Jurnal EurekaMatika

Journal homepage: https://ejournal-science.upi.edu/jem/index
doi: https://doi.org/10.17509/jem.v14i1.69

Optimisasi Jalur Evakuasi Menggunakan Algoritma *Fuzzy* Dijkstra (Studi Kasus: Jalur Evakuasi pada Gedung FPMIPA A UPI)

Nazmi Rizki Awaliyah, Khusnul Novianingsih, dan Ririn Sispiyati*

Program Studi Matematika, Universitas Pendidikan Indonesia
*Correspondence author: k_novianingsih@upi.edu

ABSTRAK

Perencanaan jalur evakuasi merupakan aspek yang sangat penting dalam menentukan keberhasilan proses evakuasi bencana. Penelitian ini meneliti masalah perencanaan jalur evakuasi pada gedung, yaitu jalur evakuasi yang dapat dilalui dengan waktu evakuasi yang paling singkat dengan mempertimbangkan parameter panjang jalan, lebar jalan, dan jumlah pengungsi. Jalur evakuasi tersebut ditentukan menggunakan Algoritma Dijkstra, dengan merepresentasikan gedung sebagai graf berbobot. Selanjutnya, jarak terpendek dihitung dari setiap simpul sumber ke simpul tujuan. Bobot graf, yang merupakan indeks waktu evakuasi, diperoleh dengan menerapkan logika *fuzzy* dengan tiga variabel *input*, yakni panjang jalan, lebar jalan, dan jumlah pengungsi. Hasil implementasi Algoritma *Fuzzy* Dijkstra pada penentuan jalur evakuasi di Gedung FPMIPA A UPI menunjukan bahwa metode yang diusulkan berhasil menemukan jalur evakuasi dengan indeks waktu tercepat di Gedung FPMIPA A UPI. Lebih jauh, hasil tersebut menunjukan bahwa struktur Gedung FPMIPA A UPI saat ini sudah sangat baik karena mendukung kelancaran proses evakuasi dengan tingkat efisiensi dan efektivitas yang tinggi dalam situasi darurat.

© 2026 Kantor Jurnal dan Publikasi UPI

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima 9 Maret 2026
Direvisi 24 April 2026
Disetujui 13 Mei 2026
Tersedia online 15 Mei 2026
Dipublikasikan 15 Mei 2026

Kata Kunci:
Algoritma Dijkstra,
Jalur evakuasi,
Jalur terpendeki,
Logika fuzzy,
Optimisasi.

ABSTRACT

Evacuation route planning is a very important aspect in determining the success of the disaster evacuation process. This study examines the problem of evacuation route planning in buildings to determine routes that minimize evacuation time, accounting for road width, length, and the number of evacuees. The building is modeled as a weighted graph, where the weight is the evacuation time index, computed using fuzzy logic with three input variables: road length, road width, and the number of evacuees. The shortest path from each source node to the destination node is calculated using the Dijkstra Algorithm. Then, the model is implemented to determine evacuation routes in the FPMIPA A UPI Building. The results show that the proposed method successfully finds the evacuation route with the fastest time index in the FPMIPA A UPI Building. Furthermore, these results indicate that the current structure of the FPMIPA A UPI Building is very good, as it supports a smooth evacuation process with high efficiency and effectiveness in emergency situations.

© 2026 Kantor Jurnal dan Publikasi UPI

Keywords:
Dijkstra Algorithm,
Evacuation routes,
Fuzzy logic,
Optimization,
Shortest path.

1. PENDAHULUAN

Menurut Undang-Undang Bangunan Gedung Nomor 28 Tahun 2002 Pasal 5 Ayat 1, bangunan gedung adalah struktur fisik yang berfungsi sebagai tempat manusia melakukan berbagai aktivitas. Berdasarkan aspek hukum mengenai Kesehatan dan Keselamatan Kerja (K3), keamanan, dan kesehatan adalah aspek penting dalam bangunan gedung, termasuk sistem evakuasi darurat yang efektif. Menurut Peraturan Menteri Pekerjaan Umum Nomor 26/PRT/M/2008, pada saat proses evakuasi, penghuni atau pengguna bangunan harus memiliki waktu yang cukup untuk meyelamatkan diri dengan aman dan meninggalkan gedung. Dalam penelitian yang dilakukan oleh Andhika *et al*., (2013), waktu optimal untuk proses evakuasi dipengaruhi oleh berbagai faktor, antara lain desain bangunan dan jalur evakuasi yang tersedia, jumlah orang yang perlu dievakuasi, serta elemen-elemen pendukung yang sesuai dengan standar terkait proses evakuasi.

Masalah penentuan jalur evakuasi tercepat dapat dipandang sebagai masalah pencarian lintasan terpendek. Algoritma Dijkstra merupakan salah satu algoritma yang populer digunakan untuk menyelesaikan masalah lintasan terpendek, termasuk perencanaan jalur evakuasi (Afandi & Mayasari, 2021). Algoritma Dijkstra bekerja dengan cara mengidentifikasi dan memperbarui lintasan terpendek dari simpul awal ke setiap simpul lain dalam graf berbobot dengan bobot non-negatif (Patel & Bagar, 2014). Algoritma Dijkstra telah tebukti berhasil diimplementasikan untuk menyelesaikan berbagai masalah seperti penyelesaian *Biobjective Shortest Path* (BSP), penentuan jalur multi-objektif, Graf Dijkstra, evakuasi darurat, permasalahan *fuzzy*, integrasi dengan fitur *location-based service*, dan distribusi rute optimal (Hakim *et al*., 2021).

Algoritma Dijkstra memiliki keunggulan dan kelemahan jika dibandingkan dengan algoritma lain. Berdasarkan Patel & Bagar (2014), pada masalah pencarian lintasan terpendek, Algoritma Dijkstra dapat memberikan hasil yang lebih cepat dan lebih cocok untuk masalah jaringan yang besar dibandingkan dengan Algoritma *Bellman-Ford*. Selain itu, pada penelitian Iskandar & Riti (2022) diperoleh hasil bahwa algoritma Dijkstra mampu menghasilkan rute lebih pendek dan waktu lebih lama dibandingkan algoritma *Greedy*. Sedangkan Tirastittam & Waiyawuththanapoom (2014) mengungkapkan bahwa waktu eksekusi dari algoritma Dijkstra lebih cepat dibandingkan dengan algoritma *Greedy*. Oleh karena itu penelitian ini memilih Algoritma Dijkstra berdasarkan pertimbangan kecepatan dan efisiensi yang telah terbukti pada penelitian sebelumnya dalam kasus pencarian jalur evakuasi tercepat.

Penentuan jalur evakuasi tercepat biasanya didasarkan oleh beberapa faktor, di antaranya adalah panjang jalan, lebar jalan, dan jumlah pengungsi. Karena penggunaan Algoritma Dijkstra memerlukan bobot tunggal, maka agar algoritma dapat digunakan pada penentuan jalur evakuasi diperlukan metode untuk mengonversi berbagai macam parameter jalur evakuasi menjadi sebuah nilai saja. Salah satu model yang dapat digunakan untuk keperluan tersebut adalah logika *fuzzy*. Logika *fuzzy* memungkinkan untuk mengatasi situasi di mana parameter seperti panjang jalan, lebar jalan, dan jumlah pengungsi yang tidak pasti. Pada logika *fuzzy*, parameter-parameter tesebut memiliki derajat keanggotaan dalam himpunan *fuzzy* yang menggambarkan tingkat ketidakpastian. Seberapa besar eksistensinya dalam himpunan tersebut dapat dilihat pada derajat keanggotaannya. Logika *fuzzy* ini bekerja dengan mengambil data masukan yang berupa nilai pasti dan mengubahnya menjadi bentuk kabur, mengevaluasi aturan-aturan *fuzzy*, menggabungkan hasilnya, dan kemudian mengubah hasil tersebut kembali menjadi nilai pasti yang dapat digunakan dalam pengambilan keputusan. Dengan demikian, penelitian ini akan mengefisienkan pencarian lintasan terpendek dengan Algoritma Dijsktra menggunakan logika *fuzzy* sebagai penentuan

bobot. Algoritma Dijsktra yang menggabungkan parameter *fuzzy* dikenal dengan sebutan Algoritma Fuzzy Dijkstra.

Penggunaan Algoritma *Fuzzy* Dijkstra pernah diteliti oleh beberapa peneliti sebelumnya. Deng *et al*., (2012) meneliti masalah jalur terpendek dalam lingkungan yang tidak pasti. Ichsan *et al*., (2012) mengoptimalkan jalur evakuasi di jalan, dan Iskierka & Lipiec (2019) mengoptimalkan jalur antar dua kota. Abdillah *et al*., (2019) menggunaan Algoritma *Fuzzy* Dijkstra dalam sistem pengangkutan sampah. Nggufron *et al*., (2019) meneliti masalah pencarian rute terbaik bagi pemadam kebakaran. Afandi & Mayasari (2021) dan juga Mayasari & Afandi (2021) menggunakan Algoritma *Fuzzy* Dijkstra untuk mengoptimalkan jalur evakuasi tsunami.

Penelitian ini bertujuan untuk mengoptimalkan jalur evakuasi pada sebuah gedung dengan menggunakan Algoritma *Fuzzy* Dijkstra. Selanjutnya, algoritma tersebut diimplementasikan untuk menyelesaikan masalah penentuan jalur evakuasi di Gedung FPMIPA A UPI. Penelitian sebelumnya dilakukan oleh Novianingsih *et al*., (2024) yang telah menggunakan Model *Maximum Dynamic Flow Problem* (MDFD) dengan pendekatan *Temporally Repeated Flow* untuk menentukan jalur evakuasi di Gedung FPMIPA A UPI. Berbeda dengan penelitian tersebut yang bertujuan untuk memaksimalkan jumlah pengungsi dalam waktu minimum dengan asumsi bahwa panjang dan lebar jalan adalah sama, penelitian ini bertujuan untuk meminimalkan waktu evakuasi dengan memperhatikan panjang jalan, lebar jalan, dan jumlah pengungsi yang sesungguhnya sehingga diharapkan akan diperoleh jalur evakuasi yang lebih akurat, karena melibatkan parameter yang lebih banyak dibandingkan penelitan sebelumnya. Hasil dari penelitian ini diharapkan dapat meningkatkan keamanan dan keselamatan penghuni gedung dalam situasi darurat.

2. METODE

Pada bagian ini dibahas representasi graf, model optimasi, dan teknik penyelesaian masalah menggunakan Algoritma *Fuzzy* Dijkstra.

2.1 Representasi Graf

Tahapan pertama dari penentuan jalur evakuasi di Gedung FPMIPA A UPI adalah merepresentasikan struktur bangunan dalam bentuk graf berbobot. Simpul pada graf mewakili lobi, sudut koridor, dan persimpangan koridor. Terdapat 3 jenis simpul, yaitu simpul sumber, simpul perantara, dan simpul tujuan. Simpul sumber merepresentasikan titik evakuasi atau titik-titik pada gedung yang memiliki tingkat kepadatan pengungsi yang tinggi seperti ruang kelas, ruang dosen, laboratorium, auditorium, kantin, dan mushola. Simpul tujuan mewakili titik kumpul di luar bangunan. Simpul perantara atau titik perantara merupakan simpul-simpul yang menghubungkan simpul sumber dengan simpul tujuan. Dua simpul dihubungkan dengan sebuah sisi jika terdapat tangga atau koridor yang menghubungkan kedua simpul tersebut. Setiap sisi memiliki bobot berupa nilai dari parameter panjang jalan, lebar jalan dan jumlah pengungsi secara berurutan yang selanjutnya akan dikonversi menjadi bobot tunggal berupa indeks waktu evakuasi menggunakan logika *fuzzy*. Misalkan e adalah sebuah sisi dengan $e = (i, j)$. Indeks waktu evakuasi pada sisi e mencerminkan berapa lama waktu yang dibutuhkan pengungsi untuk melakukan evakuasi dari simpul i ke simpul j. Pada penelitian ini, indeks waktu evakuasi diperoleh dengan mempertimbangkan panjang jalan, lebar jalan, dan banyak pengungsi yang dihitung dengan menggunakan logika *fuzzy*.

2.2 Asumsi dan Model Optimasi

Tujuan dari penelitian ini adalah menentukan jalur evakuasi yang optimal, yaitu jalur evakuasi tercepat yang dapat dilalui oleh seluruh pengungsi. Adapun asumsi-asumsi yang digunakan adalah sebagai berikut:

1. Indeks waktu evakuasi dari lokasi i ke lokasi j sama dengan indeks waktu evakuasi dari lokasi j ke lokasi i.
2. Simpul sumber mewakili beberapa ruangan yang saling berdekatan untuk mempermudah proses perhitungan.
3. Bobot sisi merupakan nilai indeks waktu evakuasi yang diperoleh menggunakan logika *fuzzy* dari parameter luas jalan, lebar jalan, dan jumlah pengungsi.
4. Banyak pengungsi dalam ruangan sama dengan kapasitas maksimum ruangan.
5. Seluruh pengungsi berada di dalam ruangan.
6. Arah sisi diasumsikan selalu menuju simpul tujuan terdekat.
7. Tidak terdapat gangguan selama proses evakuasi.
8. Lift tidak digunakan sebagai jalur evakuasi.
9. Atribut pengungsi seperti usia, jenis kelamin, dan lain-lain diabaikan.
10. Setiap pengungsi mengenal dengan baik setiap komponen dan tata letak gedung.

Tahapan pertama pemodelan adalah mendefinisikan himpunan dan parameter model. Diberikan graf $G = (V, E)$ yang merupakan representasi dari denah bangunan gedung. Misalkan S adalah himpunan simpul sumber pada graf G dan T adalah himpunan simpul tujuan pada graf G. Parameter w_{ij} menyatakan indeks waktu evakuasi yang diperlukan dari simpul i ke simpul j yang saling bertetangga. Variabel keputusan model didefinisikan untuk menentukan apakah terdapat perjalanan langsung dari simpul i ke simpul j atau tidak. Variabel keputusan ini didefinisikan sebagai berikut:

$$x_{ij} = \begin{cases} 1, & \text{Jika simpul } i \text{ dan simpul } j \text{ saling bertetangga} \\ 0, & \text{lainnya.} \end{cases}$$

Fungsi tujuan model didefinisikan untuk meminimumkan total waktu evakuasi dari setiap simpul sumber ke simpul tujuan. Fungsi tujuan ini diekspresikan sebagai berikut:

Meminimumkan:

$$z = \sum_{i \in V} \sum_{j \in V} w_{ij} x_{ij}.$$

Kendala-kendala dari model optimisasi ini adalah sebagai berikut:

1. Setiap simpul i hanya dapat menjadi simpul asal paling banyak satu kali. Kendala tersebut diekspresikan sebagai berikut:

$$\sum_{j \in V} x_{ij} \leq 1 \text{ untuk setiap } i \in V.$$

2. Setiap simpul j hanya dapat dikunjungi paling banyak satu kali. Kendala tersebut diekspresikan sebagai berikut:

$$\sum_{i \in V} x_{ij} \leq 1 \text{ untuk setiap } j \in V.$$

3. Setiap jalur evakuasi berawal dari simpul sumber. Kendala tersebut diekspresikan sebagai berikut:

$$\sum_{j \in V} x_{ij} = 1 \text{ untuk setiap } i \in S.$$

4. Setiap jalur evakuasi berakhir pada sebuah simpul tujuan. Kendala tersebut diekspresikan sebagai berikut:

$$\sum_{i \in V} x_{ij} = 1 \text{ untuk setiap } j \in T.$$

5. Setiap pengungsi yang menuju sebuah simpul (selain di simpul sumber dan tujuan) harus meninggalkan simpul tersebut. Kendala tersebut diekspresikan sebagai berikut:

$$\sum_{i \in V} x_{il} - \sum_{j \in V} x_{lj} = 0 \text{ untuk setiap } l \in (V - (S \cup T)).$$

Model optimisasi di atas termasuk dalam kategori model *binary integer programming*. Pada bagian selanjutnya dari sub bab ini akan dibahas teknik penyelesaian model di atas.

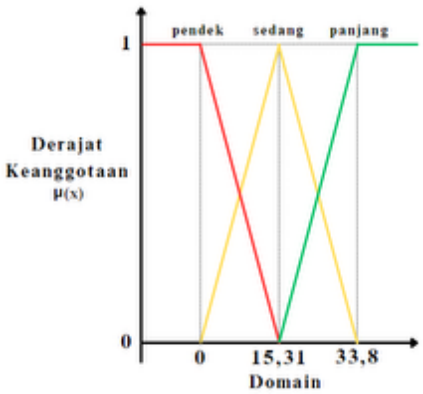
2.3 Teknik Penyelesaian Model

Jalur evakuasi tercepat ditentukan menggunakan Algoritma Dijkstra. Logika *fuzzy* digunakan untuk menentukan bobot sisi yang merupakan representasi dari panjang jalan, lebar jalan, dan jumlah pengungsi tidak pasti.

2.3.1 Penentuan Bobot Sisi Menggunakan Logika *Fuzzy*

Logika *fuzzy* digunakan untuk mengkonversi atau mengubah nilai dari berbagai macam parameter pada jalur evakuasi menjadi sebuah nilai saja. Logika *fuzzy* beroperasi dengan mengambil data *input* berupa nilai pasti, mengonversinya menjadi bentuk kabur, mengevaluasi aturan *fuzzy*, menggabungkan hasilnya, dan kemudian mengonversi hasil tersebut kembali menjadi nilai pasti untuk pengambilan keputusan. Menurut Keprate & Ratnayake (2016), secara umum proses dalam logika *fuzzy* terdiri dari, fuzzifikasi, inferensi *fuzzy*, dan tahap defuzzifikasi.

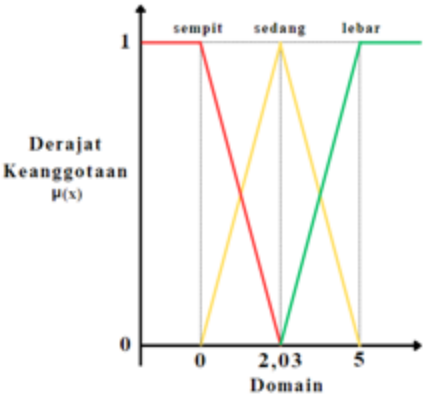
1. Fuzzifikasi
   Pada tahap ini dilakukan penentuan variabel *fuzzy*, himpunan *fuzzy*, semesta pembicaraan, domain, fungsi keanggotaan, dan aturan *fuzzy*. Pada tahap ini, diambil data *input* berupa nilai pasti, yang pada tahap selanjutnya akan dikonversi menjadi bentuk kabur. Pada penelitian ini, variabel *input* terdiri dari panjang jalan, lebar jalan, serta jumlah pengungsi dan variabel *output* berupa indeks waktu evakuasi. Berikut adalah penjelasan rinci dari tahapan di atas.
   a. Panjang Jalan
      Peneliti mengkategorikan panjang jalan menjadi 3 himpunan *fuzzy,* yaitu pendek, sedang, dan panjang. Kurva yang merepresentasikan fungsi keanggotaan dari variabel panjang jalan ditampilkan pada Gambar 1.



Gambar 1. Kurva Fungsi Keanggotaan Panjang Jalan

b. Lebar Jalan

Peneliti mengkategorikan lebar jalan menjadi 3 himpunan *fuzzy,* yaitu sempit, sedang, dan lebar. Kurva yang merepresentasikan fungsi keanggotaan dari variabel lebar jalan ditampilkan pada Gambar 2.



Gambar 2. Kurva Fungsi Keanggotaan Lebar Jalan

c. Jumlah Pengungsi

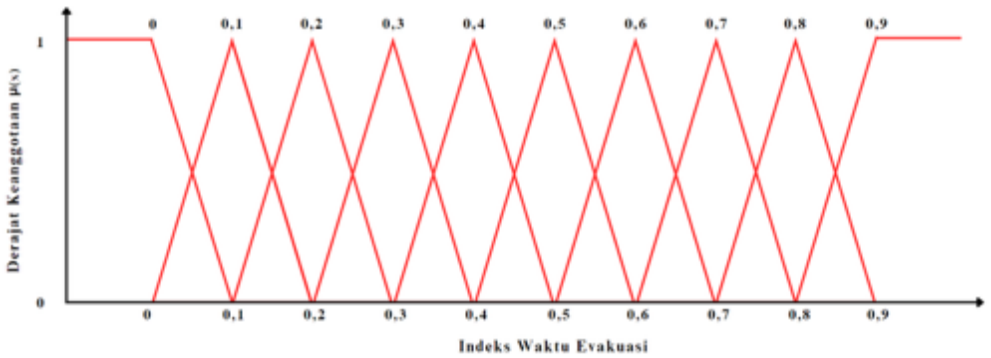
Peneliti mengkategorikan jumlah pengungsi menjadi 3 himpunan *fuzzy,* yaitu sedikit, sedang, dan banyak. Kurva yang merepresentasikan fungsi keanggotaan dari variabel jumlah pengungsi disajikan pada Gambar 3.



Gambar 3. Kurva Fungsi Keanggotaan Jumlah Pengungsi

d. Indeks Waktu Evakuasi

Peneliti membagi indeks waktu evakuasi menjadi 10 himpunan *fuzzy,* yaitu 0, 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, dan 0,9. Kurva yang merepresentasikan fungsi keanggotaan dari variabel indeks waktu evakuasi ditampilkan pada Gambar 4.



Gambar 4. Kurva Fungsi Keanggotaan Indeks Waktu Evakuasi

e. Aturan *Fuzzy*

Aturan *fuzzy* dibuat untuk menyatakan relasi antara *input* dan *output*. Fungsi implikasi *IF-THEN* digunakan sebagai metode pemetaan antara *input* dan *output*, sementara operator *AND* digunakan untuk menghubungkan dua variabel *input*. Tabel 1. menjelaskan aturan *fuzzy* yang digunakan dalam penelitian ini.

Tabel 1. Aturan *Fuzzy*

Aturan (*i*)		**Panjang Jalan**		**Lebar Jalan**		**Jumlah Pengungsi**		**Indeks Waktu Evakuasi**
1		Pendek		Lebar		Sedikit		0
2		Pendek		Lebar		Sedang		0,1
3		Pendek		Lebar		Banyak		0,2
4		Pendek		Sedang		Sedikit		0,1
5		Pendek		Sedang		Sedang		0,2
6		Pendek		Sedang		Banyak		0,3
7		Pendek		Sempit		Sedikit		0,2
8		Pendek		Sempit		Sedang		0,3
9		Pendek		Sempit		Banyak		0,4
10		Sedang		Lebar		Sedikit		0,2
11		Sedang		Lebar		Sedang		0,3
12		Sedang		Lebar		Banyak		0,4
13	**IF**	Sedang	**AND**	Sedang	**AND**	Sedikit	**THEN**	0,3
14		Sedang		Sedang		Sedang		0,4
15		Sedang		Sedang		Banyak		0,5
16		Sedang		Sempit		Sedikit		0,4
17		Sedang		Sempit		Sedang		0,5
18		Sedang		Sempit		Banyak		0,6
19		Panjang		Lebar		Sedikit		0,4
20		Panjang		Lebar		Sedang		0,5
21		Panjang		Lebar		Banyak		0,7
22		Panjang		Sedang		Sedikit		0,5
23		Panjang		Sedang		Sedang		0,6
24		Panjang		Sedang		Banyak		0,8
25		Panjang		Sempit		Sedikit		0,6
26		Panjang		Sempit		Sedang		0,7
27		Panjang		Sempit		Banyak		0,9

2. Inferensi *Fuzzy*
   Tahap ini merupakan proses penalaran menggunakan *fuzzy input* dan aturan *fuzzy* yang telah ditentukan sehingga menghasilkan *fuzzy output.* Pengonversian ini melibatkan himpunan *fuzzy* dan aturan *fuzzy*. Sistem inferensi *fuzzy* yang digunakan dalam penelitian ini adalah Metode Sugeno orde-nol, dimana penalaran menggunakan fungsi min sebagai fungsi implikasi. Berikut adalah perhitungan fungsi min sebagai fungsi implikasi:

$$\begin{aligned}\alpha - \text{predikat}_i &= \mu_{\text{Pendek}}[x] \cap \mu_{\text{Lebar}}[y] \cap \mu_{\text{Sedikit}}[z] \\ &= \min\left(\mu_{\text{Pendek}}[x], \mu_{\text{Lebar}}[y], \mu_{\text{Sedikit}}[z]\right)\end{aligned}$$

3. Defuzzifikasi
   Tahap ini merupakan proses merubah *fuzzy output* menjadi nilai yang bersifat pasti (Iskierka & Lipiec, 2019). Metode defuzzifikasi yang digunakan dalam Metode Sugeno dilakukan dengan cara mencari nilai rata-rata dari *fuzzy output* pada setiap aturan-i yang memiliki bobot relatif berupa hasil dari operasi fungsi implikasi. Berikut adalah perhitungan defuzzifikasi pada Metode Sugeno:

$$z = \frac{(\alpha - \text{pred}_1) \times z_1 + (\alpha - \text{pred}_2) \times z_2 + \cdots + (\alpha - \text{pred}_n) \times z_n}{(\alpha - \text{pred}_1) + (\alpha - \text{pred}_2) + \cdots + (\alpha - \text{pred}_n)}$$

   di mana:
   z : hasil defuzzifikasi yang merupakan nilai indeks waktu evakuasi pada jalan.
   z_i : *output fuzzy* atau nilai dari konsekuen pada aturan ke-i.
   $\alpha - \text{pred}_\text{i}$: hasil operasi fungsi implikasi pada aturan ke-i.

2.3.2 Penentuan Jalur Evakuasi Menggunakan Algoritma Dijkstra

Algoritma Dijkstra merupakan sebuah algoritma yang dapat menyelesaikan permasalahan pencarian jarak terpendek antar simpul pada sebuah graf yang memiliki bobot non-negatif (Iordan, 2012). Berikut adalah langkah-langkah perhitungan algoritma Dijsktra menggunakan aplikasi *Microsoft Excel*:

1. Bangun tabel di mana baris merepresentasikan kumpulan simpul yang akan dituju, dan setiap kolom merepresentasikan simpul inisial yang akan digunakan untuk mengevaluasi indeks waktu evakuasi.
2. Pilih simpul inisial pertama yaitu simpul sumber pada graf berbobot.
3. Catat indeks waktu evakuasi pada kolom dan baris yang sesuai dengan format '$X|Y$' di mana X adalah indeks waktu evakuasi dan Y adalah simpul inisial. Jika terdapat indeks waktu evakuasi baru, pilih waktu tempuh yang paling kecil.
4. Pilih simpul inisial baru. Jika simpul inisial sebelumnya berajasen dengan lebih dari satu simpul, pilih simpul yang memiliki waktu tempuh terkecil terlebih dahulu. Jika tidak, simpul yang dituju oleh simpul inisial sebelumnya dapat langsung dipilih sebagai simpul inisial baru.
5. Ulangi Langkah 3 hingga simpul tujuan telah dikunjungi. Jika simpul tujuan telah dikunjungi, hentikan algoritma. Pemilihan lintasan dilakukan dengan mempertimbangkan simpul pendahulu dari masing-masing simpul inisial.

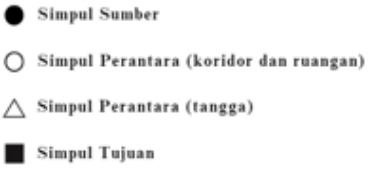
3. HASIL DAN PEMBAHASAN

Data yang diperlukan pada penelitian ini mencakup denah bangunan Gedung FPMIPA A UPI, panjang jalan dan lebar jalan koridor Gedung FPMIPA A UPI, serta kapasitas maksimal untuk setiap ruangan di Gedung FPMIPA A UPI. Denah gedung bangunan dan kapasitas maksimal ruang Gedung FPMIPA A UPI diperoleh dengan bantuan pihak akademisi.

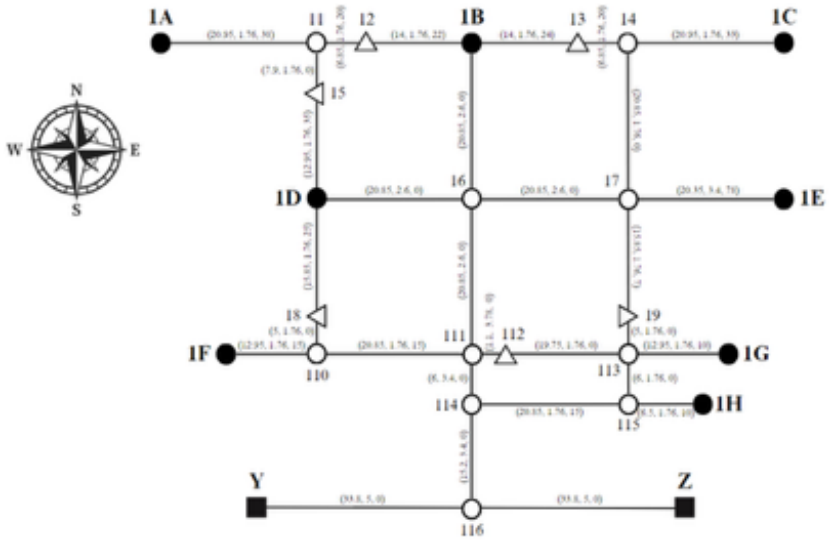
Sedangkan, panjang jalan dan lebar jalan koridor Gedung FPMIPA A UPI diperoleh dengan pengukuran manual menggunakan alat ukur meteran.

3.1 Representasi Graf

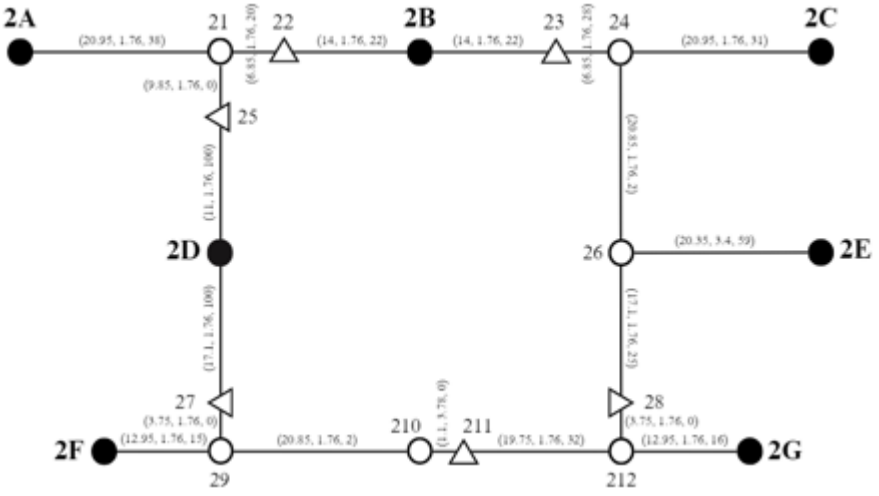
Pada penelitian ini, representasi graf berbobot dari struktur bangunan Gedung FPMIPA A UPI diwakili 77 simpul yang terdiri dari 29 simpul sumber, 46 simpul perantara, dan 2 simpul tujuan, dan 95 sisi. Gambar 5 mengilustrasikan definisi simbol-simbol yang digunakan dalam representasi graf berbobot. Representasi graf berbobot (bobot graf merupakan nilai dari pajang jalan, lebar jalan, dan jumlah pengungsi secara berturut-turut) dari Gedung FPMIPA A UPI dapat disajikan pada Gambar 6 sampai Gambar 10.



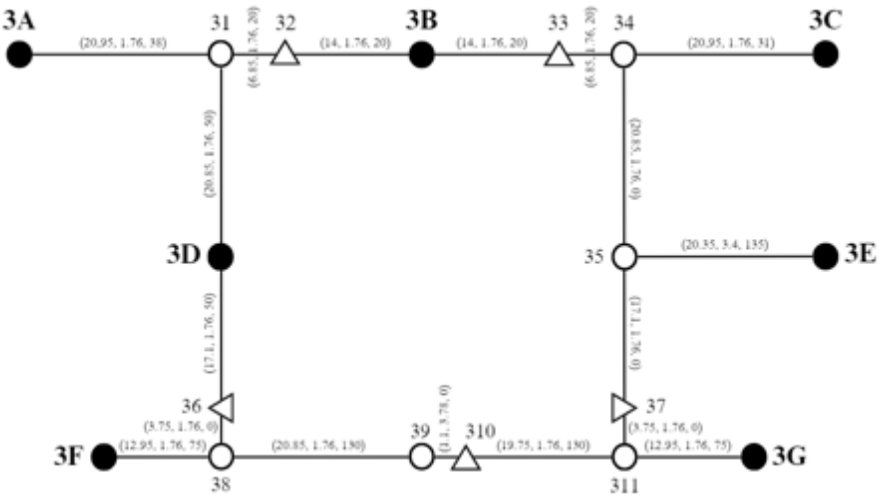
Gambar 5. Definisi Simbol-Simbol dalam Representasi Graf



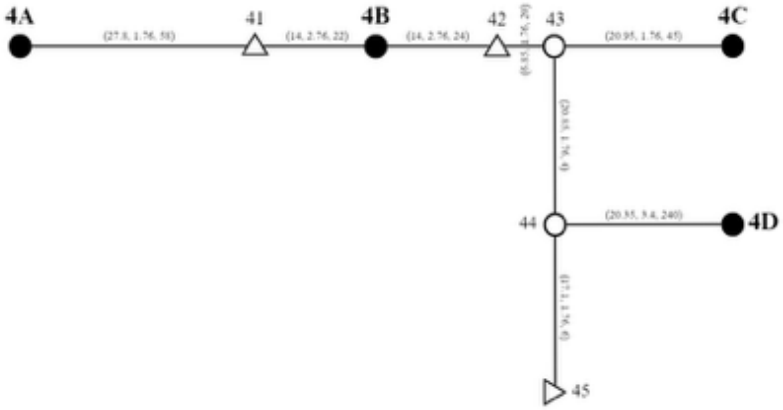
Gambar 6. Representasi Graf Berbobot Lantai 1 Gedung FPMIPA A UPI.



Gambar 7. Representasi Graf Berbobot Lantai 2 Gedung FPMIPA A UPI.



Gambar 8. Representasi Graf Berbobot Lantai 3 Gedung FPMIPA A UPI.



Gambar 9. Representasi Graf Berbobot Lantai 4 Gedung FPMIPA A UPI.



Gambar 10. Representasi Graf Berbobot Lantai 5 Gedung FPMIPA A UPI.

3.2 Penentuan Bobot Graf Menggunakan Logika *Fuzzy*

Untuk mengonversi nilai dari parameter panjang jalan, lebar jalan, dan jumlah pengungsi akan digunakan Metode *Fuzzy* Sugeno dengan bantuan aplikasi Matlab R2022a pada fitur *Fuzzy* Logic Toolbox. Himpunan *fuzzy* yang digunakan pada penelitian ini dapat dilihat pada Tabel 2.

Tabel 2. Himpunan *Fuzzy*

Variabel	Himpunan	Semesta Pembicaraan	Domain	Fungsi Keanggotaan	Parameter
Panjang Jalan	Pendek	[0, 33,8]	[0, 15, 31]	Bahu	[−15,31 0 15,31]
	Sedang		[0, 33,8]		[0 15,31 33,8]
	Panjang		[15,31, 33,8]		[15,31 33,8 52,29]
	Sempit	[0, 5]	[0, 2,03]	Bahu	[−2,03 0 2,03]

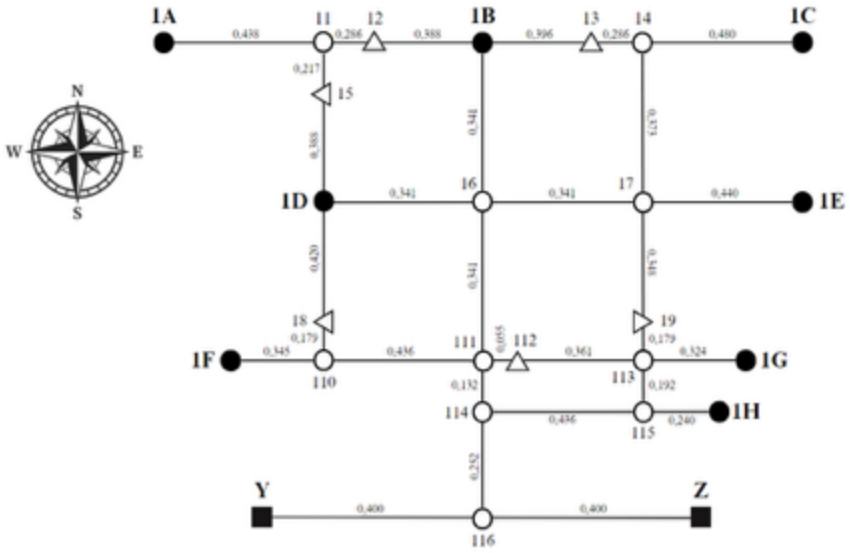
Variabel	Himpunan	Semesta Pembicaraan	Domain	Fungsi Keanggotaan	Parameter
Lebar Jalan	Sedang		[0, 5]		[0 2,03 5]
	Lebar		[2,03 ,5]		[2,03 5 7,97]
Jumlah Pengungsi	Sedikit	[0, 240]	[0, 24]	Bahu	[−24 0 24]
	Sedang		[0, 240]		[0 24 240]
	Banyak		[24, 240]		[24 240 456]
Indeks Waktu Evakuasi	0	[0, 0.9]	[0, 0,1]	Bahu	[0]
	0,1		[0, 0,2]		[0,1]
	0,2		[0,1, 0,3]		[0,2]
	0,3		[0,2, 0,4]		[0,3]
	0,4		[0,3, 0,5]		[0,4]
	0,5		[0,4, 0,6]		[0,5]
	0,6		[0,5, 0,7]		[0,6]
	0,7		[0,6, 0,8]		[0,7]
	0,8		[0,7, 0,9]		[0,8]
	0,9		[0,8, 0,9]		[0,9]

Hasil akhir perhitungan indeks waktu evakuasi menggunakan aplikasi Matlab R2022a untuk seluruh sisi graf berbobot yang terdapat pada Gambar 6 sampai Gambar 10 disajikan dalam Tabel 3. Selanjutnya representasi graf berbobot dari Gedung FPMIPA A UPI dapat diperbarui menjadi graf dengan bobot tunggal berupa indeks waktu evakuasi yang dapat dilihat pada Gambar 11 sampai Gambar 15.

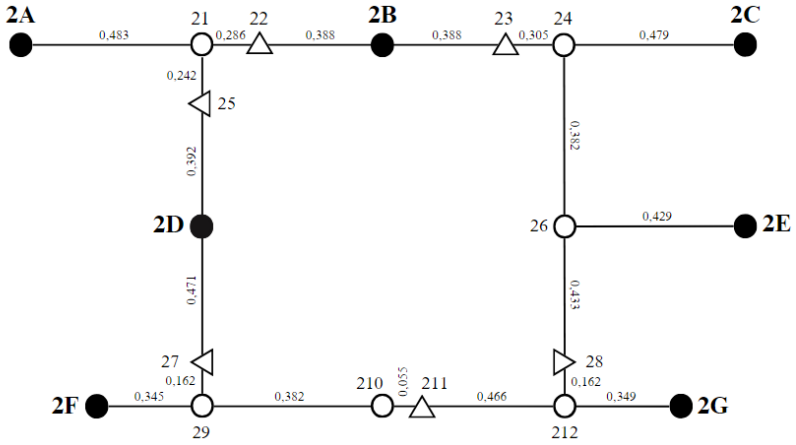
Tabel 3. Hasil Perhitungan Indeks Waktu Evakuasi.

Sisi	Indeks Waktu Evakuasi	Sisi	Indeks Waktu Evakuasi	Sisi	Indeks Waktu Evakuasi
1A – 11	0,483	25 – 2D	0,392	2G – 212	0,349
1C – 14	0,480	2D – 27	0,471	3F – 38	0,406
2A – 21	0,483	26 – 28	0,433	3G – 311	0,406
2C – 24	0,479	3D – 36	0,446	112 – 111	0,055
3A – 31	0,483	35 – 37	0,333	211 – 210	0,055
3C – 34	0,479	44 – 45	0,358	310 – 39	0,055
4C – 43	0,487	110 – 111	0,436	113 – 112	0,361
4A – 41	0,575	14 – 17	0,373	212 – 211	0,466
5A – 51	0,575	115 – 114	0,436	311 – 310	0,522
5C – 52	0,579	29 – 210	0,382	113 – 115	0,192
11 – 12	0,286	24 – 26	0,382	111 – 114	0,132
14 – 13	0,286	38 – 39	0,537	1H – 115	0,240
21 – 22	0,286	31 – 3D	0,489	114 – 116	0,252
24 – 23	0,305	34 – 35	0,373	116 – Y	0,4
31 – 32	0,286	43 – 44	0,390	116 – Z	0,4
34 – 33	0,286	1B – 16	0,341	51 – 41	0,360
43 – 42	0,286	1D – 16	0,341	52 – 42	0,360
1B – 12	0,388	17 – 16	0,341	41 – 32	0,360
1B – 13	0,396	16 – 111	0,341	42 – 33	0,360
2B – 22	0,388	18 – 110	0,179	32 – 25	0,360

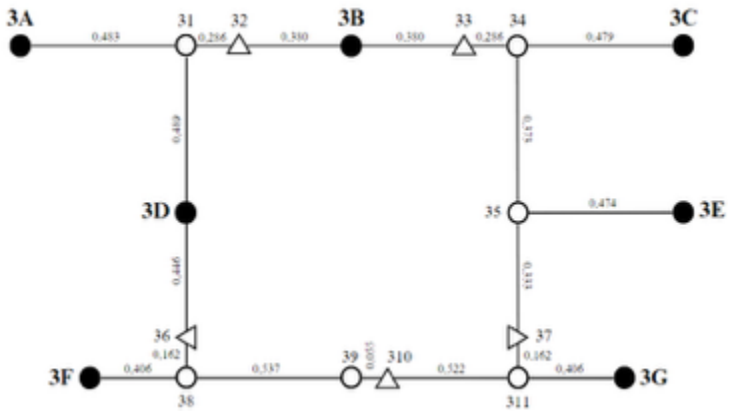
2B – 23	0,388	19 – 113	0,179	33 – 23	0,360
3B – 32	0,380	27 – 29	0,162	22 – 12	0,360
3B – 33	0,380	28 – 212	0,162	23 – 13	0,360
4B – 41	0,388	36 – 38	0,162	45 – 37	0,270
4B – 42	0,396	37 – 311	0,162	37 – 28	0,270
5B – 51	0,388	1E – 17	0,440	36 – 27	0,270
5B – 52	0,388	2E – 26	0,429	28 – 19	0,270
11 – 15	0,217	3E – 35	0,474	27 – 18	0,270
21 – 25	0,242	4D - 44	0,536	310 – 211	0,270
15 – 1D	0,388	1F – 110	0,345	211 – 112	0,270
1D – 18	0,420	1G – 113	0,324	25 – 15	0,339
17 – 19	0,348	2F – 29	0,345		



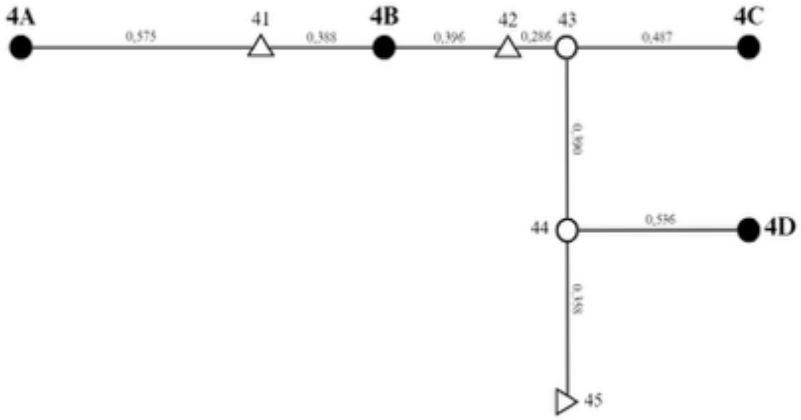
Gambar 11. Graf Berbobot Indeks Waktu Evakuasi Lantai 1 Gedung FPMIPA A UPI



Gambar 12. Graf Berbobot Indeks Waktu Evakuasi Lantai 2 Gedung FPMIPA A UPI



Gambar 13. Graf Berbobot Indeks Waktu Evakuasi Lantai 3 Gedung FPMIPA A UPI



Gambar 14. Graf Berbobot Indeks Waktu Evakuasi Lantai 4 Gedung FPMIPA A UPI



Gambar 15. Graf Berbobot Indeks Waktu Evakuasi Lantai 5 Gedung FPMIPA A UPI

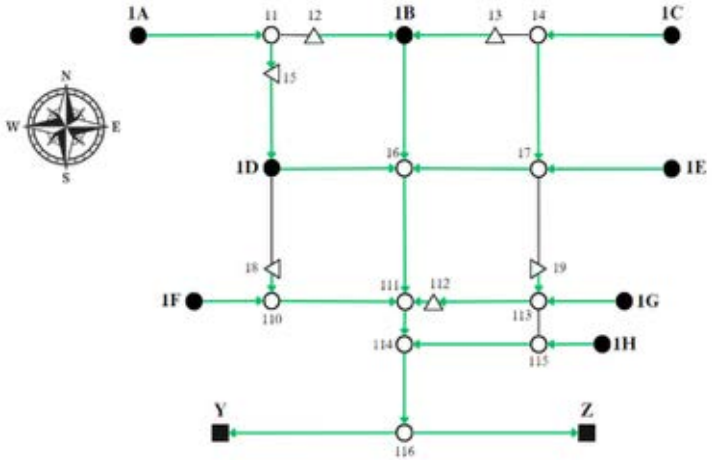
3.3 Penentuan Jalur Evakuasi Menggunakan Algoritma Dijkstra

Setelah graf berbobot diperoleh, maka tahapan selanjutnya adalah mengaplikasikan Algoritma Dijkstra untuk mencari jalur terpendek dari setiap simpul sumber ke simpul tujuan. Hasil dari perhitungan jalur evakuasi optimal untuk masing-masing titik evakuasi menggunakan Algoritma Dijkstra dengan aplikasi *Microsoft Excel*. Ilustrasi rute dari jalur evakuasi pada Gedung FPMIPA A UPI dapat dilihat pada Gambar 16 sampai Gambar 19.

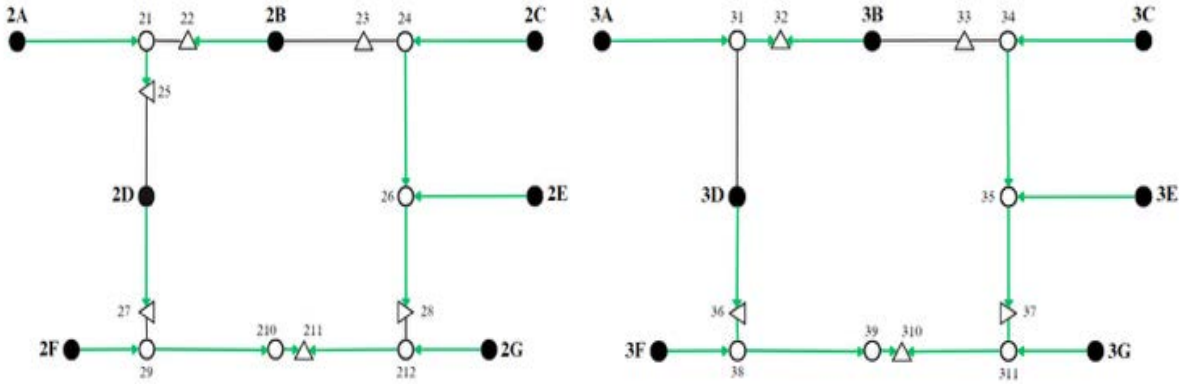
3.4 Analisi Hasil Implementasi

Jalur evakuasi optimal dituliskan pada Tabel 4. Jalur evakuasi dengan waktu evakuasi tercepat adalah jalur evakuasi dari sudut Ruang Gudang ke titik kumpul bagian Tenggara, sedangkan jalur evakuasi dengan waktu evakuasi terlama adalah jalur evakuasi dari toilet Laki-Laki bagian Timur Laut Lantai 5 Sayap *North* ke titik kumpul bagian Barat Daya. Dengan demikian, waktu evakuasi minimum yang dibutuhkan untuk mengevakuasi Gedung FPMIPA A UPI adalah ketika proses evakuasi dari toilet Laki-laki bagian Timur Laut Lantai 5 Sayap *North* ke titik kumpul bagian Barat Daya sudah selesai. Lebih jauh, hasil penelitian juga menunjukkan bahwa struktur Gedung FPMIPA A UPI saat ini sudah sangat baik. Hal ini dikarenakan setiap

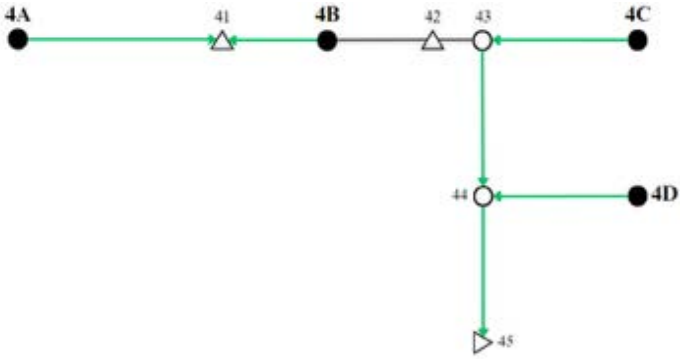
sayap pada setiap lantai memiliki setidaknya satu akses tangga, yang memungkinkan proses evakuasi berlangsung secara efektif dan efisien.



Gambar 16. Rute dari Jalur Evakuasi Gedung FPMIPA A UPI Lantai 1



Gambar 17. Rute dari Jalur Evakuasi Gedung FPMIPA A UPI Lantai 2 dan 3



Gambar 18. Rute dari Jalur Evakuasi Gedung FPMIPA A UPI Lantai 4



Gambar 19. Rute dari Jalur Evakuasi Gedung FPMIPA A UPI Lantai 5

Tabel 4. Total Indeks Waktu Evakuasi dan Rute Seluruh Jalur Evakuasi

Jalur Evakuasi	Indeks Waktu Evakuasi	Jalur Evakuasi	Indeks Waktu Evakuasi	Jalur Evakuasi	Indeks Waktu Evakuasi
1B – Y	1,466	2E – Z	2,511	3C – Y	3,104
1H – Z	1,328	2F – Z	1,891	3A – Y	3,343
1G – Z	1,524	2C – Y	2,943	4B – Y	3,322
1E – Z	1,906	2D – Y	2,140	4D – Z	3,083
1F – Z	1,565	2A – Y	2,918	4C – Y	3,424
1D – Y	1,466	3B – Y	2,954	4A – Y	3,509
1C – Y	2,319	3G – Z	2,307	5B – Y	3,682
1A – Y	2,509	3E – Z	2,726	5C – Y	3,881
2B – Y	2,602	3F – Z	2,377	5A – Y	3,869
2G – Z	1,924	3D – Y	2,385		

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, kesimpulan yang dapat diambil adalah sebagai berikut:

1. Masalah penentuan jalur evakuasi dapat dimodelkan sebagai model *binary integer programming* dari masalah penentuan rute terpendek. Fungsi tujuan dari model adalah untuk meminimumkan total indeks waktu evakuasi dengan kendala bahwa lintasan harus berawal dari setiap simpul sumber (titik-titik evakuasi) dan berakhir pada sebuah simpul tujuan (titik kumpul).
2. Jalur evakuasi yang optimal pada sebuah gedung dapat dicari dengan melibatkan beberapa parameter seperti panjang jalan, lebar jalan, dan jumlah pengungsi, dengan cara mengubah nilai-nilai dari beberapa parameter tersebut menjadi sebuah nilai menggunakan logika *fuzzy* yang selanjutnya menjadi nilai bobot setiap busur.
3. Hasil implementasi menggunakan Algoritma *Fuzzy* Dijkstra pada masalah penentuan jalur evakuasi di Gedung FPMIPA A UPI menunjukan bahwa penggunaan Algoritma *Fuzzy* Dijkstra telah berhasil menentukan jalur evakuasi yang optimal pada Gedung FPMIPA A UPI, yaitu jalur evakuasi tercepat berdasarkan pertimbangan panjang jalan, lebar jalan, dan jumlah pengungsi. Dengan kata lain, jalur evakuasi yang mempertimbangkan jarak tempuh dan kepadatan pengungsi.

5. DAFTAR PUSTAKA

Abdillah, H. N., Rakhmatsyah, A., & Putrada, A. G. (2019). Implementasi *fuzzy* dan Dijkstra pada sistem pengangkutan sampah. *Jurnal Edukasi dan Penelitian Informatika* (*JEPIN*), *5*(3), 286-293.

Afandi, N., & Mayasari, Z. M. (2021, March). An evacuation route in Bengkulu City based on *Fuzzy* Dijkstra Algorithm. In *Journal of Physics: Conference Series* (Vol. 1863, No. 1, p. 012007). IOP Publishing.

Al Hakim, R. R., Satria, M. H., Arief, Y. Z., Pangestu, A., Jaenul, A., Hertin, R. D., & Nugraha, D. (2021). Aplikasi Algoritma Dijkstra dalam Penyelesaian Berbagai Masalah. *Expert*, *11*(1), 345994.

Andhika, P. K., Kasim, F., & Hawibowo, S. (2013). Optimasi proses evakuasi dalam menghadapi situasi darurat pada Gedung Graha Sabha Pramana (studi kasus acara wisuda). *Teknofisika*, *2*(2), 35-41.

Deng, Y., Chen, Y., Zhang, Y., & Mahadevan, S. (2012). *Fuzzy* Dijkstra Algorithm for shortest path problem under uncertain environment. *Applied Soft Computing*, *12*(3), 1231-1237.

Ichsan, M. H. H., Yudaningtyas, E., & Muslim, M. A. (2012). Solusi optimal pencarian jalur tercepat dengan Algoritma Hybrid *Fuzzy* Dijkstra. *Jurnal EECCIS (Electrics, Electronics, Communications, Controls, Informatics, Sistems)*, *6*(2), 155-160.

Iskierka, J., & Lipiec, S. (2019). Determining optimal route using fuzzy logic and Dijkstra's algorithm. In G. Capizzi, R. Damaševičius, D. Połap, & M. Woźniak (Eds.), Proceedings of the Symposium for Young Scientists in Technology, Engineering and Mathematics (SYSTEM 2019) (CEUR Workshop Proceedings, Vol. 2468, pp. 10–15). CEUR-WS.

Iordan, A. E. (2012). Development of an interactive environment used for simulation of shortest paths algorithms. *Annals of The Faculty of Engineering Hunedoara*, 10(3), 97-102.

Keprate, A., & Ratnayake, R. C. (2016). A *fuzzy* logic-based approach for deciding the corrective action to minimize vibration induced fatigue damage on offshore pipework. In *2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)* (pp. 1370-1374). IEEE.

Mayasari, Z. M., & Afandi, N. (2021). Optimasi jalur evakuasi bagi pejalan kaki menggunakan Algoritma *Fuzzy* Dijkstra di Kecamatan Teluk Segara, Bengkulu. *BAREKENG: Jurnal Ilmu Matematika dan Terapan*, *15*(3), 581-590.

Nggufron, N., Rochmad, R., & Mashuri, M. (2019). Pencarian rute terbaik pemadam kebakaran kota semarang menggunakan Algoritma Dijkstra dengan logika *fuzzy* sebagai penentu bobot pada graf. *Unnes Journal of mathematics*, *8*(1), 40-49.

Novianingsih, K., Ilham, B., Yulianti, K., & Gozali, S. M. (2024). Planning evacuation routes in rise buildings using maximum dynamic flow approach. In *Proceedings of the 9th Mathematics, Science, and Computer Science Education International Seminar* (*MSCEIS 2023*) (pp. 142–148). Atlantis Press.

Patel, V., & Bagar, C. (2014). A Survey Paper of Bellman-Ford Algorithm and Dijkstra Algorithm for Finding Shortest Path in GIS Application. *International Journal of P2P Network Trends and Technology*, *5*, 1-4.

Tirastittam, P., & Waiyawuththanapoom, P. (2014). Public transport planning system by Dijkstra algorithm: Case study bangkok metropolitan area. *World Academy of Science, Engineering and Technology International Journal of Social, Behavioral, Educational, Economic, Business and Industrial Engineering*, *8*(1), 54-59.