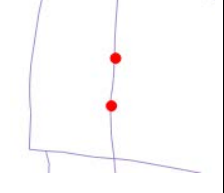
dipilih serta total ruas jalan yang terpilih. Dengan kecepatan rerata yang dipilih sebagai impedansi, maka dalam penelitian ini diperoleh perbandingan total impedansi dari setiap algoritma.

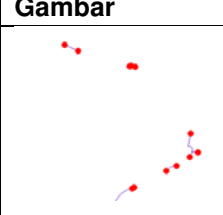
HASIL DAN PEMBAHASAN

Koreksi topologi jaringan jalan

Jaringan jalan yang direpresentasikan sebagai data spasial terbentuk dari kumpulan ruas-ruas jalan yang saling terkait satu sama lain dengan bentuk sedemikian rupa. Setiap ruas jalan memiliki pangkal dan ujung jalan yang menjadi titik temu (intersection) dari ruas jalan satu sama lain. Titik-titik tersebut direpresentasikan sebagai *node* dalam, sedangkan ruas jalannya sendiri diwakili oleh garis yang disebut sebagai *segment/link/edge*. Hubungan kedua elemen jaringan tersebut dikenal sebagai topologi. Sebagai data masukan dalam analisa jaringan topologi jaringan jalan harus terkoreksi dengan benar. Berikut di bawah ini, merupakan tabel berisi kesalahan dalam topologi jaringan jalan dan koreksi yang dilakukan;

Tabel 1. Kesalahan dan koreksi dalam topologi jaringan jalan

No	Kasus	Gambar	Koreksi
1	Ujung segmen		Dibiarkan
2	<i>Undershoot</i>		<i>Extend</i> /disambung
3	<i>Overshoot</i>		<i>Trim</i> /dipotong
4	<i>Pseudo-node</i>		<i>Merge</i> /digabung

No	Kasus	Gambar	Koreksi
5	Sisa seleksi		Dihapus

Sumber: Pengolahan data (2016)

Hampir keseluruhan sisi luar terdeteksi sebagai kesalahan karena pada node tersebut tidak memiliki koneksi dengan ruas jalan lain, namun kondisi ini tidak disertakan sebagai kesalahan topologi karena kasus ini terjadi karena ruas-ruas jalan tersebut terpotong oleh batas terluar dari daerah kajian penelitian. Kasus kedua yaitu *undershoot* merupakan kesalahan dimana antar ruas tidak terhubung sehingga terjadi *gap/celah*. Jika dilakukan analisis rute pada jaringan yang terjadi kesalahan *undershoot*, maka ruas jalan tersebut tidak dapat dilewati. Koreksi yang dilakukan adalah menyambungkan ruas jalan yang terpisah (*extend*). Kebalikan dari kasus *undershoot* adalah *overshoot*, dimana ruas jalan melebihi node simpang yang seharusnya. Kasus ini dikoreksi dengan melakukan trim atau pemotongan atau menghilangkan bagian yang berlebih. Kasus keempat adalah *pseudo-node*, dimana ruas yang seharusnya menjadi satu kesatuan terbagi menjadi beberapa bagian. Koreksi yang dilakukan untuk kasus ini adalah menggabungkan ruas-ruas tersebut menjadi satu. Terdapat juga kesalahan dimana beberapa ruas jalan yang melayang atau tidak memiliki koneksi sama sekali dengan ruas-ruas jalan di sekitarnya. Kasus ini terjadi karena sisa seleksi ruas jalan yang sudah dilakukan sebelumnya, untuk kasus ini dikoreksi dengan menghapus ruas-ruas jalan tersebut.

Jumlah total ruas dalam data jaringan jalan yang awalnya 54.241 berkurang menjadi 6455 ruas jalan. Jumlah tersebut terbagi menjadi tiga jenis atribut menurut fungsi jalannya yaitu 368 ruas jalan arteri, 676 ruas jalan kolektor dan 5411 ruas jalan. Jika dilihat dari jumlah per fungsi jalan selisih dengan data mentahnya cukup jauh berkurang. Hal ini disebabkan karena banyaknya koreksi topologi yang menyebabkan pengurangan jumlah dari data awal, yang meliputi penggabungan dan penghapusan.

Penentuan impedansi

Algoritma pencarian rute pada dasarnya memperhitungkan dan menentukan rute dengan

akumulasi beban minimal yang memungkinkan dari titik asal menuju titik tujuan. Perhitungan dan penentuan tersebut dieksekusi berdasarkan batasan tertentu berupa impedansi. Besaran impedansi ini yang dimanipulasi untuk memodelkan skenario keadaan yang terjadi di lapangan. Penelitian ini menggunakan waktu tempuh sebagai skenario impedansinya. Adapun waktu tempuh tersebut tidak bersifat dinamis, dimana besarnya diturunkan dari kecepatan rencana berdasarkan fungsi jalan menurut peraturan pemerintah nomer 34 tahun 2006 tentang jalan. Kecepatan rencana tersebut tertulis pada tabel dibawah ini;

Tabel 2. Kecepatan rencana berdasarkan fungsi jalan

Fungsi jalan	Jenis angkutan yang dilayani	Jarak perjalanan	Kecepatan rencana (km/jam)
Arteri	Angkutan utama	Jauh	60
Kolektor	Angkutan pengumpul/pembagi	Sedang	40
Lokal	Angkutan Setempat	Dekat	20

Sumber: Peraturan Pemerintah no 34 tahun 2006 tentang jalan

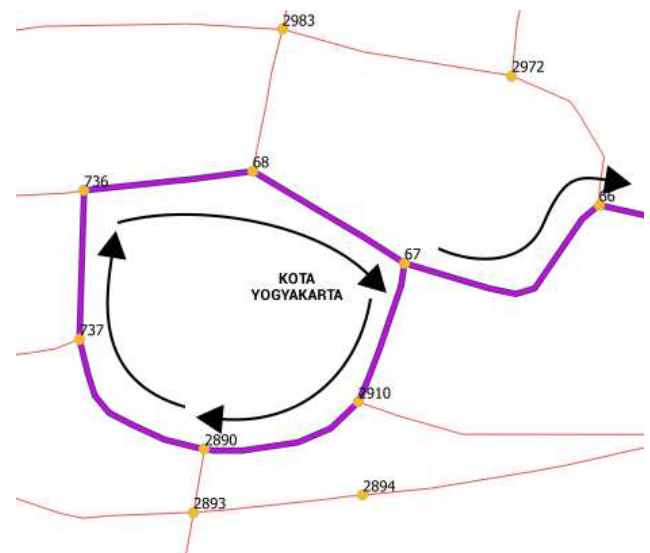
Kecepatan rencana tersebut di atas digunakan sebagai kunci untuk memanipulasi data atribut yang ada untuk mendapat data baru berupa waktu tempuh yang digunakan sebagai impedansi pada penelitian ini. Waktu tempuh dapat dihitung berdasarkan rumus kecepatan sederhana, yaitu menggunakan kecepatan rencana sebagai faktor pembagi terhadap jarak/panjang ruas jalan dengan menggunakan rumus kecepatan sebagai berikut;

$$t = s/v$$

Dimana t merupakan waktu tempuh yang dihasilkan dari jarak/panjang (s) yang dibagi dengan kecepatan rencana (v).

Penentuan rute optimum yang dilakukan pada penelitian ini mempertimbangkan arah jalan, sehingga terdapat dua beban/impedansi yang disertakan. Impedansi tersebut yaitu impedansi ($cost$) dan impedansi terbalik ($reverse cost$). Angka yang dimasukkan pada impedansi tersebut disesuaikan dengan arah jalan yang dihasilkan oleh pemrosesan topologi dari perangkat lunak *pgRouting* yaitu berdasarkan id vertek yang tercantum pada kolom atribut *source* dan *target*. Id vertek pada kolom *source* merupakan pangkal segment sedangkan ujungnya terdapat pada vertek dengan id *target*,

dengan kata lain perangkat lunak membaca arah ruas jalan tersebut searah dengan id asal ke id tujuan. Sehingga impedansi terbalik dibebankan jika arah jalan di lapangan berlawanan dengan arah yang dibaca perangkat lunak. Pada dasarnya impedansi dan impedansi terbalik ini merupakan representasi dari lamanya waktu tempuh rencana, namun besarnya akan berbeda jika ruas jalan tersebut merupakan ruas jalan searah. Lebih jelasnya diterangkan dengan ilustrasi di bawah ini;



Gambar 1. Ilustrasi pembobotan beban untuk jalan searah

Sumber: Pengolahan data, 2016

Contoh jalan searah yang digunakan dalam ilustrasi di atas berlokasi di daerah sekitar Stadion Kridosono. Rangkaian jalan yang mengelilingi stadion tersebut diberlakukan searah putaran jarum jam. Titik berwarna kuning merupakan visualisasi node dengan nomer id yang tertera. Pada kasus tersebut diatas maka beban/cost dari segmen dengan id node 67-2910, 2910-2890, 2890-737, 737-736, 736-68, 68-67 dan 67-66 disesuaikan dengan waktu tempuh rencana yang digunakan sebagai impedansi pada penelitian ini, sedangkan untuk arah kebalikan misalnya 2910-67, 67-68, 68-736 dan seterusnya diberlakukan *reverse-cost* dengan nilai maksimal yang berfungsi sebagai restriction atau halangan sehingga ruas jalan tersebut dihindari dari rute optimum yang dihasilkan.

Penyusunan Basisdata Spasial

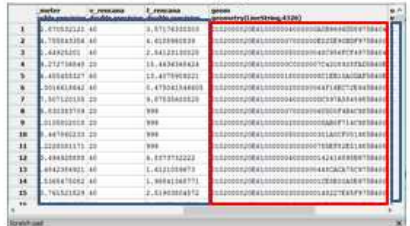
Basis data merupakan sekumpulan data/informasi yang teratur berdasarkan kriteria tertentu yang saling berhubungan. Tersimpan dalam perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya dengan

skema atau struktur tertentu (Suryantoro, 2013). Menurut sudut pandang ilmu geografi, data spasial diklasifikasikan menjadi dua jenis secara umum, yaitu data grafis yang dapat berbentuk data titik, garis maupun area dan data tabular berupa data atribut sebagai informasi pendukungnya. Semua data tersebut harus memiliki identitas khusus tertentu, bukan hanya berfungsi sebagai pembeda satu sama lain namun juga sebagai kunci hubungan dalam manipulasi dan relasi dengan data lainnya dalam satu database sehingga terjadi dependensi data.

Basisdata pada penelitian ini menggunakan *PostgreSQL* sebagai perangkat lunak DBMS yang menjadi wadah dari perhitungan algoritma untuk penentuan rute. *PostgreSQL* merupakan perangkat lunak dengan sumber terbuka (*open source*) dimana dalam proses manipulasi dan perhitungan datanya menggunakan bahasa SQL (*Structured Query Language*). Fungsi spasial pada dasarnya tidak disediakan oleh *PostgreSQL* secara langsung, karena data yang dapat diakomodasi hanya berbentuk teks dan tabel. Oleh karena itu digunakan perangkat lunak pendukung/ekstensi bernama *PostGIS* yang menyediakan dukungan terhadap data-data spasial.

Input data *shapefile* ke dalam basisdata *PostGIS* mengkonversi data tersebut menjadi data SQL, dimana pada data tersebut memiliki data grafis, data atribut maupun informasi proyeksi yang saling terkait satu sama lain. Umumnya data yang dikenal dengan data *shapefile* terdiri dari paling tidak empat jenis ekstensi data, yaitu *.shp, *.shx, *.dbf dan *.prj. Keseluruhan data tersebut dikonversi menjadi data SQL yang berbasis tabular dengan gambaran proses sebagai berikut;

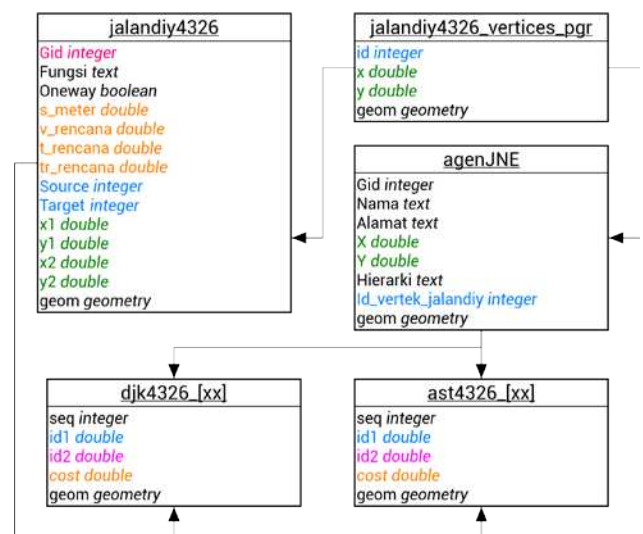
Tabel 3. Bagan konversi data *shapefile* menjadi SQL

Konversi data <i>shapefile</i> ke SQL		
input	proses	Output
jalandyi.shp jalandyi.shx jalandyi.prj jalandyi.dbf	Aplikasi <i>shapefile</i> loader & exporter	

Sumber: Pengolah data (2016)

Bagan tersebut diatas menunjukkan hubungan antara data input yang berupa *shapefile* menjadi hasil output yang berformat SQL. Data grafis yang disebut sebelumnya tersimpan pada berkas dengan ekstensi *.shp dan *.shx, dimana berkas *.shp merupakan data grafis utama dan berkas *.shx menyimpan indeks dari data grafis tersebut. Sedangkan informasi referensi geografis pada data *shapefile* tersimpan pada berkas *.prj yang berisi keterangan sistem proyeksi dan datum yang digunakan sebagai referensi spasial data tersebut. Berkas tersebut dikonversi kedalam satu kolom dengan kode unik dengan proses sedemikian rupa oleh perangkat lunak *PostgreSQL* yang pada gambar di atas ditandai dengan warna merah. Berkas *jalandyi.dbf* diubah menjadi kolom yang ditandai dengan warna biru. Berkas *.dbf menyimpan keseluruhan data atribut yang berbentuk tabular, berkas ini pada dasarnya merupakan berkas basisdata yang banyak digunakan oleh perangkat lunak lain yang berkaitan dengan pengolahan basisdata.

Basisdata spasial yang digunakan sebagai media analisis jaringan jalan pada penelitian ini menggunakan model basisdata relasional. Struktur data dalam model basisdata relasional memiliki objek-objek yang saling terhubung satu sama lain. Hubungan ini bisa dalam bentuk identitas atau atribut dari objek-objek dalam basisdata. Desain basisdata spasial digambarkan secara visual seperti pada gambar di bawah ini menggunakan *Entity Relationship Diagram* (ERD) untuk menggambarkan hubungan yang terjadi diantara objek-objek dalam basisdata;



Gambar 2. *Entity relationship Diagram* (ERD) Basisdata Spasial

Sumber: Pengolahan data (2016)

Topologi yang sudah dilakukan sebelumnya menghasilkan data *node/vertex* yang merupakan perpotongan dari setiap ruas jalan yang ada dalam keseluruhan jaringan jalan. *Node/vertex* tersebut memiliki identitas unik yang selanjutnya dicocokkan posisinya dengan posisi agen-agen JNE yang digunakan sebagai titik awal dari rute. Proses perhitungan rute optimum antara algoritma *Dijkstra* dan *A** dalam basisdata spasial tersebut memiliki perbedaan dalam kebutuhan data masukan. Pada algoritma *Dijkstra*, perhitungan dapat dilakukan dengan data identitas node, identitas ruas jalan dan beban yang dimiliki ruas jalan tersebut. Sedangkan pada algoritma *A** setiap ruas jalan yang dihitung harus diketahui koordinatnya.

Perbandingan Rute Optimum

Perhitungan rute optimum dilakukan berdasarkan sistem operasional yang diterapkan oleh jasa ekspedisi paket JNE yang diperoleh dari hasil wawancara. Berbekal data tersebut dibuat skenario perhitungan rute untuk membatasi jumlah rute yang dihitung dan memperkirakan sirkulasi paket yang terjadi dalam proses *outbond*. JNE D.I. Yogyakarta membagi daerah operasionalnya menjadi 6 kelompok *pick-up*. Skenario dibatasi untuk layanan YES (Yakin Esok Sampai). Khusus untuk layanan ini paket yang tersebar di seluruh agen harus terkumpul di gudang *outbond* sebelum jam 12.00 WIB, sehingga agen-agen dengan jadwal *pick-up* mendekati jam inilah yang berhubungan langsung dengan gudang *outbond*. Batasan tersebut membuat jumlah keseluruhan sebanyak 76 agen tadi hanya tersisa 28 agen saja.

Dua algoritma penentuan rute optimum yang digunakan pada penelitian ini memiliki perbedaan dari beberapa aspek. Jika dilihat dari prinsip kerjanya, algoritma *Dijkstra* menggunakan dasar prinsip *greedy*, dimana pada setiap iterasi mencari ruas dengan nilai impedansi minimum yang menghubungkan satu node dengan semua node disekitarnya yang belum terpilih tanpa pertimbangan apapun untuk mempercepat proses perhitungan. Salah satu kekurangan tersebut yang diperbaiki oleh algoritma *A** dengan mengedepankan efisiensi solusi optimal dengan menerapkan prinsip *best-first search* dalam perhitungannya. Fungsi heuristik melakukan perhitungan biaya perkiraan dari titik asal menuju titik tujuan. Penggunaan heuristik dalam perhitungan rute optimum bisa membuat proses hitung menjadi lebih efisien karena selain impedansi yang dihitung dari setiap ruas dalam

jaringan jalan juga ditambah dengan biaya perkiraan yang berupa arah. Semakin rendah biaya perkiraan, semakin besar kemungkinan node yang sedang dihitung mendekati titik tujuan dibandingkan simpul lain yang lebih jauh. Hal ini membuat pencapaian titik tujuan diproses dengan lebih cepat.

Perbedaan prinsip kerja dari dua algoritma tersebut juga membuat data masukan yang dibutuhkan berbeda, khususnya jika dilihat dari kebutuhan ekstensi PostGIS dalam memeksekusi kedua algoritma tersebut. Prinsip kerja algoritma *Dijkstra* yang tanpa mempertimbangkan arah melalui penggunaan heuristik membuat data yang dibutuhkan untuk menemukan rute hanya berupa data jaringan jalan dan id node titik awal dan tujuan pada data tersebut. Berbeda dengan penggunaan heuristik pada algoritma *A** yang bisa disebut melakukan perkiraan arah dari titik asal ke titik tujuan. Untuk melakukan prinsip kerja tersebut diperlukan data koordinat dari setiap node dalam jaringan jalan. Prinsip *best-first search* pada algoritma *A** dengan menggunakan heuristik untuk mempercepat proses perhitungan nyatanya dibuktikan dengan hasil dari semua skenario yang dapat dilihat pada tabel di bawah ini;

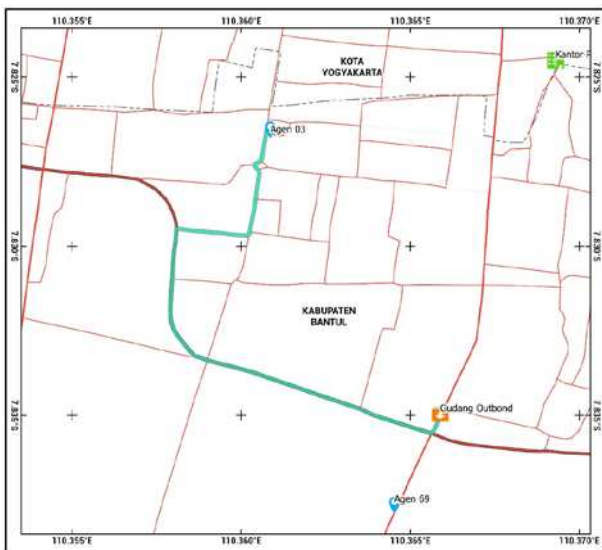
Tabel 4. Perbandingan waktu pemrosesan rute optimum dari algoritma *Dijkstra* dan *A**

No	Rute	Waktu proses <i>Dijkstra</i>	Waktu proses <i>A-star</i>
1	Agen 03	58,00	58,00
2	Agen 04	61,00	65,00
3	Agen 06	62,00	65,00
4	Agen 09	77,00	70,00
5	Agen 13	83,00	73,00
6	Agen 20	77,00	73,00
7	Agen 26	75,00	71,00
8	Agen 27	80,00	73,00
9	Agen 29	71,00	69,00
10	Agen 30	79,00	72,00
11	Agen 35	77,00	70,00
12	Agen 36	70,00	68,00
13	Agen 41	75,00	71,00
14	Agen 46	69,00	66,00
15	Agen 49	67,00	65,00
16	Agen 52	64,00	64,00
17	Agen 54	63,00	64,00
18	Agen 56	75,00	72,00
19	Agen 62	64,00	67,00
20	Agen 67	68,00	65,00
21	Agen 68	68,00	69,00
22	Agen 69	48,00	50,00
23	Kantor Cabang DIY	66,00	63,00
24	Kantor Perwakilan Condong Catur	76,00	74,00
25	Kantor Perwakilan Menukan	52,00	53,00
26	Kantor Perwakilan Randu Belang	49,00	50,00
27	Sub Agen Wates	70,00	75,00
28	Sub Agen Wonosari	74,00	76,00

Sumber: Pengolahan data (2016)

Perbedaan waktu pemrosesan yang terjadi memang hanya terpaut beberapa ms. Namun secara umum algoritma A* dapat menyelesaikan rute dengan waktu yang lebih singkat, terlebih untuk rute dengan titik asal dan tujuan yang memiliki jarak cukup jauh. Jarak yang dimaksud disini memang bukan dijadikan sebagai impedansi, namun pada jaringan jalan semakin jauh jarak yang harus dilalui berarti semakin kompleks jaringan jalan dan selanjutnya semakin rumit perhitungan yang harus dilakukan. Terlebih lagi untuk algoritma *Dijkstra* yang berarti semakin banyak itersi yang harus dilakukan. Algoritma A* lebih unggul dari segi pemrosesan dengan waktu yang bisa mencapai 13% lebih cepat dibanding algoritma *Dijkstra*.

Waktu pemrosesan dari kedua algoritma dalam menyelesaikan rute memang memiliki hasil yang berbeda. Namun rute optimum yang dihasilkan sama persis satu sama lain. Tidak ditemukan satupun skenario yang menghasilkan rute optimum yang berbeda. Berikut beberapa contoh skenario yang semuanya mewakili hasil dari algoritma *Dijkstra* dan A* yang memiliki persamaan. Hasil penentuan rute optimum untuk skenario dari agen 03 menuju gudang *outbond* JNE memiliki Impedansi total 196,65 detik atau 3,275 menit. Berikut di bawah ini visualisasi dari rute optimum tersebut;

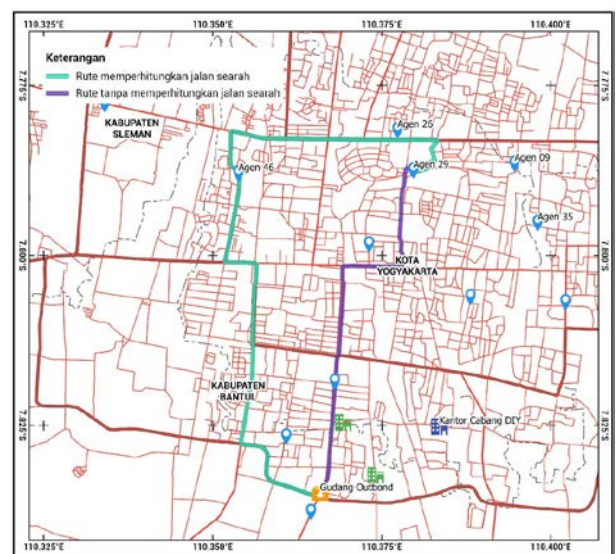


Gambar 3. Peta rute optimum sirkulasi paket JNE dari agen 03 hasil algoritma *Dijkstra*
Sumber: Pengolahan data, 2016

Rute optimum yang ditunjukkan dengan garis berwarna biru muda pada gambar di atas. Rute tersebut bagi pengemudi yang memiliki pemahaman lokal mungkin akan terlihat sedikit

janggal, dimana belokan ke kanan pertama justru terkesan menambah jarak perjalanan. Hal ini terjadi karena impedansi yang digunakan pada penelitian ini merupakan waktu tempuh rencana yang didasarkan pada fungsi jalan, dimana semakin tinggi fungsi jalan maka semakin pendek waktu tempuh rencananya, sehingga rute akan cenderung mencari fungsi jalan yang lebih tinggi. Pada skenario tersebut diatas, ruas rute mendekat pada jalan lingkaran padahal pada posisi tersebut akan lebih efisien jika tetap mengambil jalan lurus. Terlebih lagi pada rute tersebut akan menambah waktu tempuh karena semakin banyak persimpangan dengan lampu lalu lintas yang dilewati. Bertambahnya persimpangan dengan lampu lalu lintas tentu akan memungkinkan bertambahnya waktu tempuh yang disebabkan oleh waktu tunggu pada tiap persimpangan tersebut.

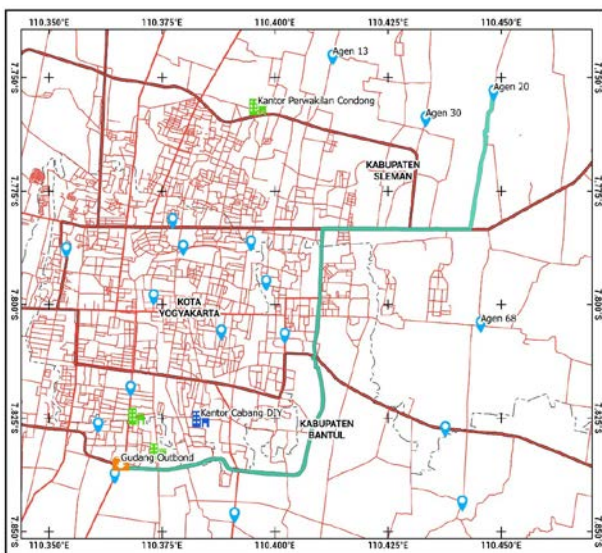
Contoh skenario perhitungan rute selanjutnya diawali dari agen 29 yang berada di Jl. Kusbini no.10 Klitren Yogyakarta. Skenario ini merupakan salah satu skenario rute yang melewati dan terhalang oleh jalan searah. Adapun jalan searah tersebut berada pada ruas Jalan Suryopranoto di kelurahan Semaki, Umbulharjo. Waktu tempuh yang dibutuhkan dari skenario tersebut berdasarkan rute optimum yang dihasilkan adalah 600,8 detik atau kurang lebih 10 menit. Apabila halangan berupa jalan searah tadi tidak dihiraukan akan menghasilkan rute yang divisualisasikan dengan garis berwarna ungu pada gambar dibawah ini;



Gambar 4. Peta rute optimum sirkulasi paket JNE dari agen 26 hasil algoritma *Dijkstra*
Sumber: Pengolahan data, 2016

Skenario ini membuktikan bahwa algoritma *Dijkstra* berhasil menyelesaikan perhitungan rute yang memiliki halangan dengan mencari ruas jalan lain untuk menghasilkan rute optimum yang dapat dilewati. Ruas jalan yang ditempuh untuk menghindari Jl. Suryopranoto adalah ruas Jl. Urip Sumoharjo sehingga rute yang ditempuh sedikit memutar dengan mengambil arah ke utara terlebih dahulu. Hasil ini disebabkan oleh impedansi pada ruas Jl. Urip Sumoharjo yang lebih rendah karena fungsi jalannya lebih tinggi yaitu sebagai jalan arteri.

Skenario perhitungan rute lainnya termasuk salah satu skenario dengan jarak terjauh dari titik tujuan. Rute diawali dari agen 20 yang beralamat di Ruko R.C2 Purwomartani, Kecamatan Kalasan. Waktu tempuh yang dibutuhkan untuk mencapai titik tujuan adalah 1494,314 detik atau kurang lebih 24,9 menit. Berikut di bawah ini visualisasi skenario rute tersebut;



Gambar 5. Peta rute optimum sirkulasi paket JNE dari agen 20 hasil algoritma *Dijkstra*
Sumber: Pengolahan data, 2016

Rute yang ditunjukkan dengan garis berwarna biru muda pada gambar di atas menunjukkan ruas-ruas jalan yang diambil dari agen 20 menuju gudang *outbond* sebagai rute optimum. Rute tersebut melintasi pinggiran kawasan perkotaan Yogyakarta tidak seperti dua rute sebelumnya yang harus menembus jalan perkotaan. Dilihat dari waktu pemrosesannya, skenario ini memiliki selisih waktu yang tidak terpaut jauh dibanding dengan skenario rute agen 29 yang hanya terpaut sekitar 6 ms. Hasil ini disebabkan karena rute ini didominasi oleh ruas-

ruas jalan dengan fungsi arteri dengan impedansi yang rendah, sehingga algoritma menghitung secara lebih sederhana, sebagaimana kecenderungan rute akan selalu memilih ruas dengan impedansi yang rendah dalam rangkaian rute optimum yang akan dipilih.

Persamaan tersebut lebih jelas terlihat pada tabel di bawah ini. Jumlah segmen merupakan banyaknya ruas jalan yang terpilih untuk dijadikan sebagai rangkaian rute optimum. Dari kedua algoritma yang dipakai jumlah segmen rute optimum untuk semua skenario sama jumlahnya.

Tabel 5. Perbandingan jumlah *sequence* rute optimum dan total impedansi dari algoritma *Dijkstra* dan *A**

No	Rute	<i>Dijkstra</i>		<i>A-star</i>	
		Jumlah <i>sequence</i>	Total impedansi	Jumlah <i>sequence</i>	Total impedansi
1	Agen 03	17	196,650	17	196,650
2	Agen 04	47	606,473	47	606,473
3	Agen 06	53	489,376	53	489,376
4	Agen 09	40	831,727	40	831,727
5	Agen 13	45	1383,525	45	1383,525
6	Agen 20	42	1494,314	42	1494,314
7	Agen 26	74	517,864	74	517,864
8	Agen 27	44	1454,209	44	1454,209
9	Agen 29	78	600,800	78	600,800
10	Agen 30	42	1054,239	42	1054,239
11	Agen 35	65	602,163	65	602,163
12	Agen 36	21	694,140	21	694,140
13	Agen 41	35	670,517	35	670,517
14	Agen 46	50	462,686	50	462,686
15	Agen 49	14	291,199	14	291,199
16	Agen 52	11	689,497	11	689,497
17	Agen 54	37	442,782	37	442,782
18	Agen 56	36	905,824	36	905,824
19	Agen 62	24	1046,804	24	1046,804
20	Agen 67	19	170,630	19	170,630
21	Agen 68	25	1232,816	25	1232,816
22	Agen 69	4	33,827	4	33,827
23	Kantor Caba	19	425,898	19	425,898
24	Kantor Perw	94	670,443	94	670,443
25	Kantor Perw	10	142,754	10	142,754
26	Kantor Perw	8	147,108	8	147,108
27	Sub Agen W	59	1840,453	59	1840,453
28	Sub Agen W	50	2328,510	50	2328,510

Sumber: Pengolahan data (2016)

Persamaan ruas yang menjadi penyusun dalam rangkaian rute optimum yang sama jelas berarti total impedansi dari setiap skenario juga sama karena total impedansi merupakan hasil penjumlahan dari setiap beban dari ruas jalan yang terpilih.

Data tersebut di atas menunjukkan bahwa hasil perhitungan dari algoritma *Dijkstra* dan *A** dari skenario dengan impedansi yang sama memberikan hasil rute optimum yang identik.

Perbedaan terletak pada kecepatan pemrosesan/perhitungan rutenya. Dimana algoritma A* menghabiskan waktu proses yang relatif lebih singkat jika dibandingkan waktu proses dari algoritma *Dijkstra*. Perbedaan kecepatan pemrosesan algoritma ini disebabkan oleh perbedaan sistem kerja seperti yang disinggung sebelumnya. Algoritma A* menggunakan heuristik untuk meningkatkan efisiensi hitung dengan membuat perkiraan dalam rangkaian penyelesaian masalah. Perbedaan kecepatan ini bisa dikatakan karena perbedaan jumlah node yang disertakan dalam perhitungan untuk menghasilkan rute optimum.

KESIMPULAN

Berdasarkan hasil dan pembahasan penelitian yang dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Algoritma *Dijkstra* dan A* mampu menghasilkan rute optimum untuk sebagian tahapan dalam sirkulasi paket JNE D.I Yogyakarta. Kedua algoritma tersebut juga mampu menyelesaikan skenario rute yang terhalang oleh restriction berupa ruas-ruas jalan searah. Meskipun dalam beberapa skenario rute yang dihasilkan kurang tepat karena rute memiliki kecenderungan mencari ruas jalan dengan fungsi tertinggi sebagai ruas dengan impedansi rendah yang dimungkinkan terlebih dahulu, sehingga dalam beberapa kasus rute yang dihasilkan mengambil jalan yang memutar dan menambah waktu tempuh jika dilakukan di lapangan.
2. Penggunaan impedansi yang sama pada skenario penentuan rute optimum pada penelitian ini membuat algoritma *Dijkstra* dan A* menghasilkan rute yang sama. Perbedaan terletak pada kebutuhan data masukan dan sisi pemrosesan. Algoritma *Dijkstra* tidak memerlukan data koordinat dari setiap *node* dalam jaringan jalan seperti algoritma A*. Koordinat *node* digunakan oleh algoritma A* untuk memperkirakan arah menuju titik tujuan sesuai dengan prinsip heuristik yang digunakan algoritma tersebut untuk meningkatkan efisiensi perhitungan yang selanjutnya berimbas pada kebutuhan waktu proses algoritma A* yang lebih singkat dibandingkan algoritma *Dijkstra*.

DAFTAR PUSTAKA

- Chang, kang-tsung. 2006. *Introduction do Geographic Information Systems*. New York: McGraw-Hill.
- By, Rolf A. De etall. 2000. *Principles of Geographic Information Systems*. Enschede, Belanda: ITC.
- Hoel, Lester A. 2011. *Transportation Infrastructure Engineering: A Multimodal Integration*. Charlottesville: University of Virginia.
- Maantay, Juliana. Ziegler. 2006. *GIS for urban environment*. California: ESRI PRESS
- Merrifield, Timothy. 2010. *Heuristic Route Search in Public Transportation Networks*. Thesis. Chicago: University of Illinois.
- Suryantoro, Agus. 2013. *Integrasi Aplikasi Sistem Informasi Geografis*. Yogyakarta: Penerbit Ombak.
- Yousman, Yeyep. 2004. *Sistem Informasi Geografis dengan ArcView3.3 Professional* Yogyakarta: Andi Offset