OPTIMALISASI ALGORITMA DIJKSTRA DALAM MENGHADAPI PERBEDAAN BOBOT JALUR PADA WAKTU YANG BERBEDA

Isnaeni Setiyadi[1)], Teguh Bharata Adji[2)], Noor Akhmad Setiawan[3)]

[1,2,3)] Jurusan Teknik Elektro dan Teknologi Informasi Universitas Gadjah Mada Yogyakarta
Jalan Grafika No.2 Yogyakarta
Email : isnaeni_s2te12@mail.ugm.ac.id[1)], adji@mti.ugm.ac.id[2)], noorwewe@ugm.ac.id [3)]

Abstrak

Informasi sangat dibutuhkan dalam dunia industri. Salah satu informasi yang penting adalah informasi rute kendaraan untuk distribusi barang. Algoritma yang digunakan salah satunya adalah algoritma dijkstra. Pemasalahan pada algoritma dijstra adalah pemanfaatan bobot pada masing-masing jalur yang masih bersifat statik. Sedangkan situasi kepadatan lalu lintas di jalan raya dapat berubah sewaktu-waktu.

Penelitian ini akan melakukan perbaikan algoritma dijkstra dengan menerapkan bobot yang berubah-ubah. Penelitian dilakukan pada rute jalan di Kota Palu Provinsi Sulawesi Tengah. Metode yang digunakan adalah metode kuantitatif dengan menggunakan data dari Dinas Pekerjaan Umum Kota Palu. Data tersebut berupa jenis atau kelas jalan dan jumlah kendaraan per jam. Kemudian data tersebut diolah menggunakan manual MKJI 1997. Dari hasil pengolah data, diperoleh variabel bobot waktu berupa waktu tempuh. Setelah metode ditemukan, variabel tersebut digunakan bersama dengan metode untuk diuji coba dan hasilnya dibandingkan antara metode dijkstra konvensional dengan metode yang diusulkan.

Hasil dari penelitian ini membuktikan bahwa metode yang diusulkan bisa diterapkan dengan algoritma dijktra konvensional, dan terbukti mampu mengatasi perubahan bobot(kepadatan lalu lintas).

Kata kunci: *Vehicle Routing Problem, Shortest Path Problem, Algoritma Dijkstra, Bobot Dinamik, Time Window.*

1. Pendahuluan

a. Latar Belakang

Teknologi informasi telah menjadi kebutuhan manusia di dunia. Kehadirannya menjadi solusi bagi manusia untuk memperoleh informasi yang dibutuhkannya. Dengan teknologi, informasi yang dibutuhkan dapat diperoleh sesuai waktu yang dibutuhkan dan sesuai kebutuhan pengguna informasi.

Dalam dunia industri, salah satu informasi yang dibutuhkan adalah informasi rute dalam distribusi barang. Permasalahan distribusi merupakan salah satu faktor yang penting dalam mempengaruhi peningkatan pendapatan. Berdasarkan penelitian para ahli, menyatakan bahwa biaya distribusi rata-rata sebesar 16% dari harga jual barang yang dihasilkan. Ini berarti bahwa perlu adanya metode yang digunakan untuk mengurangi biaya distribusi barang.

Vehicle Routing problem (VRP) merupakan salah satu jenis permasalahan pencarian rute terdekat. Pemanfaatannya di dalam dunia industri adalah pendistribusian barang. Salah satu *algoritma* dalam *VRP* adalah *Algorima Dijkstra*. Permasalahan pada *algoritma* ini adalah bobot yang *statik*. Bila diterapkan dalam VRP, bobot pada *algoritma* ini tidak berubah, padahal situasi kepadatan jalan selalu berubah-ubah setiap waktu[1].

Salah satu pengembangan dari *VRP* adalah *Dynamic Vehicle Routing Problem with Time Window (DVRPTW)*[2]. *DVRPTW* merupakan permasalahan *VRP* yang tidak hanya mengatasi kendala kapasitas jalan saja, tapi juga kendala yang mengharuskan kendaraan melayani konsumen pada *time frame* tertentu. Dimana *time trame* tersebut merupakan *interval* waktu yang diperlukan untuk dilakukannya suatu perjalanan yang satu ke perjalanan berikutnya. Interval waktu tersebut bisa juga merupakan waktu tunggu akan dilakukannya perjalanan dan waktu yang diperlukan untuk melakukan pelayanan di suatu toko atau agen.

Dengan Adanya *time window,* rute berikutnya akan mulai ditempuh setelah *time window* tertentu dilewati. Hal ini mengakibatkan perubahan bobot jalur yang akan dilalui.

Telah banyak penelitian yang dilakukan untuk mengatasi permasalahan *VRP*. Perbedaan penelitian ini dengan penelitian yang sudah ada adalah penggunaan *DVRPTW* pada algoritma Djisktra.

Bila pada penelitian yang sudah ada, *DVRPTW* diterapkan pada interval waktu tunggu pelayanan atau waktu istirahat, maka pada penelitian ini *DVRPTW* akan dimanfaatkan untuk mengatasi perubahan kepadatan jalur pada jam-jam tertentu. Hal ini diprediksi sejak awal sebelum dilakukannya perjalanan. Sehingga bisa diperoleh jalur yang optimal meskipun menghadapi perubahan bobot jalur pada jam-jam berbeda.

b. Rumusan Masalah

Dari latar belakang di atas dapat diambil rumusan masalah yaitu keterbatasan algoritma Dijsktra dalam menghadapi bobot (kepadatan lalu lintas) yang berubah-ubah.

c. Tujuan Penelitian

Tujuan dalam penelitian ini adalah mengatasi keterbatasan algorithma dijkstra dalam menghadapi bobot (kepadatan lalu lintas) yang berubah-ubah.

d. Metodologi Penelitian

Penelitian dilakukan di Kota Palu Provinsi Sulawesi Tengah dengan menggunakan *metode kuantitatif*. Data pada penelitian ini berupa kapasitas jalan yang terdiri dari jenis atau kelas jalan raya, jumlah kendaraan dan jenis kendaraan pada jam-jam tertentu. Berdasarkan data tersebut akan diolah sehingga diperoleh bobot dalam bentuk waktu tempuh pada masing-masing ruas jalan.

Dari data bobot dalam bentuk waktu tempuh tersebut, akan dirancang metode yang menggabungkan antara metode konvensional *(Dijkstra)* dengan metode *heuristik (DVPRTW)*. Setelah metode ditemukan, akan dilakukan uji coba dan dibandingkan antara hasil uji coba metode konvensional Dijkstra dengan hasil uji coba metode yang diusulkan

e. Tinjauan Pustaka

Bing Liu (1994) memodifikasi *algoritma dijkstra* dengan menggunakan 3 tahapan[3]; yaitu :

1. Menggunakan algoritma *Case Base Reasoning (CBR)* untuk mengambil rute yang sudah pernah dilalui dan menggunakannya sebagai rute terbaik.
2. Jika menggunakan langkah 1 tidak diperoleh rute terbaik, maka akan digunakan *algoritma Neighborhood*, yaitu menggunakan *algorithma CBR* namun titik tujuan tidak sama persis. Titik tujuan adalah titik yang sudah ada dan tersimpan pada *Case Base Reasoning*.
3. Jika dengan cara pertama dan cara kedua juga tidak ditemukan rute terbaik, maka langkah ketiga adalah menggunakan *algoritma Dijkstra* yang standar, sampai diperoleh rute yang terbaik.

Dong Zhang (2010) melakukan penelitian pada *algoritma Dijsktra* dengan menambah kepadatan lalu lintas sebagai *atribut*nya[4]. Penambahan *atribut* tersebut berguna untuk meningkatkan algoritma dijkstra dalam *aplikasi GIS* (*Geographic Information System)*. Hasil penelitian ini adalah bobot *algoritma dijkstra* pada aplikasi GIS akan ditambahkan bobot kepadatan jalan.

Much Aziz Muslim (2005) melakukan penelitian terhadap algoritm dijkstra dengan bobot lebih dari satu. Yaitu bobot jarak, bobot waktu dan bobot biaya. Semua bobot bersifat statik namun pada bobot waktu, dibagi menjadi 3 kategori yaitu *waktu sibuk, waktu normal dan waktu sepi.* Ketiga kategori tersebut bersifat statik, tidak berubah pada waktu yang berbeda.

Pada tahun 2013, Ali mukhsinin dan kawan-kawan melakukan penelitian tentang *VRP* dengan objek penelitian di CV.IFFA[2]. CV. IFFA merupakan distributor Sanitary yang berada di Jawa Barat. Perusahaan ini melayani distribusi barang di tiga Kabupaten, yaitu Kabupaten Karawang, Kabupaten Purwakarta dan Kabupaten Subang. Pada penelitian ini, dilakukan penerapan *DVRPTW* pada rentan waktu tunggu pelayanan toko atau langganan.

f. Landasan Teori

1. **Algoritma**

Menurut K.R. Rao (2010) Algoritma merupakan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis terhadap permasalahan yang akan diselesaikan[5][6]. Dalam bidang komputer, algoritma sangat diperlukan dalam menyelesaikan berbagai masalah pemrograman, terutama dalam proses komputasi numerik. Tanpa algoritma yang dirancang dengan baik maka proses pemrograman akan menjadi salah, rusak atau lambat dan tidek efisien.

2. **Algoritma Pencarian Rute Terpendek**

Menurut Mutakhiroh (2010) dalam [7], secara umum, pencarian jalur terpendek dibagi menjadi dua metode, yaitu metode konvensional dan metode *heuristik*. Metode konvensional cenderung mudah dipahami dari pada metode heuristik. Tetapi bila dibandingkan, hasil yang diperoleh dari metode heuristik lebih variatif dan waktu yang diperlukan lebih singkat.

Metode konvensional, berupa algoritma yang menggunakan perhitungan matematis biasa. Ada beberapa metode yang biasa digunakan untuk melakukan pencarian rute terpendek, diantaranya *algoritma Dijkstra, algoritma Floyd-Warshal*, *algoritma Bellman-Ford*, dan lain-lain.

Metode *Heuristik* adalah sub bidang kecerdasan buatan yang digunakan untuk melakukan pencarian dan penentuan rute terpendek. Ada beberapa algoritma pada metode *heuristik* yang biasanya digunakan di antaranya adalah *Algoritma Semut (Ant Colony)* dan *Algoritma Genetika (Genetic Algorithm)*.

3. **Algoritma Dijkstra**

Algoritma Dijkstra adalah salah satu metode yang digunakan untuk memecahkan permasalahan pencarian rute terpendek. Istilah yang sering digunakan adalah *Shortest Path Problem (SPP)* atau *Vehicle Routing Problem (VRP)*. Algoritma ini termasuk dalam algoritma konvensional. Salah satu pemanfaatannya adalah pada sebuah aplikasi pencarian rute terdekat atau *aplikasi navigator* pada suatu daerah.

Algoritma ini pertama kali dikemukakan oleh *Edger W. Dijkstra (1959)* dan telah secara luas digunakan dalam

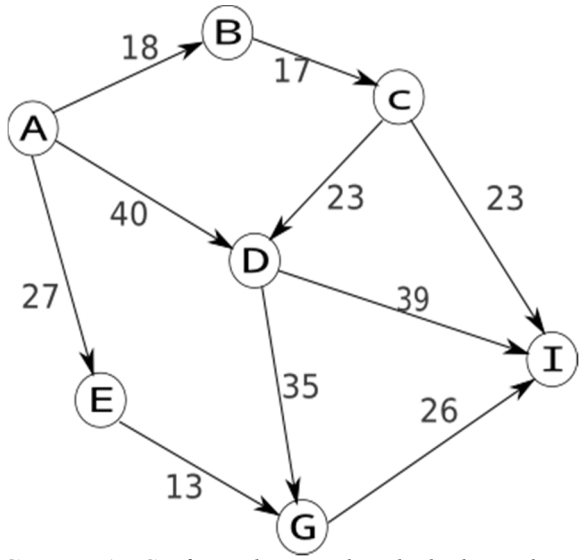
menentukan rute terpendek berdasarkan kriteria tertentu sebagai parameternya. Parameter tersebut bisa jarak antara *titik,* waktu tempuh, dan biaya yang dibutuhkan untuk menempuh rute tersebut[8].

4. MKJI 1997

MKJI merupakan singkatan dari *Manual Kapasitas Jalan Indonesia*. MKJI 1997 merupakan manual yang dikeluarkan pada tahun 1997, dan menggantikan manual sementara untuk fasilitas lalu lintas perkotaan (Januari 1993) dan lalu lintas luar perkotaan (Agustus 1994) yang telah dikeluarkan sebelumnya.
MKJI dapat digunakan untuk menganalisa rute atau jaringan jalan pada suatu kawasan perkotaan, yaitu dengan penerapan yang sesuai dengan *tipe fasilitas* lalu-lintasnya. Kemudian waktu tempuh total dapat diperoleh sebagai jumlah waktu tempuh dan tundaan pada setiap *ruas jalan* dan *titik simpul* sepanjang rute yang dipelajari/diteliti[9].

2. PEMBAHASAN

a. **Menentukan Graph**
Graph merupakan *visualisasi* rute yang dibutuhkan untuk mencoba metode yang diusulkan. Rute tersebut diambil di Kota Palu. Titik-titik pada objek penelitian yang sesungguhnya akan digantikan menjadi huruf dengan tujuan untuk memudahkan pemahaman. Rute tersebut merupakan rute dari *Kecamatan Palu Barat* menuju *Bandara Mutiara Sisaljufri* yang berada di *Kecamatan Palu Utara*. Antara Kecamatan Palu Barat dan Kecamatan Palu Utara, dipisahkan oleh *Sungai Gumbasa*. Untuk menempuh rute tersebut, ada 3 (tiga) alternatif yang bisa dilalui, yaitu melalui Jembatan 1, Jembatan 2 dan Jembatan 3. Jembatan 1, 2 dan 3 adalah nama jembatan yang ada di Kota Palu. Adapun rute tersebut dapat dilihat pada gambar 1.



Gambar 1. *Graf visualisasi titik-titik objek penelitian*

b. **Pembobotan Jalur (Waktu Tempuh)**
Bobot pada setiap *ruas jalan* akan ditampilkan dalam bentuk *waktu tempuh rata-rata* sebuah kendaraan untuk melewati jalur tertentu. Untuk memperoleh waktu tempuh kendaraan pada ruas jalan, menggunakan rumus 1 [10] :

$$TT = \frac{L}{V} \quad \text{..... (1)}$$

Dimana :
TT = waktu tempuh rata-rata kendaraan (jam)
L = panjang penggal jalan (km;m)
V = Kecepatan rata-rata (km/jam; m/dt)

Dari rumus tersebut, dibutuhkan data panjang penggal jalan yang diperoleh dari Dinas Pekerjaan Umum provinsi Sulawesi Tengah dan atau Dinas Pekerjaan Umum Kota Palu.
Sedangkan data kecepatan rata-rata kendaraan dapat dihitung menggunakan rumus dari MKJI 1997 [10]:

$$FV = (FV_0 - FV_W) + FFVsf \times FFVcs \quad \text{..... (2)}$$

Dimana :
FV = Kecepatan arus bebas(km/jam)
FV_0 = Kecepatan arus bebas dasar (km/jam)
FVw = Penyesuaian kecepatan akibat lebar jalur (km/jam)
FFVsf = Penyesuaian hambatan samping dan lebar bahu jalan
FFVcs = Penyesuaian ukuran kota

FV_0 merupakan *kecepatan arus bebas dasar*. Dalam MKJI 1997 kecepatan arus bebas dasar dapat dilihat pada *tabel 1*.

Tabel 1. *Kecepatan arus bebas dasar*

Jenis Jalan	FV_0 (km/jam)			
	MC	HV	LV	Rata[2]
6 lajur terbagi	48	52	61	57
4 lajur terbagi	47	50	57	55
4 lajur tak terbagi	43	46	53	51
2 lajur tak terbagi	40	40	44	42

Sumber : MKJI 1997[9]

FV_w merupakan penyesuaian kecepatan arus bebas untuk lebar jalur lalu lintas. Dalam MKJI 1997 penyesuaian ini dapat dilihat pada *tabel 2* :

Tabel 2. *Penyesuaian arus bebas lebar jalur*

Lebar Jalan (m)	5	6	7	8	9	10	11
FV_W (km/jam)	-9,5	-3	0	3	4	6	7

Sumber : MKJI 1997[9]

FFVsf adalah merupakan penyesuaian pengaruh hambatan samping dan lebar bahu jalan ke penghalang yang ditentukan dalam MKJI 1997 seperti pada *Tabel 3*.

Tabel 3 : *Penyesuaian hambatan samping dan lebar bahu*

Tipe Jalan	Kelas Hambatan Samping	Faktor Penyesauain Hambatan Samping dan Lebar Bahu			
		Lebar Bahu Efektif Rata-rata Ws (meter)			
		≤ 0,5	1,0	1,5	> 2,0
2/2 UD atau jalan satu arah	VL	0,94	0,96	0,90	1,01
	L	0,92	0,94	0,97	1,00
	M	0,89	0,92	0,95	0,98
	H	0,82	0,86	0,90	0,95
	VH	0,73	0,79	0,85	0,91

Sumber : MKJI 1997[9]

FFVcs adalah faktor penyesuaian kapasitas untuk ukuran kota yang ditentukan dalam MKJI 1997 dapat dilihat pada *Tabel 4* .

Tabel 4. *Penyesuaian ukuran kota*

No	Ukuran Kota (Juta Penduduk)	Faktor Penyesuaian Ukuran Kota
1	< 0,1	0,86
2	0,1 – 0,5	0,90
3	0,5 – 1,0	0,94
4	1,0 – 3,0	1,00
5	> 3,0	1,04

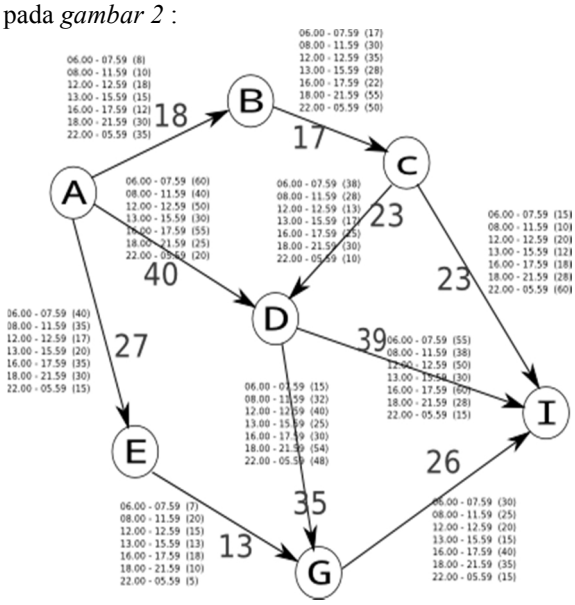
Sumber : MKJI 1997[9]

Setelah data diolah menggunakan manual pada MKJI, maka diperoleh daftar waktu tempuh yang bervariasi. Data tersebut dapat dilihat pada *Tabel 5*.

Tabel 5. *Daftar waktu tempuh masing-masing jalur*

No	Waktu	Waktu Tempuh pada masing-masing jalur (Menit)									
		AB	AD	AE	EG	CD	DG	BC	GI	CI	DI
1	06.00 – 07.59	8	60	40	7	38	5	17	30	15	55
2	08.00 – 11.59	10	40	35	20	28	32	30	25	10	38
3	12.00 – 12.59	18	50	17	15	13	40	35	20	20	50
4	13.00 – 15.59	15	30	20	13	17	25	28	15	12	30
5	16.00 – 17.59	12	55	35	18	25	30	22	40	18	60
6	18.00 – 21.59	30	25	30	10	30	54	55	35	28	28
7	22.00 – 05.59	35	20	15	5	10	48	50	15	60	15

Bila data pada tabel 5 dimasukkan pada gambar 1, maka akan diperoleh jalur dan data seperti tampak pada *gambar 2* :



Gambar 2 :
Graf dan waktu tempuh masing-masing jalur

c. **Algoritma Dijkstra Standar**

Algoritma Dijkstra, (dinamai menurut penemunya, seorang ilmuwan komputer, *Edsger Dijkstra*), adalah sebuah algoritma yang dipakai dalam memecahkan permasalahan *rute* terpendek (*shortest path problem*) untuk sebuah *graf* berarah (*directed graph*) dengan bobot-bobot sisi (*edge weights*) yang bernilai *non-negatif*. Contoh *graf* dapat dilihat pada gambar 1.

Algoritma Dijsktra bisa ditulis menggunakan model matriks [11]. Penjelasannya adalah sebagai berikut: bila diasumsikan sebuah graf memiliki titik sebanyak *N* titik. *Variabel cost[i,j]* adalah bobot dari titik *i* ke titik *j*. V_s adalah titik mulai sebuah *jalur*. *Jalur* adalah satu arah langsung dari titik sumber ke titik tujuan. Pada beberapa penelitian, jalur ini disebut *edge*. Variabel *dist[i] merepresentasikan* titik mula V_s *ke setiap titik tujuan* V_j's dengan bobot paling kecil. Karena itu, bentuk matematikanya menjadi :

$Dist[i] = Cost[s,i],\ V_i \in V$

Dimana *V* adalah kumpulan semua titik. Yaitu sejumlah *N* titik. Dengan menganggap *S* adalah himpunan dari rute tercepat yang sudah ditemukan dari titik awal ke semua titik tujuan, maka nilai ini dikenali dengan $S = (V_s)$, maka rute tercepat pada semua titik awal V_s ke semua titik pada *graf* adalah :

$Dist[i] = Cost[s,i],\ V_i \in V$

Langkah 1 : Pilih titik *Vj*, dan

$$Dist[j] = min\{Dist[i] \mid V_i \in V - S\}$$

V_j adalah titik yang terdekat dengan i. Yang diperoleh dari himpunan S yaitu pada S_j.

Langkah 2 :

Jika $Dist[j] + Cost[j,k] < Dist[k]$ maka rute tercepat menuju k harus di update dengan nilai *Dist[j]*.

Langkah 3 :
Update variabel *Dist[k]* menjadi
Dist[k] = Dist[j]+Cost[j,k]
Ulangi langkah 2 dan 3 sampai semua titik yang ada pada *graf* telah dikunjungi.

d. **Permasalahan Algoritma Dijkstra Standar**

Algoritma dijkstra bekerja dengan bobot yang *statik* (*bobot tidak pernah berubah)*. Bila diterapkan pada *VRP* maka bobot yang bersifat statik tersebut akan sangat berpengaruh. Karena situasi kepadatan lalu lintas di setiap ruas jalan pada suatu waktu, berbeda dengan waktu yang lain. Jadi, bobot pada situasi jalan yang sebenarnya, haruslah *dinamik (tidak tetap).*

e. **Metode Yang Diusulkan**

Metode yang diusulkan dalam penelitian ini merupakan perbaikan terhadap kinerja *algoritma Dijkstra* pada sisi *pembobotan*. Pembobotan pada algoritma standar yang bersifat statik akan diubah menjadi *dinamik*. Ke-*dinamik*-an tersebut timbul disebabkan oleh kepadatan lalu lintas yang berbeda pada setiap waktu. Perbedaan waktu yang diteliti dapat dilihat pada *gambar 2*.

Untuk mempermudah pemahaman terhadap metode yang diusulkan, maka akan dibuat *model matematika* terhadap metode *standar dikjstra*. Model matematika yang akan dibuat dibatasi pada penambahan bobot saja. Yaitu sebagai berikut :

$$Dist[i,j] = Dist[i] + Dist[j] \quad \text{..... (3)}$$

Dimana :
Dist[i,j] adalah bobot dari *i* ke *j*.
Dist[i] adalah bobot sampai pada titik *i*
Dist[j] adalah bobot ke titik j

Sedangkan metode yang diusulkan adalah sebagai berikut :

$$Dist[i,j] = Dist_n[i] + Dist_n[j] \quad \text{bila } n \le ny$$
$$Dist[i,j] = Dist_n[i] + Dist_{n+1}[j] \quad \text{bila } n > ny$$
Bila n+1 > 7 maka n = 1

..... (4)

Dimana :
n adalah urutan pada variasi waktu tabel 1
ny adalah waktu akhir pada *n* tabel 1

f. **Pengujian Metode**

Rumus 4 merupakan metode yang diusulkan dalam penelitian ini. Perbedaan antara rumus 3 dan rumus 4 adalah perbedaan bobot waktu pada masing-masing *iterasi*. Pengertian *Iterasi* adalah proses dilakukannya pencarian rute dari titik sumber ke titik tujuan.
Untuk membuktikan *keefektifan* rumus yang diusulkan, maka harus dilakukan uji coba terhadap kedua rumus tersebut, kemudian dibandingkan hasilnya dan dianalisis sehingga diperoleh hasil penelitian.
Untuk melakukan uji coba terhadap metode, akan diambil 1 sumber dan 1 tujuan yang sama namun dilakukan pada 2 perbedaan waktu dimulainya iterasi. Titik sumber adalah titik ***A*** dan tujuannya adalah titik ***I***. Adapun data yang dipergunakan untuk uji coba dapat dilihat pada tabel 6.

Tabel 6. *Data uji coba rumus 3 dan rumus 4*

No	Rumus	Jam dimulainya Iterasi
1	Rumus 3	07.00
2	Rumus 3	22.00
3	Rumus 4	07.00
4	Rumus 4	22.00

Dari hasil pengujian tersebut, diperoleh hasil sebagai berikut :

1. Waktu yang dibutuhkan untuk menempuh rute dari titik ***A*** sampai ke titik ***I*** yang dimulai pada jam ***07.00*** dan dimulai pada jam ***22.00*** menggunakan *metode standar dijkstra (rumus1)* adalah sama. Yaitu :
   - Pada jam *07.00* diperoleh hasil rute terbaik ***ABCI*** dengan total waktu tempuh yang dibutuhkan ***2 jam 25 menit***.
   - Pada jam *22.00* diperoleh hasil rute terbaik yang sama yaitu ***ABCI*** dengan total waktu tempuh juga sama yaitu ***2 jam 25 menit***.

2. Sedangkan waktu yang dibutuhkan untuk menempuh rute dari titik ***A*** sampai ke titik ***I*** yang dimulai pada jam ***07.00*** dan jam ***22.00*** menggunakan metode yang diusulkan adalah sebagai berikut :
   - Pada jam *07.00* diperoleh hasil rute terbaik ***ABCI*** dengan total waktu tempuh yang dibutuhkan ***40 menit***.
   - Pada jam *22.00* diperoleh hasil rute terbaik ***ADI*** dengan total waktu tempuh yang dibutuhkan ***35 menit***.

g. **Analisis Hasil**

Dari hasil pengujian tersebut, menunjukkan perbedaan antara kedua rumus yang diujicoba. Bila menggunakan *rumus 3* (*dikjstra konvensional*) maka hasilnya tidak berbeda antara iterasi yang dimulai pada jam 07.00 dan iterasi yang dimulai pada jam 22.00. Namun bila menggunakan rumus 4 (*metode yang diusulkan*), terdapat perbedaan antara iterasi yang dimulai pada jam 07.00 dan iterasi yang dimulai pada jam 22.00.
Perbedaan hasil tersebut dikarenakan perbedaan bobot pada rumus 3 dan rumus 4. Bobot yang statik mengakibatkan hasil yang sama pada waktu iterasi

yang berbeda. Sedangkan bobot yang *dinamik* mengakibatkan perbedaan hasil bila *iterasi* dilakukan pada jam yang berbeda.

Hasil pada rumus 4 (*metode yang diusulkan*) lebih mendekati keadaan yang sebenarnya. Karena situasi kepadatan di jalan raya tidaklah sama antara waktu yang satu dengan waktu yang lainnya *(berubah-ubah)*.

Sebuah penelitian tentunya memiliki keterbatasan, keterbatasan yang dapat diambil dari penelitian ini diharapkan dapat diperbaiki pada penelitian-penelitian selanjutnya. Keterbatasan tersebut adalah sebagai berikut :

1. Alokasi waktu masih kurang *detail*. Masih menggunakan waktu awal saja. Cara ini memiliki kelemahan dari sisi *akurasi*. Yaitu waktu yang digunakan adalah saat akan dimulainya perjalanan pada sebuah jalur. Sedangkan untuk menempuh sebuah jalur dibutuhkan alokasi waktu tempuh yang berbentuk *range* dari awal sampai akhir, dan bisa saja pada suatu saat waktu akhir akan melewati alokasi waktu yang ada pada *n* waktu (tabel 5).
2. Data waktu tempuh masing-masing jalur (tabel 5) harus dilakukan perbaikan pada sisi *pemutakhiran*. Pemutakhiran data tersebut harus dilakukan karena situasi pada suatu daerah yang tentunya dapat berubah sesuai perkembangan daerah tersebut. Penyebabnya antara lain :
   a. Pertambahan jumlah penduduk yang mempengaruhi pembangunan perumahan dan pertokoan, dapat mengakibatkan pertambahan jumlah dan jenis kendaraan. Hal tersebut bisa merubah waktu tempuh untuk melewati sebuah *edge/jalur*.
   b. Perubahan kondisi jalan akibat dari proses pembangunan fisik daerah. Hal tersebut juga bisa mengakibatkan perubahan waktu tempuh untuk melewati sebuah edge/jalur.

3. Kesimpulan

Dari penelitian ini dapat ditarik kesimpulan bahwa metode yang diusulkan dapat digunakan untuk mengatasi keterbatasan *algoritma dijkstra* dalam menghadapi perbedaan bobot jalur yang berubah-ubah.

Saran :
Perlu dilakukan penelitian lanjut untuk kesempurnaan penelitian ini. Penelitian yang bisa dilakukan selanjutnya adalah :

1. Menambah akurasi dimulai dan diakhirinya sebuah penempuhan rute. Yaitu waktu yang digunakan adalah saat akan dimulainya perjalanan pada sebuah *edge/jalur*. Permasalahan muncul saat *alokasi waktu* yang *diperlukan* melewati *alokasi waktu* yang telah *ditentukan* (*alokasi waktu yang ditentukan dapat dilihat pada tabel 5*).
2. Pemutakhiran data.
   Pemutakhiran data tersebut harus dilakukan karena situasi pada suatu daerah tentunya berubah sesuai perkembangan daerah tertentu.

Daftar Pustaka

[1] A. F. Nurul Alam, N. Gamayanti, and A. Alkaff, "Algoritma Improved Ant Colony System (IACS) untuk menyelesaikan Dynamic Vehicle Routing Problem with Time Window dengan Variabel Travel Time."
[2] A. Mukhsinin, A. Imran, and S. Susanty, "Penentuan Rute Distribusi CV. IFFA Menggunakan Metode Nearest Neighbour dan Local Search," *J. Online Inst. Teknol. Nas.*, vol. 01.
[3] B. Liu, S.-H. Choo, and S.-L. Lok, "Finding the Shortest Route Using Cases, Knowledge, and Dijkstra's Algorithm," *CAIA*, vol. 1994.
[4] D. Zhang, Z. Wei, J.-H. Kim, and S. Tang, "An Optimized Dijkstra Algorithm for Embedded-GIS," *ICCDA*, vol. 2010.
[5] K. R. Rao, "Design & Analisys of Algorithms - In Simple Way," *KL Univ.*, vol. 2010.
[6] "Data Structure and Algorithms Analysis," vol. Edition 3.2 (Java Version), Departemen of Computer Science Vifginia Tech Blacksburg, 2012.
[7] Mutakhiroh, "Pemanfaatan Metode Heuristik Dalam Pencarian Jalur Terpendek Dengan Algoritma Semut dan Algoritma Genetika," *Univ. Islam Indones.*, 2007.
[8] Z. Yan and Z. Jun, "Dijkstra's Algorithm Based Robust Optimization to Airline Network Planning," *IEEE*.
[9] P. Bina Marga, "Manual Kapasitas Jalan Indonesia." Direktorat Jenderal Bina Marga, Feb-1997.
[10] O. Z. Tamin, *Perencanaan Pemodelan & Rekayasa Transportasi*. ITB, 2008.
[11] D. Xie, H. Zhu, L. Yan, S. Yuan, and J. Zhang, "An improved Dijkstra algorithm in GIS application," *Proc. 2010 Conf. Dependable Comput. CDC'2010*.

Biodata Penulis

Isnaeni Setiyadi, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika STMIK BINA MULIA PALU Sulawesi Tengah, lulus tahun 2011. Saat ini sedang menempuh pendidikan Program Pasca Sarjana Magister Teknologi Informasi Jurusan Teknik Elektro dan Teknologi Informasi Fakultas Teknik Universitas Gajah Mada Yogyakarta.

Teguh Bharata Adji, memperoleh gelar Sarjana Teknik (ST), Jurusan Teknik Elektro dan Teknologi Informasi Fakultas Teknik Universitas Gadjah Mada Yogyakarta, Program Magister di Jepang, gelah Phd di peroleh di Petronas Malaysia. Sekarang menjadi Dosen tetap di Jurusan Teknik Elektro dan Teknologi Informasi Fakultas Teknik Universitas Gadjah Mada Yogyakarta.