

# Solusi Optimal Pencarian Jalur Tercepat dengan Algoritma Hybrid Fuzzy-Dijkstra

Moch. Hannats Hanafi Ichsan, Erni Yudaningtyas, M. Aziz Muslim

**Abstrak**— Pada kasus pencarian jalur, seringkali hanya panjang dari tiap ruas jalan yang dipergunakan sebagai parameter. Dalam kenyataannya banyak faktor yang semestinya digunakan sebagai pertimbangan dalam menentukan rute yang dipilih. Salah satu faktor yang mempengaruhi adalah kepadatan jalan juga sebagai pertimbangan untuk jalur yang akan dipilih. Tulisan ini, merupakan penelitian tentang optimasi logika fuzzy dan algoritma dijkstra pada kasus pencarian jalur tercepat. Logika fuzzy dipergunakan untuk memodelkan multi parameter yang dimiliki oleh jalan, yaitu pada kasus ini panjang dan kepadatan jalan. Model logika fuzzy menggunakan fuzzy sugeno orde-nol. Hasil keluaran dari logika fuzzy yang merupakan nilai dari tiap jalan, diolah dengan algoritma dijkstra. Algoritma dijkstra adalah algoritma untuk melakukan pencarian jalur terpendek. Sehingga hasil yang didapat dari optimasi kedua algoritma adalah hasil optimal pencarian jalur tercepat.

**Kata Kunci** — fuzzy-dijkstra, multi parameter pencarian jalur.

## I. PENDAHULUAN

DALAM aktifitas sehari-hari, penggunaan jalan selalu menjadi aktifitas yang tidak dapat dihindari. Dalam proses berpindah dari lokasi awal sampai menuju lokasi tujuan, melewati beberapa ruas jalan. Jalan adalah prasarana perhubungan darat yang dipergunakan bagi lalu lintas kendaraan, orang dan hewan tetapi tidak untuk kereta api [1]. Jalan, adalah sarana perhubungan yang tiap rutennya adalah jaringan yang rumit [2]. Jumlah kendaraan saat ini bertambah dengan pesat tetapi tidak diiringi dengan perbaikan kondisi jalan [3]. Sehingga terjadi penumpukan kendaraan pada ruas jalan tertentu dan terjadi kemacetan ataupun perjalanan tidak lancar. Fenomena alam yang terjadi pada pengambilan jalur, tidak selalu terpaku dengan jumlah kendaraan yang melewati suatu ruas jalan tertentu sehingga mempengaruhi jumlah kepadatan jalan. Sehingga pada implementasi analisa dan simulasi, dipergunakan pembangkitan bilangan acak atau masukan manual untuk diproses lebih lanjut.

Hasil pencarian jalur tercepat, dapat dimanfaatkan untuk mencapai lokasi yang dituju pada umumnya. Pada

kondisi khusus, antara lain seperti mencari lokasi tempat wisata [2], membantu ambulan untuk menuju rumah sakit [3], menghindari kemacetan dan pusat keramaian [4], meminimalisir penggunaan bahan bakar [5], memprediksi aliran lalu lintas yang akan dilewati [6], dan untuk menentukan posisi *router* yang tepat untuk mengefisienkan kecepatan internet [7].

Logika fuzzy digunakan untuk memodelkan kuantitas dari input [8]. Logika ini digunakan untuk situasi model dimana pembuatan keputusan dalam lingkup yang kompleks dan sulit untuk melakukan pengembangan model matematis [9]. Semua nilai keluaran dari logika fuzzy, bisa dipakai sebagai input algoritma yang lain [7]. Logika fuzzy sangat dekat dengan pemikiran manusia [10], secara luas diterima dan diaplikasikan pada berbagai permasalahan nyata [6], serta untuk memodelkan pengetahuan yang dimiliki oleh manusia [5]. Oleh karena itu, penulis menggunakan logika fuzzy untuk memodelkan karakteristik yang dimiliki oleh jalan. Logika fuzzy seringkali dipergunakan untuk menyelesaikan permasalahan kontrol otomatisasi sistem [11], tetapi pada kasus optimasi pencarian jalur tercepat juga bisa dilakukan [6].

Algoritma dijkstra adalah algoritma untuk memecahkan masalah jalur terpendek, dengan nilai tiap jalur non-negatif [12]. Algoritma ini, menyelesaikan masalah dengan menghasilkan satu rute [13], dari satu lokasi awal sampai satu lokasi tujuan [14], untuk menemukan rute terdekat, tercepat ataupun termudah [15].

Oleh karena itu dibutuhkan sebuah analisis serta perhitungan matematis yang sangat detail dalam proses menentukan rute jalan yang akan dipilih. Penulis dengan pertimbangan karakteristik jalan serta fenomena yang terjadi, maka dipergunakan logika fuzzy untuk memberi pertimbangan yang spesifik dalam memberikan nilai bobot tiap ruas jalan dan algoritma dijkstra untuk mencari rute yang diambil, sehingga didapat rute jalan tercepat.

## II. METODOLOGI

Pengumpulan data diambil dengan beberapa cara antara lain, survey, pembagian kuisioner, pengambilan data dari arsip-arsip yang sudah ada pada dinas terkait, wawancara dan observasi. Data yang dibutuhkan antara lain :

- a) data panjang jalan
- b) data kepadatan jalan

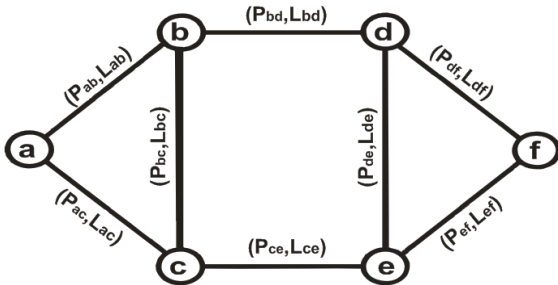
M. Hannats Hanafi Ichsan adalah mahasiswa Program Magister Fakultas Teknik Universitas Brawijaya Malang (082334050999, hanas.hanafi@gmail.com)

Erni Yudaningtyas adalah Dosen pada jurusan Teknik Elektro Universitas Brawijaya Malang (erni@ub.ac.id)

M. Aziz Muslim adalah Dosen jurusan Teknik Elektro Universitas Brawijaya Malang (muh\_aziz@ub.ac.id)

Variabel serta atribut penelitian yang digunakan antara lain :

- a)  $\mu$  = variabel parameter (*road-length*, dan *density*)
- b)  $P$  = *road-length* (*short*, *moderate*, dan *long*)
- c)  $L$  = *road-density* (*loose*, *normal*, dan *dense*)
- d)  $O$  = output (*smooth*, *moderate*, *little-dense*, *creep-solid*, dan *jams*)

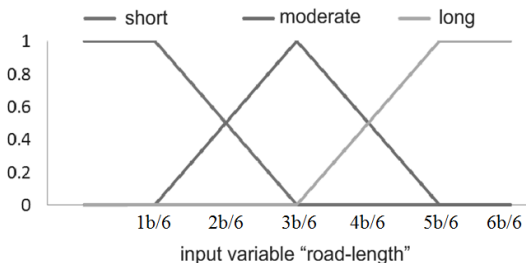


Gambar 1. Representasi Jalan pada Graf yang Memiliki dua parameter dalam tiap busur

Sebuah jalan memiliki beberapa parameter untuk pertimbangan pengambilan keputusan jalur mana yang akan diambil seperti pada Gambar 1. Dalam kasus ini digunakan panjang dan kepadatan jalan. Selanjutnya dari representasi graf yang menggambarkan sebuah jaringan jalan, dibuat fungsi keanggotaannya. Pada fungsi keanggotaan panjang dan kepadatan jalan, nilai keanggotaan pada masing-masing grafik direpresentasikan dengan huruf a dan b. Huruf a dan b merupakan nilai maksimum panjang dan kepadatan jalan pada node graf a sampai f, yaitu node awal sampai node akhir. Sehingga masing-masing nilai akan menjadi batas akhir dari fungsi keanggotaan yang dimiliki oleh masing-masing parameter. Penggunaan huruf a dan b, adalah agar jika ada perubahan nilai pada parameter, tidak mengubah bentuk dari fungsi keanggotaan yang dimiliki.

A. Fungsi keanggotaan panjang jalan

Pada fungsi keanggotaan panjang jalan, menggambarkan derajat keanggotaan yang dimiliki oleh panjang jalan, yang memiliki tiga derajat keanggotaan. Antara lain, pendek, sedang dan panjang.



Gambar 2. Fungsi Keanggotaan Himpunan Panjang Jalan

Tiap fungsi keanggotaan, memiliki fungsi linear. Fungsi linear dipergunakan untuk memberikan informasi nilai yang dimiliki tiap derajat keanggotaan. Fungsi linear yang dimiliki oleh panjang jalan sebagai berikut. Fungsi linear untuk himpunan *short* :

$$\mu_{jshort}(P) \begin{cases} 1 & ; P \leq 1a/6 \\ \frac{(1a/6 - P)}{(3a/6 - 1a/6)} & ; 1a/6 \leq P \leq 3a/6 \\ 0 & ; P \geq 3a/6 \end{cases} \quad (1)$$

Fungsi linear untuk himpunan *moderate* :

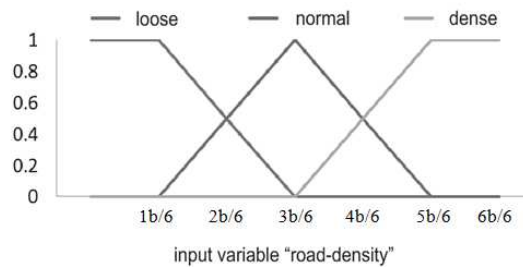
$$\mu_{jmoderat}(P) \begin{cases} 0 & ; P \leq 1a/6 \text{ atau } P \geq 5a/6 \\ \frac{(P - 1a/6)}{(3a/6 - 1a/6)} & ; 1a/6 \leq P \leq 3a/6 \\ \frac{(5a/6 - P)}{(5a/6 - 3a/6)} & ; 3a/6 \leq P \leq 5a/6 \end{cases} \quad (2)$$

Fungsi linear untuk himpunan *long* :

$$\mu_{jlong}(P) \begin{cases} 0 & ; P \leq a/3 \\ \frac{(P - 1a/6)}{(5a/6 - 3a/6)} & ; 3a/6 \leq P \leq 5a/6 \\ 1 & ; P \geq 5a/6 \end{cases} \quad (3)$$

B. Fungsi keanggotaan kepadatan jalan

Pada fungsi keanggotaan kepadatan jalan, menggambarkan derajat keanggotaan yang dimiliki oleh kepadatan jalan, yang memiliki tiga derajat keanggotaan. Antara lain: *loose*, *normal* dan *padat*.



Gambar 3. Fungsi Keanggotaan Himpunan Kepadatan Jalan

Fungsi keanggotaan untuk panjang jalan digambarkan pada Gambar 3. Tiap fungsi keanggotaan, memiliki fungsi linear sebagai berikut.

Fungsi linear untuk himpunan *loose* :

$$\mu_{jloose}(L) \begin{cases} 1 & ; L \leq 1b/6 \\ \frac{(1b/6 - L)}{(3b/6 - 1b/6)} & ; 1b/6 \leq L \leq 3b/6 \\ 0 & ; L \geq 3b/6 \end{cases} \quad (4)$$

Fungsi linear untuk himpunan *normal* :

$$\mu_{jnormal}(L) \begin{cases} 0 & ; L \leq 1b/6 \text{ atau } L \geq 5b/6 \\ \frac{(P - 1b/6)}{(3b/6 - 1b/6)} & ; 1b/6 \leq L \leq 3b/6 \\ \frac{(5b/6 - L)}{(5b/6 - 3b/6)} & ; 3b/6 \leq L \leq 5b/6 \end{cases} \quad (5)$$

Fungsi linear untuk himpunan *dense* :

$$\mu_{jdense}(L) \begin{cases} 0 & ; L \leq 3b/6 \\ \frac{(L - 1b/6)}{(5b/6 - 3b/6)} & ; 3b/6 \leq L \leq 5b/6 \\ 1 & ; L \geq 5b/6 \end{cases} \quad (6)$$

III. PROSES INFERENSI FUZZY

Jika pada tiap fungsi keanggotaan memiliki tiga variabel keanggotaan, maka menghasilkan aturan atau *rule evaluation* seperti pada Tabel I.

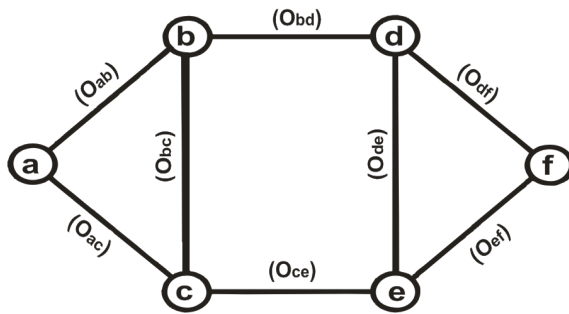
TABEL I.  
RULE EVALUATION

Weight		Road-Density		
		Loose	Normal	Dense
Road	Short	0,1	0,25	0,5
Length	Moderate	0,25	0,5	0,75
	Long	0,5	0,75	1

Pada Tabel I, *rule evaluation* dituliskan semua kemungkinan dari kombinasi empat parameter. Setelah proses *rule evaluation*, untuk menghasilkan keluaran dari logika fuzzy, penulis menggunakan fuzzy model sugeno untuk memodelkan hasil dari logika fuzzy. Notasi 0,1 sampai 1 adalah nilai ataupun bobot untuk menunjukkan tingkat atau grade yang dimiliki oleh jalan. Nilai tersebut adalah himpunan untuk masukan *rule evaluation* yang dimiliki oleh logika fuzzy. Pada proses fuzzy output, menggunakan model sugeno orde-nol.

If (Xi is Pi) AND (Yi is Li) then z = O (7)

P adalah parameter panjang jalan, L adalah parameter. Karena pada kasus pencarian jalur tercepat dicari nilai paling kecil yang dimiliki oleh tiap jalan, maka menggunakan operator logika fuzzy "AND". Jadi pembacaan rule adalah "If Panjang jalan(P) and kepadatan jalan(L) then jalan(O)". Dan *i* adalah setiap busur yang dimiliki.



Gambar 4. Hasil Graf dari Proses Logika Fuzzy

TABEL II.  
BOBOT PADA TIAP SIMPUL PADA GRAF SETELAH DIPROSES LOGIKA FUZZY

Simpul	Bobot
(a,b)	Oab
(a,c)	Oac
(b,c)	Obc
(b,d)	Obd
(c,e)	Oce
(d,e)	Ode
(d,f)	Odf
(e,f)	Oef

Setelah mendapatkan hasil dari logika fuzzy, maka diproses dengan algoritma dijkstra. Pada algoritma ini,

memiliki bobot tiap simpul yaitu "O" seperti digambarkan pada Gambar 4 dan pada Tabel II.

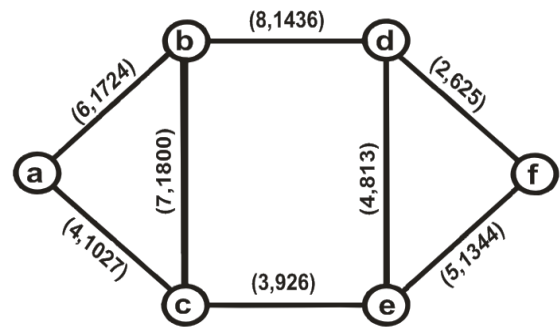
IV. HASIL DAN PEMBAHASAN

Pada tahap ini, algoritma akan diuji. Penggabungan logika fuzzy dan algoritma dijkstra memiliki perhitungan yang detail dan panjang. Proses pertama dari proses pencarian jalur yaitu pemberian nilai dari parameter yang dimiliki, antara lain panjang jalan dan kepadatan jalan.

TABEL III.  
NILAI BOBOT TIAP SIMPUL.

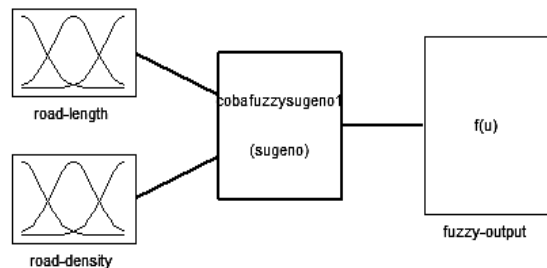
Simpul	Length weight	Density weight
(a,b)	6	1724
(a,c)	4	1027
(b,c)	7	1800
(b,d)	8	1436
(c,e)	3	926
(d,e)	4	813
(d,f)	2	625
(e,f)	5	1344

Setiap simpul, memiliki dua nilai parameter. Nilai tersebut sebagai nilai masukan dari algoritma fuzzy. Nilai untuk masing-masing simpul dapat dilihat pada Tabel III. Nilai tersebut berdasarkan survei, atau pemberian nilai acak untuk proses pengujian algoritma. Nilai yang dimiliki pada Tabel III, digambarkan pada Gambar 5.



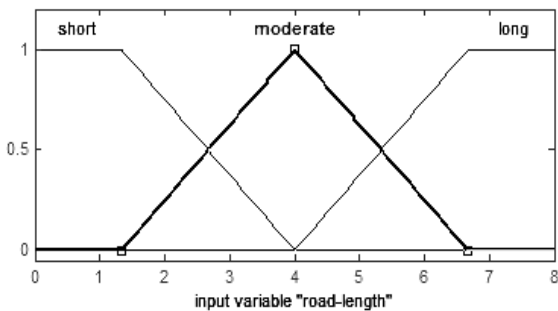
Gambar 5. Sebuah graf yang memiliki dua nilai untuk masing-masing parameter

Setiap nilai panjang jalan maupun kepadatan jalan, dimasukkan kedalam fungsi keanggotaan yang dimiliki oleh logika fuzzy. Tiap simpul dimasukkan satu persatu.



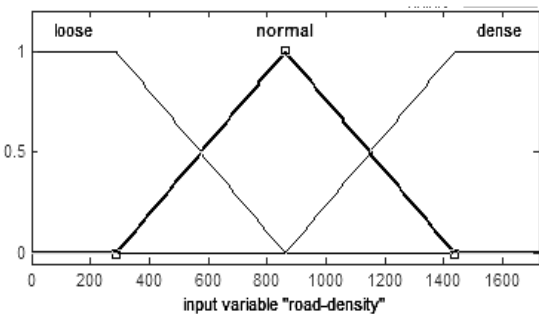
Gambar 6. Proses Logika Fuzzy

Proses logika fuzzy, seperti yang dimiliki oleh Gambar 6. Yaitu parameter panjang jalan memiliki fungsi keanggotaan sendiri, begitu pula kepadatan jalan. Kemudian kedua parameter akan diolah dengan fuzzy sugeno dan menghasilkan *fuzzy output*.



Gambar 7. Fungsi Keanggotaan Panjang Jalan

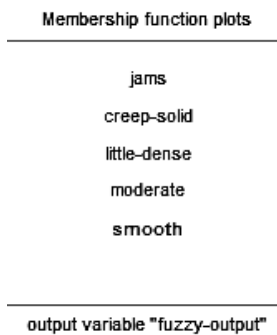
Begitu juga fungsi keanggotaan kepadatan jalan dibagi menjadi tiga variabel parameter. Gambar 7 memberikan informasi tentang model fungsi keanggotaan kepadatan jalan. Masing masing variabel parameter memiliki batas yang berbeda, sesuai dengan data panjang jalan yang dimiliki oleh jalur yang akan diambil.



Gambar 8. Fungsi keanggotaan Kepadatan jalan

If (road-length is short) and (road-density is loose) then (fuzzy-output is smooth) (0.1)  
 If (road-length is short) and (road-density is normal) then (fuzzy-output is moderate) (0.25)  
 If (road-length is short) and (road-density is dense) then (fuzzy-output is little-dense) (0.5)  
 If (road-length is moderate) and (road-density is loose) then (fuzzy-output is moderate) (0.25)  
 If (road-length is moderate) and (road-density is normal) then (fuzzy-output is little-dense) (0.5)  
 If (road-length is moderate) and (road-density is dense) then (fuzzy-output is creep-solid) (0.75)  
 If (road-length is long) and (road-density is loose) then (fuzzy-output is little-dense) (0.5)  
 If (road-length is long) and (road-density is normal) then (fuzzy-output is creep-solid) (0.75)  
 If (road-length is long) and (road-density is dense) then (fuzzy-output is jams) (1)

Gambar 9. Rule evaluation



Gambar 10. Fuzzy Output

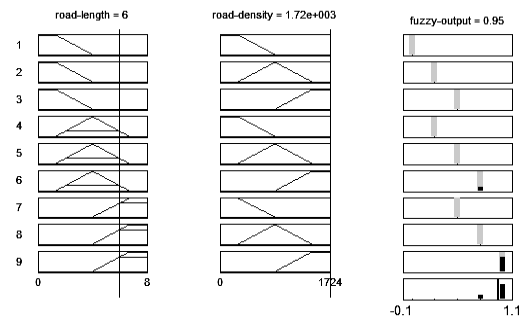
Fungsi keanggotaan kepadatan jalan dibagi menjadi

tiga variabel parameter. Gambar 8 memberikan informasi tentang model fungsi keanggotaan kepadatan jalan. Masing masing variabel parameter memiliki batas yang berbeda, sesuai dengan data kepadatan jalan yang dimiliki oleh jalur yang akan diambil.

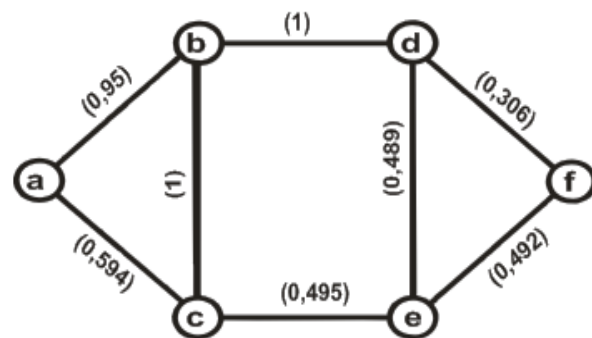
Masing masing parameter, yaitu panjang jalan dan kepadatan jalan, memiliki tiga variabel parameter, maka *rule evaluation* yang dimiliki oleh proses ini ada 9. Detail aturan yang dimiliki pada Gambar 9. Bobot yang paling ringan adalah 0.1 adalah jalan dengan bobot yang paling kecil sampai bobot yang paling besar.

Pada himpunan *fuzzy output*, memiliki lima variabel. Antara lain lancar (*smooth*), sedang (*moderate*), sedikit padat (*little dense*), padat merayap (*creep-solid*), dan macet atau sangat padat (*dense*). Yang masing masing memiliki batas nilai 0-0,1(*smooth*); 0,25(*moderate*); 0,5(*little dense*); 0,75(*creep-solid*) dan 1(*dense*).

Proses terakhir dari algoritma fuzzy adalah proses defuzzifikasi. Jika diambil contoh untuk proses defuzzifikasi, untuk node (a,b) memiliki panjang jalan sebesar 6 dan kepadatan 1724 maka proses defuzzifikasi seperti pada Gambar 11 yang memiliki nilai fuzzy output sebesar 0,95. Nilai panjang dan kepadatan diproses dengan 9 *rule evaluation* yang dimiliki.



Gambar 11. Proses Defuzzifikasi

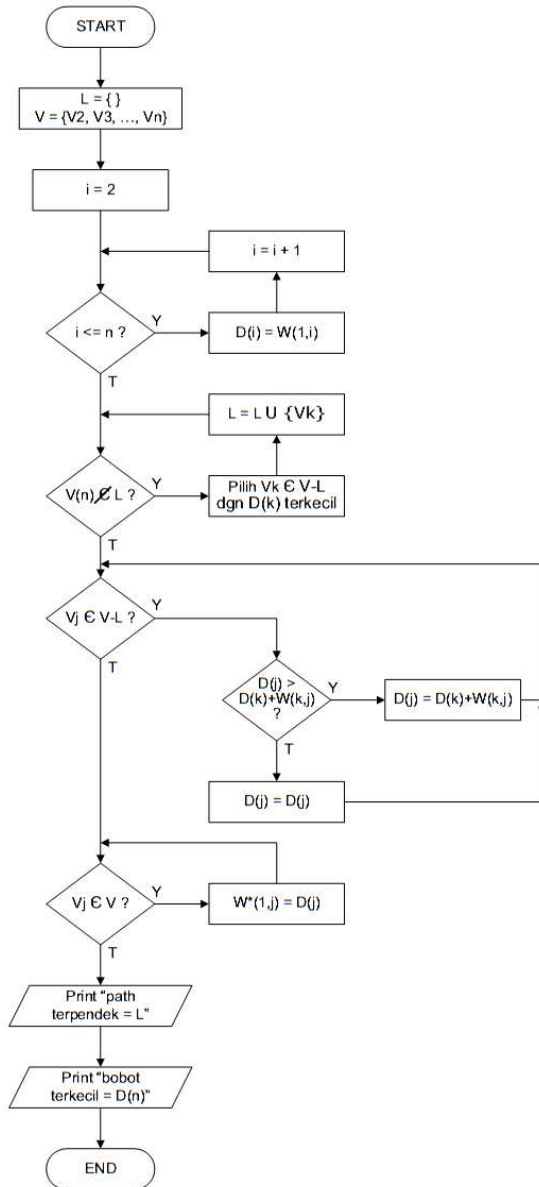


Gambar 12. Graf hasil keluaran logika fuzzy

Setelah masing-masing simpul dari jalan masuk dalam proses logika fuzzy, maka akan didapatkan hasil keluaran seperti yang digambarkan pada Gambar 12. Yaitu tiap ruas node jalan, masing-masing node memiliki satu nilai. Yaitu hasil dari dua parameter yang diproses dengan logika fuzzy. Sehingga menghasilkan *fuzzy output* yang dimiliki oleh tiap node jalan. Hasil ini yang dipergunakan dalam pemilihan rute jalan, yang diproses dengan algoritma dijkstra.

Masing-masing jalan memiliki nilai yang berbeda

sesuai dengan masukan awal, sekarang graf jalan sudah memiliki satu nilai yaitu fuzzy output (lihat Tabel IV). Fuzzy output digunakan sebagai masukan dari algoritma dijkstra. Proses dari algoritma dijkstra, antara lain seperti langkah berikut. Algoritma mencari rute terpendek ini dikembangkan oleh Dijkstra [20] atau dijelaskan pada Gambar 13.



Gambar 13. Flowchart Algoritma Dijkstra (Jong Jek Siang, 2002).

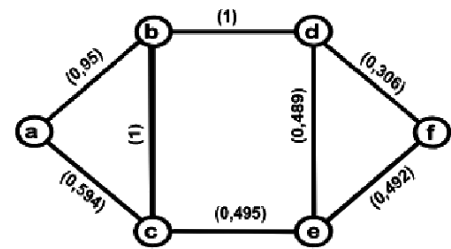
Misalkan :  
 $V(G) = \{V1, V2, \dots, Vn\}$   
 $L$  = Himpunan titik-titik  $\in V(G)$  yang sudah terpilih dalam alur path terpendek  
 $D(j)$  = Jumlah bobot path terkecil dari  $V1$  ke  $Vj$   
 $W(i,j)$  = Bobot garis dari titik  $Vi$  ke titik  $Vj$   
 $W^*(1,j)$  = Jumlah bobot terkecil dari  $V1$  ke  $Vj$   
 Maka, algoritma Dijkstra untuk mencari path terpendek adalah sebagai berikut :  
 a)  $L = \{ \}$ ;  $V = \{V2, V3, \dots, Vn\}$   
 b) Untuk  $i = 2, \dots, n$ , lakukan  $D(i) = W(1,i)$

- c) Selama  $Vn \in L$  lakukan :
  - a. Pilih titik  $Vk \in V-L$  dengan  $D(k)$  terkecil  
 $L = L \cup \{Vk\}$
  - b. Untuk setiap  $Vj \in V-L$  lakukan :  
 Jika  $D(j) > D(k) + W(k,j)$  maka ganti  $D(j)$  dengan  $D(k) + W(k,j)$
- d) Untuk setiap  $Vj \in V$ ,  $W^*(1,j) = D(j)$

TABEL IV.  
FUZZY OUTPUT

Simpul	Rule	Fuzzy Output
(a,b)	9	0,95
(a,c)	6	0,594
(b,c)	9	1
(b,d)	9	1
(c,e)	5	0,495
(d,e)	5	0,489
(d,f)	5	0,306
(e,f)	1	0,82

Semua proses dilakukan sampai *node* f, sehingga dari *node* a sampai *node* f didapatkan solusi jalur tercepat. Menurut Gambar 4.8, jalur terpendek dari titik  $V1$  ke  $Vn$  adalah melalui titik-titik dalam  $L$  secara berurutan dan jumlah bobot jalur terkecilnya adalah  $D(n)$ .



Gambar 14. Hasil rute yang dipilih

Setelah ke-tiga langkah tersebut dilakukan, maka akan didapatkan jarak yang menjadi hasil final dari jalur yang dipilih. Sehingga memiliki hasil pada gambar 14. Yaitu rute a-c-e-f sebagai rute yang dipilih. Dengan total *fuzzy output* adalah 1,581.

V. KESIMPULAN DAN SARAN

Permasalahan pencarian jalur, dengan menggunakan berbagai macam kriteria, dapat diselesaikan dengan gabungan antara logika fuzzy dan algoritma dijkstra. Proses pencarian jalur dengan gabungan dua algoritma ini, memerlukan proses berfikir yang detail. Dengan pendekatan ini, maka bisa dimasukkan beberapa parameter yang lain, misal kecepatan rata-rata pengguna kendaraan, volume jalan yang dimiliki, ataupun tingkat kerusakan jalan. Atau beberapa parameter lain yang mempengaruhi tingkat laju kendaraan. Dengan pendekatan algoritma ini, nilai yang dimiliki oleh jalan selalu dinamis, sehingga proses yang dilalui akan bisa berubah setiap saat, dan rute yang dipilih bisa berubah setiap saat.

Untuk saran, pengukuran parameter panjang jalan hanya membutuhkan satu kali pengukuran. Untuk parameter kepadatan dan kecepatan rata-rata pengguna

jalan, harus dapat diukur setiap waktu. Atau dengan kata lain, harus ada masukan secara *realtime* untuk memaksimalkan algoritma ini.

#### DAFTAR PUSTAKA

- [1] Peraturan Menteri Perhubungan, nomor : KM 14 tahun 2006, Manajemen dan Rekayasa Lalu Lintas di Jalan.
- [2] Pugas O.D., Somantri M., Satoto K.I., Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra dan Astar (A\*) pada SIG Berbasis Web Untuk Pemetaan Pariwisata Kota Sawah Lunto.
- [3] Shekar B.S., Kumar G.N., Rani H.V.U., Divyashree C.K., George G., and Murali A., GPS Based Shortest Path for Ambulances using VANETS, ICWN 2012, V49.35.
- [4] Efentakis, A., Theodoratis, D., dan Pfoser, D., 2012 Crowdsourcing Computing Resources For Shortest Path Computation, ACM sigspatial GIS, USA.
- [5] Linda, O. dan Manic, M. 2012. Improving Vehicle Fleet Fuel Economy via Learning Fuel Efficient Driving Behaviors, IEEE.
- [6] Wang, Y., dan Chen, Y. 2012. Short Term Traffic Flow Prediction By A Sugeno Fuzzy System Based On Gaussian Mixture Models, JATIT & LLS, Vol 44 No. 1.
- [7] Teuber, A., and Eissfeller B., 2006. WLAN Indoor Positioning Based on Euclidian Distances and Fuzzy Logic.
- [8] Golnarkar, A., Alesheikh A. A., dan Malek, M. R. (2010). Solving Best Path Problem on Multimodal Transportation Network with Fuzzy Costs, IJFS Vol. 7 no. 3, pp. 1-13.
- [9] Shankar, N. Ravi, V. Shireesha, dan P. Phani B. R. 2010. An Analytical Method for Finding Critical Path in a Fuzzy Project Network, Int. J. Contemp. Math. Sciences. Vol 5, no. 20, 953-962.
- [10] Kaur, Arshdeep dan Kaur, Amrit. 2012. Comparison Of Mamdani-Type And Sugeno-Type Fuzzy Inference Systems For Air Conditioning System, IJSCE Vol. 2 Issue 2.
- [11] Dewi, A.R. Wahjono E., dan Karim S., 2012, Pemanfaatan Kontrol Logika Fuzzy Untuk Pemanfaatan Adhesive Glue, SCITEC, TE-03-062.
- [12] Jordan, Anca E. 2012 Development of an Interactive Environment Used for Simulation of Shortest Path Algorithm, Annals of Faculty Engineering Huneodara.
- [13] Dai, L. 2005. Fast Shortest Path Algorithm For Route Network And Implementation.
- [14] Kesswani, N., dan Gopalani, D. 2011. Design and Implementation of Multi-Parameter Dijkstra (MPD) Algorithm : A Shortest Path Algorithm for Real Road Networks, IJAER, Vol. No. 2, Issue III.
- [15] Navratil, G. 2012. Curviness As A Parameter For Route Determination , FDE VERLAG GMBH, GI Forum.
- [16] Dijkstra, E.W. 1959. A note on two problems in connexion with graphs. Numerische Mathematic. Vol.1, Mathematisch Centrum, Amsterdam, The Netherland 269-271.
- [17] Ichsan, M. Hannats H. 2011. Pemanfaatan Logika Fuzzy dalam SIG Untuk Pencarian Jalur Berdasarkan Kepadatan dan Panjang Jalan, SEIE A2-82.
- [18] Kusumadewi, S. dan Purnomo, H. 2004. Aplikasi Logika Fuzzy Untuk Pendukung Keputusan, Graha Ilmu.
- [19] Kusumadewi, Sri, Sistem Inferensi Fuzzy Metode TSK Untuk Penentuan Kalori Harian.
- [20] Siang, J. Jek. 2002. Matematika Diskrit dan Aplikasinya pada Ilmu Komputer, Yogyakarta : Andi Offset.