

**IMPLEMENTASI ALGORITMA A-STAR DALAM
MENENTUKAN RUTE TERPENDEK ANTAR RUMAH SAKIT
DI BALI**

SKRIPSI

**OLEH:
MAHMUDA
NIM. 210601110008**



**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2025**

**IMPLEMENTASI ALGORITMA A-STAR DALAM
MENENTUKAN RUTE TERPENDEK ANTAR RUMAH SAKIT
DI BALI**

SKRIPSI

**Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Matematika (S.Mat)**

**Oleh
MAHMUDA
NIM. 210601110008**

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2025**

IMPLEMENTASI ALGORITMA A-STAR DALAM MENENTUKAN RUTE TERPENDEK ANTAR RUMAH SAKIT DI BALI

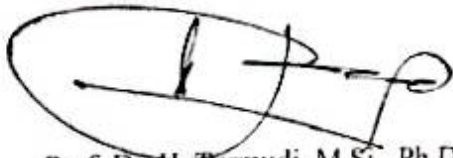
SKRIPSI

Oleh
Mahmuda
NIM. 210601110008

Telah Disetujui Untuk Diuji

Malang, 2 Juni 2025

Dosen Pembimbing I



Prof. Dr. H. Turmudi, M.Si., Ph.D
NIP. 19571005 198203 1 006

Dosen Pembimbing II



Dr. Fachrud Rozi, M.Si
NIP. 19800527 200801 1 012

Mengetahui,

Ketua Program Studi Matematika




Dr. Elly Susanti, S.Pd., M.Sc
NIP. 19741129 200012 2 005

IMPLEMENTASI ALGORITMA A-STAR DALAM MENENTUKAN RUTE TERPENDEK ANTAR RUMAH SAKIT DI BALI

SKRIPSI

Oleh
Mahmuda
NIM. 210601110008

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal 20 Juni 2025

Ketua Penguji : Hisyam Fahmi, M.Kom

Anggota Penguji 1 : Mohammad Nafie Jauhari, M.Si

Anggota Penguji 2 : Prof. Dr. H. Turmudi, M.Si., Ph.D

Anggota Penguji 3 : Dr. Fachrur Rozi, M.Si

Mengetahui,
Ketua Program Studi Matematika



Dr. Elly Susanti, M.Sc

NIP. 19741129 200012 2 005

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini

Nama : Mahmuda

NIM : 210601110008

Program Studi : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Implementasi Algoritma A-Star dalam Menentukan Rute
Tependek Antar Rumah Sakit di Bali

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini merupakan hasil karya sendiri, bukan pengambilan tulisan atau pemikiran orang lain yang saya akui sebagai pemikiran saya, kecuali dengan mencantumkan sumber cuplikan pada daftar rujukan di halaman terakhir. Apabila dikemudian hari terbukti skripsi ini adalah jiplakan atau tiruan, maka saya bersedia menerima sanksi yang berlaku atas perbuatan tersebut.

Malang, 20 Juni 2025



Mahmuda
NIM. 210601110008

MOTTO

فَاصْبِرْ إِنَّ وَعْدَ اللَّهِ حَقٌّ

“Dan bersabarlah kamu, sesungguhnya janji Allah adalah benar “

~ QS. AR-Rum [30]: 60

PERSEMBAHAN

Bismillahirrahmanirrahim,

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala limpahan rahmat, nikmat, pertolongan, serta kemudahan yang diberikan dalam setiap proses kehidupan ini. Dengan penuh rasa syukur dan cinta, karya sederhana ini penulis persembahkan kepada kedua orang tua tercinta, Bapak Adi dan Ibu Sati, atas doa, kasih sayang, kesabaran, dan dukungan yang menjadi sumber kekuatan dalam setiap langkah, kepada adik tersayang, Amira, yang selalu menjadi penyemangat dan sumber keceriaan dalam hari-hari penulis, kepada kerabat, sahabat, dan teman-teman seperjuangan yang telah menghadirkan tawa, semangat, serta kebersamaan di sepanjang perjalanan ini, serta untuk diri sendiri, sebagai bentuk penghargaan atas keteguhan hati dalam bertahan, berjuang, dan tetap berharap hanya kepada Allah SWT dalam setiap ujian dan proses yang dijalani. Semoga segala usaha ini membawa kebaikan, keberkahan, dan menjadi awal dari perjalanan panjang yang diridhai Allah SWT.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas segala limpahan rahmat, karunia, dan petunjuk-Nya. sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Algoritma *A-Star* dalam Menentukan Rute Terpendek Antar Rumah Sakit di Bali” dengan lancar. Skripsi ini merupakan bagian dari pemenuhan syarat akademik untuk menyelesaikan studi di Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Penulis menyadari bahwa penyusunan skripsi ini dapat disusun dan diselesaikan dengan baik karena atas dukungan, bantuan, motivasi dari berbagai pihak. Oleh karena itu, dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Elly Susanti, S.Pd., M.Sc, selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. H. Turmudi, M.Si., Ph.D, selaku dosen pembimbing I yang dengan penuh kesabaran dan dedikasi telah meluangkan waktu, tenaga, serta memberikan bimbingan, arahan, dan masukan yang sangat berharga kepada penulis.
5. Dr. Fachrur Rozi, M.Si, selaku dosen pembimbing II yang dengan penuh kesabaran dan dedikasi telah meluangkan waktu, tenaga, serta memberikan bimbingan, arahan, dan masukan yang sangat berharga kepada penulis.
6. Hisyam Fahmi, M.Kom, selaku dosen ketua penguji yang dengan penuh kesabaran dan dedikasi telah meluangkan waktu, tenaga, serta memberikan bimbingan, arahan, dan masukan yang sangat berharga kepada penulis.
7. Mohammad Nafie Jauhari, M.Si, selaku anggota penguji 1 yang dengan penuh kesabaran dan dedikasi telah meluangkan waktu, tenaga, serta memberikan bimbingan, arahan, dan masukan yang sangat berharga kepada penulis.

8. Terima Kasih kepada Dosen Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah memberikan ilmu yang bermanfaat dan nasehat selama perkuliahan.
9. Terkhusus, kepada Ibu Sati dan Ayah Adi atas doa, motivasi, dan dukungan semangat yang tiada henti. Cinta, kasih sayang, dan pengorbanan mereka menjadi sumber kekuatan dalam setiap langkah saya.
10. Terima kasih kepada teman kelompok PKL yang telah memberi dukungan, doa, dan semangat selama masa perkuliahan.
11. Terima kasih kepada teman, sahabat atas segala dukungan, doa dan kebersamaannya selama di perantauan.
12. Terima kasih kepada teman angkatan Matematika 2021 (TEOREMA) yang telah menjadi teman saat masa perkuliahan.
13. Terima kasih kepada diri sendiri yang telah berusaha dan pantang menyerah dalam pembuatan skripsi.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, sehingga kritik dan saran sangat diharapkan untuk perbaikan di masa yang akan datang. Penulis berharap, semoga skripsi ini dapat memberikan manfaat tidak hanya bagi penulis sendiri, tetapi juga bagi pengembangan ilmu pengetahuan, khususnya di bidang Matematika. Akhir kata, semoga Allah SWT senantiasa memberikan rahmat, petunjuk, dan ridha-Nya kepada semua pihak yang telah berkontribusi dalam penyusunan skripsi ini serta memberikan kemudahan dan kelancaran dalam proses penelitian yang akan dilaksanakan.

Malang, 20 Juni 2025

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiv
DAFTAR LAMPIRAN	xv
ABSTRAK	xvi
ABSTRACT	xvii
ملخص البحث	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Tujuan Penelitian	5
1.4 Manfaat Penelitian	6
1.5 Batasan Masalah	7
BAB II KAJIAN TEORI	9
2.1 Teori Pendukung	9
2.1.1 Teori Graf	9
2.1.2 Algoritma A-Star	13
2.1.3 Fungsi Heuristik	17
2.1.4 Rute Terpendek	22
2.1.5 Rumah Sakit di Bali	23
2.2 Menjaga Keselamatan dalam Ajaran Islam	24
2.3 Kajian Topik Dengan Teori Pendukung	27
BAB III METODE PENELITIAN	30
3.1 Jenis Penelitian	30
3.2 Data dan Sumber Data	30
3.3 Tahapan Penelitian	31
BAB IV HASIL DAN PEMBAHASAN	36
4.1 Analisis Data	36
4.2 Simpul Awal dan Simpul Tujuan	37
4.3 Perhitungan Fungsi Heuristik dan Implementasi Algoritma A-Star	38
4.4 Analisis Perbandingan Implementasi Algoritma A-Star dengan Menggunakan Metode <i>Haversine</i> dan <i>Vincenty</i>	97
4.5 Integrasi Nilai-nilai Islam dalam Penerapan Algoritma A-Star untuk Penyelamatan Jiwa	99
BAB V PENUTUP	102
5.1 Kesimpulan	102
5.2 Saran	102
DAFTAR PUSTAKA	104

LAMPIRAN.....	107
RIWAYAT HIDUP	113

DAFTAR TABEL

Tabel 4.1	Daftar Pasangan RS Awal dan RS Tujuan di Bali	38
Tabel 4.2	Koordinat Simpul Awal dan Simpul Tujuan	38
Tabel 4.3	RS Awal (RSU Balimed) ke RS Tujuan (RSUD Wangaya).....	40
Tabel 4.4	Titik-titik Persimpangan Jalan (RSU Balimed ke RSUD Wangaya)	40
Tabel 4.5	Nama Jalan Antar Persimpangan (RSU Balimed ke RSUD Wangaya)	40
Tabel 4.6	Data Jarak Aktual ($g(n)$) (RSU Balimed ke RSUD Wangaya)	41
Tabel 4.7	Data Heuristik Menggunakan Formula Haversine (RSU Balimed ke RSUD Wangaya)	44
Tabel 4.8	Perhitungan Konsistensi Heuristik <i>Haversine</i> (RSU Balimed ke RSUD Wangaya)	45
Tabel 4.9	Perhitungan <i>Admissible</i> Heuristik <i>Haversine</i> (RSU Balimed ke RSUD Wangaya)	46
Tabel 4.10	Data Heuristik Menggunakan Formula <i>Vincenty</i> (RSU Balimed ke RSUD Wangaya)	52
Tabel 4.11	Perhitungan Konsistensi Heuristik <i>Vincenty</i> (RSU Balimed ke RSUD Wangaya)	53
Tabel 4.12	Perhitungan <i>Admissible</i> Heuristik <i>Vincenty</i> (RSU Balimed ke RSUD Wangaya)	54
Tabel 4.13	RS Awal (RSU Premagana) ke RS Tujuan (RSD Mangusada).....	63
Tabel 4.14	Titik-titik Persimpangan Jalan (RSU Premagana ke RSD Mangusada).....	63
Tabel 4.15	Nama Jalan Antar Persimpangan (RSU Premagana ke RSD Mangusada).....	64
Tabel 4.16	Data Jarak Aktual ($g(n)$) (RSU Premagana ke RSD Mangusada).....	64
Tabel 4.17	Data Heuristik Menggunakan Formula Haversine (RSU Premagana ke RSD Mangusada)	65
Tabel 4.18	Perhitungan Konsistensi Heuristik <i>Haversine</i> (RSU Premagana ke RSD Mangusada)	66
Tabel 4.19	Perhitungan <i>Admissible</i> Heuristik <i>Haversine</i> (RSU Premagana ke RSD Mangusada)	67
Tabel 4.20	Data Heuristik Menggunakan Formula <i>Vincenty</i> (RSU Premagana ke RSD Mangusada)	68
Tabel 4.21	Perhitungan Konsistensi Heuristik <i>Vincenty</i> (RSU Premagana ke RSD Mangusada)	69
Tabel 4.22	Perhitungan <i>Admissible</i> Heuristik <i>Vincenty</i> (RSU Premagana ke RSD Mangusada)	70
Tabel 4.23	RS Awal (RS TK. II Udayana) ke RS Tujuan (RSUD Bali Mandara).....	80
Tabel 4.24	Titik-titik Persimpangan Jalan (RS TK. II Udayana ke RSUD Bali Mandara)	81
Tabel 4.25	Nama Jalan Antar Persimpangan (RS TK. II Udayana ke RSUD Bali Mandara)	81

Tabel 4.26	Data Jarak Aktual ($g(n)$) (RS TK. II Udayana ke RSUD Bali Mandara).....	82
Tabel 4.27	Data Heuristik Menggunakan Formula <i>Haversine</i> (RS TK. II Udayana ke RSUD Bali Mandara).....	83
Tabel 4.28	Perhitungan Konsistensi Heuristik <i>Haversine</i> (RS TK. II Udayana ke RSUD Bali Mandara).....	84
Tabel 4.29	Perhitungan <i>Admissible</i> Heuristik <i>Haversine</i> (RS TK. II Udayana ke RSUD Bali Mandara).....	85
Tabel 4.30	Data Heuristik Menggunakan Formula <i>Vincenty</i> (RS TK. II Udayana ke RSUD Bali Mandara).....	86
Tabel 4.31	Perhitungan Konsistensi Heuristik <i>Vincenty</i> (RS TK. II Udayana ke RSUD Bali Mandara).....	87
Tabel 4.32	Perhitungan <i>Admissible</i> Heuristik <i>Vincenty</i> (RS TK. II Udayana ke RSUD Bali Mandara).....	88
Tabel 4.33	Rute dan Jarak Hasil Perhitungan A-Star (Metode <i>Haversine</i> dan <i>Vincenty</i>).....	98

DAFTAR GAMBAR

Gambar 2.1	Graf Sederhana	10
Gambar 2.2	Graf Tak Sederhana.....	11
Gambar 2.3	Contoh Graf Berbobot	11
Gambar 2.4	Contoh Graf Tak-Berarah Berbobot.....	12
Gambar 2.5	Contoh Graf Berarah Berbobot.....	12
Gambar 2.6	Graf Berarah	13
Gambar 2.7	<i>Flowchart</i> Algoritma A- <i>Star</i>	16
Gambar 4.1	<i>Screenshoot</i> data lokasi rumah sakit di Bali	36
Gambar 4.2	Peta Rute dari RSU Balimed ke RSUD Wangaya	39
Gambar 4.3	Graf Berarah Berbobot (RSU Balimed ke RSUD Wangaya).....	41
Gambar 4.4	Graf Hasil Rute Terpendek dari RSU Balimed ke RSUD Wangaya (Heuristik <i>Haversine</i>)	58
Gambar 4.5	Graf Hasil Rute Terpendek dari RSU Balimed ke RSUD Wangaya (Heuristik <i>Vincenty</i>)	62
Gambar 4.6	Peta Rute dari RSU Premagana ke RSD Mangusada	62
Gambar 4.7	Graf Berarah Berbobot (RSU Premagana ke RSD Mangusada)	65
Gambar 4.8	Graf Hasil Rute Terpendek dari RSU Premagana ke RSD Mangusada (<i>Haversine</i>).....	75
Gambar 4.9	Graf Hasil Rute Terpendek dari RSU Premagana ke RSD Mangusada (<i>Vincenty</i>)	79
Gambar 4.10	Peta Rute dari RS TK. II Udayana ke RSUD Bali Mandara	80
Gambar 4.11	Graf Berarah Berbobot (RS TK. II Udayana ke RSUD Bali Mandara).....	83
Gambar 4.12	Graf Hasil Rute Terpendek dari RS TK. II Udayana ke RSUD Bali Mandara (<i>Haversine</i>)	93
Gambar 4.13	Graf Hasil Rute Terpendek dari RS TK. II Udayana ke RSUD Bali Mandara (<i>Vincenty</i>)	97

DAFTAR LAMPIRAN

Lampiran 1. <i>Script Code</i>	107
Lampiran 2. Script Perhitungan Jarak Terpendek $g^*(n)$ ke T_3	111

ABSTRAK

Mahmuda. 2025. **Implementasi Algoritma A-Star dalam Menentukan Rute Terpendek Antar Rumah Sakit di Bali**. Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing : (1) Prof. Dr. H. Turmudi, M.Si., Ph.D. (2) Dr. Fachrur Rozi, M.Si.
Kata Kunci: Algoritma A-Star, Heuristik, *Haversine*, *Vincenty*, Rute Terpendek, graf berbobot, rumah sakit, Bali

Penentuan rute terpendek antar rumah sakit merupakan hal penting dalam mendukung pelayanan kesehatan, khususnya di wilayah Bali yang memiliki jaringan jalan yang kompleks. Penelitian ini bertujuan untuk mengimplementasikan algoritma A-Star dalam menentukan rute terpendek antar rumah sakit di Bali dengan memanfaatkan data koordinat geografis dan jarak aktual antar titik pada jaringan jalan. Data yang digunakan berupa koordinat *latitude* dan *longitude* rumah sakit serta titik persimpangan jalan yang diperoleh dari platform *Google Maps*. Algoritma A-Star dipilih karena kemampuannya dalam mengoptimalkan pencarian jalur dengan menggunakan fungsi heuristik yang efisien. Implementasi algoritma A-Star dilakukan dengan merepresentasikan jaringan jalan dalam bentuk graf berarah berbobot, di mana bobot pada setiap sisi graf menunjukkan jarak aktual antar simpul. Penelitian ini juga membandingkan dua metode heuristik, yaitu *Haversine* dan *Vincenty*, untuk mengukur estimasi jarak dalam proses pencarian rute. Hasil penelitian menunjukkan bahwa kedua metode heuristik menghasilkan rute dan jarak yang identik pada setiap pasangan rumah sakit yang diuji. Dengan demikian, algoritma A-Star berhasil diterapkan dalam penentuan rute terpendek, dan baik heuristik *Haversine* maupun *Vincenty* menunjukkan estimasi jarak yang setara dalam konteks graf dan data geografis yang digunakan.

ABSTRACT

Mahmuda. 2025. **Implementation of the A-Star Algorithm in Determining the Shortest Route Between Hospitals in Bali.** Undergraduate Thesis. Mathematics Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisors: (1) Prof. Dr. H. Turmudi, M.Si., Ph.D. (2) Dr. Fachrur Rozi, M.Si.

Kata Kunci: A-Star Algorithm, Heuristic, Haversine, Vincenty, Shortest Route, Weighted Graph, Hospitals, Bali

Determining the shortest route between hospitals is important in supporting health services, especially in the Bali region which has a complex road network. This study aims to implement the A-Star algorithm in determining the shortest route between hospitals in Bali by utilizing geographic coordinate data and actual distances between points on the road network. The data used are the latitude and longitude coordinates of hospitals and road intersection points obtained from the Google Maps platform. The A-Star algorithm was chosen because of its ability to optimize path searches using efficient heuristic functions. The implementation of the A-Star algorithm is carried out by representing the road network in the form of a weighted directed graph, where the weight on each side of the graph indicates the actual distance between nodes. This study also compares two heuristic methods, namely Haversine and Vincenty, to measure distance estimates in the route search process. The results show that both heuristic methods produce identical routes and distances for each pair of hospitals tested. Thus, the A-Star algorithm was successfully applied in determining the shortest route, and both the Haversine and Vincenty heuristics show equivalent distance estimates in the context of the graph and geographic data used.

مستخلص البحث

محمودة. ٢٠٢٥. تطبيق خوارزمية *A-Star* في تحديد المسارات أقصر مسافة بين المستشفيات في بالي. البحث العلمي. قسم الرياضيات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف (١) الأستاذ الدكتور ترمودي، الماجستير في العلوم؛ (٢) الدكتور فخر الرزي، الماجستير في العلوم.

الكلمات الأساسية: خوارزمية *A-Star*، *Heuristic*، *Haversine*، *Vincenty*، أقصر طريق،

الرسم البياني المرجح، المستشفى، بالي

ويعد تحديد أقصر طريق بين المستشفيات أمراً مهماً لدعم الخدمات الصحية، وخاصة في منطقة بالي التي لديها شبكة طرق معقدة. هدفت هذه الدراسة إلى تنفيذ خوارزمية *A-Star* في تحديد أقصر طريق بين المستشفيات في بالي من خلال الاستفادة من بيانات الإحداثيات الجغرافية والمسافات الفعلية بين النقاط على شبكة الطرق. البيانات المستخدمة هي إحداثيات خطوط العرض والطول لنقاط تقاطع المستشفى والطرق التي تم الحصول عليها من منصة *Google Maps*. تم اختيار خوارزمية *A-Star* بسبب قدرتها على تحسين البحث عن المسار باستخدام وظائف استدلالية فعالة. تم تنفيذ خوارزمية *A-Star* من خلال تمثيل شبكة الطرق في شكل رسم بياني موجه مرجح، حيث يشير الوزن على كل حافة من الرسم البياني إلى المسافة الفعلية بين العقد. وتقارن هذه الدراسة أيضاً بين طريقتين استدلاليتين، وهما *Haversine* و *Vincenty*، لقياس تقدير المسافة في عملية البحث عن المسار. وأظهرت النتائج أن كلا الطريقتين الاستدلالتين أنتجتا مسارات ومسافات متطابقة لكل زوج من المستشفيات التي تم اختبارها. وبالتالي، تم تطبيق خوارزمية *A-Star* بنجاح في تحديد أقصر طريق، وأظهرت كل من خوارزميات *Vincenty* و *Haversine* تقديرات مكافئة للمسافة في سياق الرسم البياني والبيانات الجغرافية المستخدمة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Algoritma *A-Star* pertama kali diperkenalkan oleh Peter Hart, Nils Nilsson, dan Bertram Raphael pada tahun 1968 (Marcelina & Yulianti, 2020). Algoritma *A-Star* bertujuan untuk menemukan rute terpendek, jarak tempuh paling singkat, serta waktu tempuh tercepat (Tryastie et al., 2024). Awalnya algoritma ini disebut sebagai Algoritma A, namun kemudian disempurnakan dengan penambahan heuristik sehingga dikenal sebagai *A-Star* (A^*). Algoritma *A-Star* merupakan metode pencarian rute yang optimal dan efisien, yang berarti algoritma ini tidak hanya dapat menemukan rute terbaik, tetapi juga dapat mencapai tujuan yang ditetapkan secara efisien. Kemampuan ini menjadikan algoritma *A-Star* sebagai pilihan yang tepat dalam menyelesaikan permasalahan pencarian rute optimal (Sriyanto et al., 2024).

Menurut Abdurahman & Maulana (2018), algoritma *A-Star* adalah salah satu teknik pencarian jalur yang paling efektif dan lengkap untuk menyelesaikan masalah pencarian atau penentuan rute terpendek. Algoritma *A-Star* yaitu salah satu teknik *best-first search* yang paling dikenal (Wati & Permatasari, 2019). Algoritma ini mengevaluasi simpul dengan menggabungkan dua komponen utama: $g(n)$, yang menunjukkan biaya rute dari simpul awal ke simpul tujuan, dan $h(n)$, yang menunjukkan estimasi jarak dari simpul awal ke simpul tujuan dan dihitung dengan fungsi *heuristic* (Nursaily & Agustian, 2023). Dalam proses pencarian, algoritma *A-Star* mengembangkan simpul dengan nilai gabungan biaya aktual dan estimasi terkecil untuk memastikan jalur optimal dari simpul awal menuju simpul tujuan dapat ditemukan secara efisien (Wiratama & Wibawa, 2023).

Beberapa algoritma umum yang sering digunakan dalam pencarian jalur adalah algoritma Bellman-Ford, algoritma Dijkstra, algoritma Distance Vector, algoritma Floyd-Warshall, algoritma Fulkerson, algoritma *Ant Colony*, dan algoritma *A-Star* (Dalem, 2018). Di antara algoritma tersebut, algoritma *A-Star* dipilih karena keunggulannya dalam menemukan rute secara efisien dan optimal. Algoritma ini memiliki waktu pencarian yang lebih cepat dibandingkan algoritma pencarian yang lain, dengan jumlah iterasi atau *loop* yang lebih sedikit, sehingga prosesnya lebih efisien. Selain itu, rute yang ditemukan oleh algoritma *A-Star* mungkin berbeda, tetapi memiliki biaya yang sama, menjadikannya sangat fleksibel dalam berbagai aplikasi (Tiwa, 2018). Dengan kelebihan-kelebihan ini, algoritma *A-Star* sangat efektif dalam aplikasi yang membutuhkan *respons* cepat, seperti penentuan rute ambulans dalam kondisi darurat, di mana jarak tempuh yang paling singkat menjadi prioritas utama untuk mempercepat penanganan.

Berbagai penelitian telah membuktikan efektivitas algoritma *A-Star* dalam menyelesaikan masalah pencarian rute. Penelitian yang dilakukan oleh Arsyad et al., (2019) menunjukkan bahwa penggunaan algoritma *A-Star* sangat efektif dalam mencari rute terpendek menuju Puskesmas rawat inap di Banyumas. Hasil pengujian menunjukkan bahwa algoritma *A-Star* berhasil menemukan rute terpendek dengan tingkat keberhasilan yang tinggi, serta lebih cepat dibandingkan dengan algoritma Dijkstra. Selain itu, penelitian yang dilakukan oleh Ardiansyah et al., (2022) juga mengungkapkan bahwa algoritma *A-Star* sangat efektif dalam menentukan rute terpendek menuju rumah sakit rujukan COVID-19 di Kota Makassar. Penelitian tersebut juga mengungkapkan bahwa penerapan algoritma *A-Star* dapat menghitung jarak dan waktu tempuh dengan mempertimbangkan faktor-

faktor seperti koordinat geografis dan kecepatan rata-rata kendaraan, menghasilkan jalur tercepat yang dapat membantu masyarakat dalam situasi darurat. Hasil pengujian aplikasi menunjukkan tingkat keberhasilan sebesar 79,2%, yang mencerminkan kepuasan pengguna terhadap aplikasi ini dalam menentukan rute terpendek menuju rumah sakit COVID-19. Penelitian-penelitian tersebut menunjukkan bahwa algoritma *A-Star* dapat diimplementasikan dengan baik dan memberikan hasil yang efisien dalam berbagai aplikasi pencarian rute.

Penentuan rute terpendek memiliki dampak signifikan dalam berbagai aplikasi praktis, khususnya di sektor transportasi medis. Di Bali, yang merupakan destinasi wisata internasional dengan jumlah rumah sakit yang tersebar di berbagai wilayah, efisiensi transportasi sangat penting, terutama dalam kondisi darurat medis. Dalam situasi di mana pasien membutuhkan perawatan lanjutan di rumah sakit lain, proses pemindahan pasien harus dilakukan secepat mungkin. Oleh karena itu, diperlukan sistem yang mampu menentukan rute terpendek agar ambulans dapat menempuh jalur dengan jarak paling singkat. Dengan menerapkan algoritma *A-Star*, proses pencarian rute antar rumah sakit dapat dioptimalkan, sehingga ambulans dapat sampai ke tujuan dengan efisien tanpa memperhitungkan waktu tempuh yang mungkin dipengaruhi oleh kondisi lalu lintas. Implementasi sistem ini tidak hanya membantu menghemat jarak perjalanan, tetapi juga berperan penting dalam meningkatkan respons layanan kesehatan dan keselamatan pasien, terutama dalam situasi darurat.

Menjaga dan melindungi nyawa merupakan prinsip utama dalam Islam, sebagaimana yang terkandung dalam *maqashid syariah*, yaitu perlindungan terhadap jiwa (*hifz al-nafs*) (Harahap, 2021). Prinsip ini menekankan pentingnya

setiap usaha yang dilakukan untuk melindungi nyawa manusia dari segala bentuk bahaya atau ancaman. Hal ini ditegaskan dalam firman Allah SWT dalam QS. Al-Isra' [17]: 33:

وَلَا تَقْتُلُوا النَّفْسَ الَّتِي حَرَّمَ اللَّهُ إِلَّا بِالْحَقِّ ۚ وَمَنْ قُتِلَ مَظْلُومًا فَقَدْ جَعَلْنَا لَوْلِيهِ سُلْطَانًا فَلَا يُسْرِفُ فِي الْقَتْلِ ۚ إِنَّهُ كَانَ مَنْصُورًا

(Kementerian Agama, 2022)

Artinya: "*Dan janganlah kamu membunuh jiwa yang diharamkan Allah melainkan dengan haq. Dan barang siapa dibunuh secara zalim, maka sesungguhnya Kami telah memberi kekuasaan kepada walinya, tetapi janganlah keluarganya melampaui batas dalam membunuh. Sesungguhnya ia adalah orang yang dimenangkan.*" (QS. Al-Isra': 33).

Ayat ini mengandung makna mendalam bahwa Islam sangat menekankan pentingnya melindungi nyawa manusia. Allah secara tegas melarang pembunuhan terhadap jiwa manusia tanpa alasan yang dibenarkan oleh syariat. Perlindungan ini mencakup tidak hanya keselamatan diri sendiri, tetapi juga usaha menjaga keselamatan orang lain. Upaya menjaga dan menyelamatkan nyawa merupakan wujud nyata dari penerapan *maqashid syariah*, khususnya perlindungan terhadap jiwa (*hifz al-nafs*). Segala bentuk usaha yang bertujuan untuk mencegah bahaya atau ancaman terhadap keselamatan manusia merupakan bagian dari nilai-nilai luhur yang diajarkan dalam Islam. Salah satu wujud implementasi nilai tersebut adalah dengan memanfaatkan teknologi modern untuk meningkatkan efisiensi dan kecepatan pelayanan kesehatan. Sebagai contoh, penerapan algoritma A-*Star* dalam menentukan rute terpendek antar rumah sakit dapat membantu meminimalkan jarak tempuh dalam kondisi darurat, sehingga nyawa dapat terselamatkan dengan lebih cepat dan efektif.

Penggunaan ilmu pengetahuan dan teknologi untuk meminimalkan jarak tempuh dalam situasi darurat mencerminkan upaya nyata untuk melindungi nyawa

manusia. Dengan menerapkan algoritma *A-Star* dalam menentukan rute terpendek antar rumah sakit, layanan kesehatan dapat merespons lebih cepat dan memberikan manfaat nyata bagi masyarakat luas. Penelitian ini sejalan dengan prinsip Islam yang mendorong optimalisasi segala bentuk ikhtiar untuk menjaga keselamatan jiwa sebagai prioritas utama.

Dengan demikian, penelitian ini bertujuan untuk mengimplementasikan algoritma *A-Star* dalam menentukan rute terpendek antar rumah sakit di Bali. Diharapkan penerapan algoritma ini dapat mempercepat penanganan pasien darurat, meningkatkan efisiensi, dan memperluas jangkauan rute antar rumah sakit. Penelitian ini juga mencerminkan upaya untuk merealisasikan nilai-nilai Islam dalam menjaga nyawa umat manusia.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah yang akan dikaji pada penelitian ini adalah sebagai berikut:

1. Bagaimana perbandingan keakuratan heuristik *Haversine* dan *Vincenty* dalam algoritma *A-Star* untuk menentukan rute terpendek antar rumah sakit di Bali?
2. Bagaimana implementasi algoritma *A-Star* dalam menentukan rute terpendek antar rumah sakit di Bali?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah di atas, tujuan dari penelitian ini adalah:

1. Membandingkan dan menganalisis keakuratan serta efektivitas heuristik *Haversine* dan *Vincenty* dalam algoritma *A-Star* untuk menentukan rute terpendek antar rumah sakit di Bali.

2. Mengimplementasi algoritma *A-Star* dalam menentukan rute terpendek antar rumah sakit di Bali.

1.4 Manfaat Penelitian

Manfaat dari penelitian ini dibedakan ke dalam tiga aspek, yaitu aspek pelayanan kesehatan, masyarakat umum, dan ilmiah. Penjelasan sebagai berikut:

1. Pelayanan Kesehatan

Penelitian ini dapat memberikan kontribusi dalam meningkatkan efisiensi sistem rujukan antar rumah sakit di Bali. Dengan informasi rute terpendek berdasarkan jarak (kilometer) yang diperoleh melalui algoritma *A-Star*, rumah sakit dapat merencanakan jalur transportasi antar fasilitas kesehatan secara lebih optimal. Meskipun waktu tempuh bisa dipengaruhi faktor eksternal, informasi mengenai jarak terpendek tetap menjadi dasar penting dalam proses pengambilan keputusan logistik medis.

2. Masyarakat Umum

Masyarakat, termasuk pasien atau keluarga pasien, dapat menggunakan hasil penelitian ini sebagai panduan dalam memilih rute rumah sakit berdasarkan jarak terpendek. Hal ini sangat berguna untuk perencanaan perjalanan yang efisien, terutama di wilayah yang belum *familiar* atau dalam kondisi darurat. Walaupun waktu tempuh dapat bervariasi, rute terpendek tetap memberikan orientasi yang bermanfaat dalam pengambilan keputusan terkait jalur perjalanan.

3. Ilmiah

Penelitian ini memberikan kontribusi pada pengembangan bidang ilmu komputer dan matematika terapan, khususnya dalam pemanfaatan algoritma *A-Star* dan evaluasi fungsi heuristik. Dengan membandingkan keakuratan fungsi heuristik *Haversine* dan *Vincenty* dalam konteks jarak geografis, penelitian ini memperkaya literatur terkait optimasi rute berbasis jarak. Hasil penelitian juga dapat menjadi referensi untuk pengembangan sistem pendukung keputusan di bidang transportasi medis, sistem informasi geografis (GIS), serta bidang lain yang membutuhkan pencarian jalur optimal.

1.5 Batasan Masalah

Agar penyelesaian masalah dalam penelitian ini dapat dilakukan secara terarah dan optimal, maka ditetapkan batasan-batasan sebagai berikut:

1. Penentuan rute terpendek dilakukan antar rumah sakit yang telah ditentukan sebagai simpul awal dan tujuan, dan tidak mencakup seluruh kombinasi antar rumah sakit di Bali.
2. Penelitian ini hanya mempertimbangkan jarak (dalam satuan kilometer) dalam proses pencarian rute terpendek menggunakan algoritma *A-Star*, tanpa mempertimbangkan waktu tempuh, kemacetan, atau lalu lintas lainnya.
3. Fungsi heuristik yang digunakan dalam algoritma *A-Star* terbatas pada fungsi heuristik *Haversine* dan *Vincenty*.
4. Data koordinat dan jaringan jalan antar rumah sakit diperoleh melalui *Google Maps*.
5. Penelitian ini merepresentasikan jaringan jalan dalam bentuk graf berarah berbobot (*weighted directed graph*), di mana setiap sisi menunjukkan arah dan

jarak antar simpul yang merepresentasikan lokasi rumah sakit dan titik persimpangan jalan.

6. Jenis permasalahan rute terpendek yang digunakan adalah *single-pair shortest path*, yaitu pencarian rute terpendek dari satu simpul awal ke satu simpul tujuan tertentu.

BAB II

KAJIAN TEORI

2.1 Teori Pendukung

2.1.1 Teori Graf

Teori graf pertama kali diperkenalkan oleh Leonhard Euler pada tahun 1736 dalam tulisannya yang membahas upaya untuk menyelesaikan masalah jembatan Königsberg yang terkenal di Eropa. Masalah yang dimaksud adalah menunjukkan kemungkinan melintasi empat wilayah yang terhubung dengan tujuh jembatan di atas sungai Pregel di Königsberg, Rusia, dengan melewati setiap jembatan hanya sekali dan kembali ke simpul awal. Untuk memecahkan masalah tersebut, Euler memodelkan wilayah daratan sebagai simpul dan jembatan sebagai garis atau sisi yang menghubungkan simpul-simpul tersebut. Dari hasil pembuktiannya, Euler menyimpulkan bahwa tidak ada cara untuk melintasi semua jembatan tersebut dalam satu kali tanpa mengulang dan kembali ke posisi awal. Pemecahan masalah Königsberg ini menjadi awal lahirnya teori graf dan menjadi salah satu fondasi dasar dalam pengembangan matematika diskrit (Slamin, 2019).

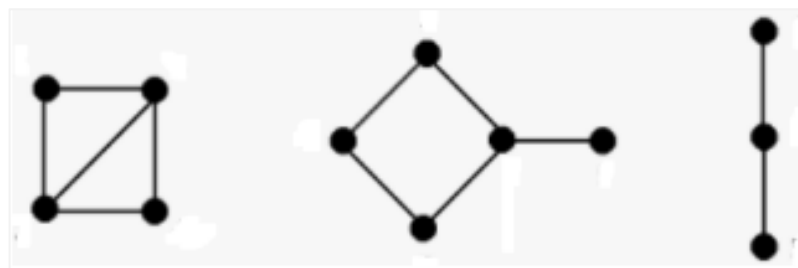
Graf didefinisikan sebagai pasangan himpunan (V, E) , yang ditulis dalam notasi $G = (V, E)$ (Munir, 2023). Graf G terdiri dari dua himpunan: himpunan berhingga dan tak kosong. Himpunan tak kosong $V(G)$ mencakup objek-objek yang disebut simpul, dan himpunan berhingga $E(G)$, yang terdiri dari elemen-elemen yang disebut sisi. Setiap elemen e dalam $E(G)$ menghubungkan pasangan tak berurutan dari simpul-simpul di $V(G)$. Dengan demikian $V(G)$ disebut sebagai himpunan simpul G , dan $E(G)$ disebut sebagai himpunan sisi G (Diana et al., 2016).

Menurut Marsudi (2016), graf G didefinisikan sebagai pasangan himpunan (V, E) , dimana V merupakan himpunan tak kosong berhingga yang terdiri dari simpul-simpul, dan E adalah himpunan berhingga yang mungkin kosong, terdiri dari pasangan-pasangan simpul dalam $V \times V$ yang disebut sisi (*edges*). Jika $V = V(G)$ dan $E = E(G)$, maka graf dapat dinotasikan sebagai $G = (V, E)$ atau $G = (V(G), E(G))$. Dengan demikian, V merupakan himpunan simpul yang tidak kosong, sedangkan E boleh kosong, yang berarti sebuah graf mungkin tidak memiliki jalur tetapi harus memiliki simpul. Graf yang hanya memiliki satu simpul tanpa jalur disebut graf trivial, yang merupakan bentuk graf paling sederhana dan menjadi dasar bagi konsep graf yang lebih kompleks (Tiwa, 2018).

Berdasarkan pada pengelompokannya, graf dapat diklasifikasikan ke dalam berbagai kategori. Keberadaan atau ketiadaan gelang (*loop*) maupun sisi ganda dalam graf menjadi faktor utama dalam menentukan kategori (jenis) graf. Menurut Cahyono (2022), terdapat berbagai jenis graf, di antaranya:

1. Graf Sederhana (*simple graph*)

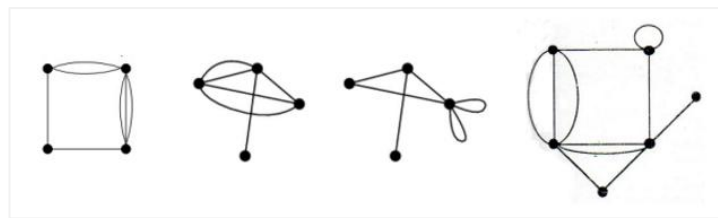
Menurut (Munir, 2023), graf sederhana adalah graf yang tidak memiliki *loop* dan sisi ganda. Sisi ganda terjadi ketika dua simpul dihubungkan oleh lebih dari satu sisi, sedangkan *loop* adalah sisi yang dimulai dan diakhiri pada simpul yang sama.



Gambar 2.1 Graf Sederhana
(Sumber: Munir, 2023)

2. Graf Tak-Sederhana (*unsimple graph*)

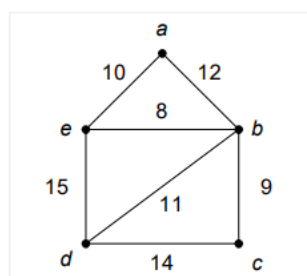
Graf tidak sederhana adalah graf yang mempunyai *loop* atau banyak sisi (Munir, 2023). Graf ganda dan graf semu adalah dua kategori yang termasuk dalam graf ini. Graf yang dua atau lebih sisinya menghubungkan pasangan simpul yang sama disebut graf ganda (*multi-graph*). Sebaliknya, graf dengan *loop* yakni sisi-sisi yang menghubungkan suatu simpul dengan dirinya sendiri disebut graf semu.



Gambar 2.2 Graf Tak Sederhana
(Sumber: Munir, 2023)

3. Graf Berbobot

Menurut (Munir, 2023), dalam graf berbobot, setiap sisi memiliki nilai atau bobot tertentu. Nilai pada tiap sisi bisa berbeda-beda tergantung pada permasalahan direpresentasikan oleh graf. Graf berbobot dapat digunakan untuk memodelkan berbagai hal, seperti jarak antara dua simpul dalam jaringan jalan, rute pengiriman barang dengan jarak tempuh tertentu, jarak antar rumah sakit dalam sistem rujukan pasien, dan sebagainya.

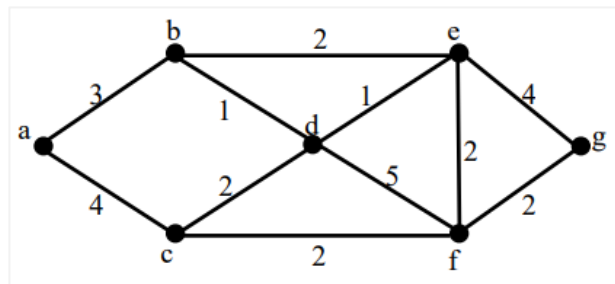


Gambar 2.3 Contoh Graf Berbobot
(Sumber: Munir, 2023)

Graf berbobot dapat dikelompokkan ke dalam dua kategori, yaitu:

a. Graf Tidak Berarah Berbobot

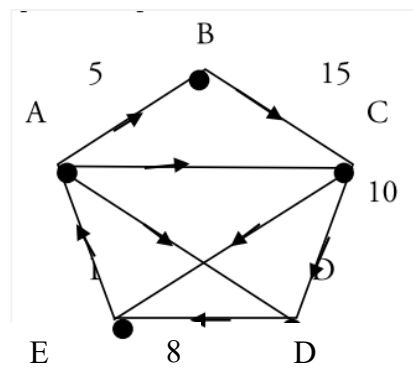
Graf tanpa arah dengan bobot pada tiap sisinya disebut graf tak berarah berbobot. Bobot ini dapat merepresentasikan berbagai hal, seperti biaya, jarak, kapasitas, atau aspek lain dengan makna serupa (Panjaitan dan Aprilia, 2022).



Gambar 2.4 Contoh Graf Tak-Berarah Berbobot
(Sumber: Panjaitan dan Aprilia, 2022)

b. Graf Berarah Berbobot

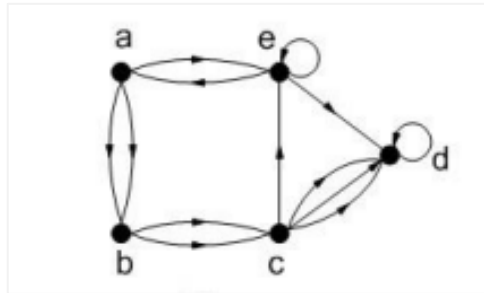
Suatu graf berarah yang tidak mempunyai jalur ganda disebut graf berarah berbobot jika setiap jalurnya dipetakan dengan bilangan yang disebut dengan bobot (Panjaitan dan Aprilia, 2022).



Gambar 2.5 Contoh Graf Berarah Berbobot
(Sumber: Panjaitan dan Aprilia, 2022)

4. Graf Berarah

Graf berarah adalah graf yang setiap sisinya mempunyai arah tertentu (Munir, 2023).



Gambar 2.6 Graf Berarah
(Sumber: Panjaitan dan Aprilia, 2022)

2.1.2 Algoritma A-*Star*

Dalam buku yang berjudul *A-Star Partfinding for Beginner*, Patrick Lester menceritakan sejarah dan penjelasan mengenai algoritma A-*Star*. Peter Hart, Nils Nilsson, dan Betram Raphael pertama kali merepresentasikan algoritma ini pada tahun 1968 (Hermawan & Setiyani, 2019). Dalam ilmu komputer, A-*Star* dikenal sebagai algoritma pencarian dalam graf yang sangat efisien dalam menemukan rute dengan biaya paling rendah dari suatu simpul awal menuju simpul tertentu, atau hingga mencapai satu atau lebih kemungkinan tujuan (Hart et al., 1968).

Algoritma A-*Star* adalah salah satu metode yang banyak diterapkan dalam penelusuran graf (*graph traversal*) dan pencarian jalur (*part finding*), yaitu proses menentukan rute paling optimal antara dua simpul (Prasetyo et al., 2019). Algoritma ini dikembangkan dari pendekatan BFS (*Best First Search*) dengan melakukan modifikasi pada aspek heuristiknya. A-*Star* berfungsi untuk mengurangi total biaya lintasan yang terdapat dalam pendekatan BFS (Yamin & Talai, 2015). Dalam jangka waktu yang ideal, A-*Star* akan menawarkan solusi terbaik. A-*Star* bekerja sama cepat dan efektifnya dengan BFS dalam pencarian rute dasar tanpa hambatan di peta. Saat dihadapkan pada penghalang di peta, A-*Star* dapat menemukan jalan keluarnya tanpa terjebak.

Menurut Russel & Norvig (2003), dikutip dalam Mauliddiyah (2021), algoritma *A-Star* terdiri dari lima bagian utama, yaitu: simpul awal, simpul tujuan, *open list*, *closed list*, dan *cost*. Simpul awal adalah posisi permulaan dari lokasi saat ini, sedangkan simpul tujuan adalah lokasi akhir yang ingin dicapai. *Cost* merupakan nilai jarak yang ditempuh untuk mencapai tujuan. *Open list* berfungsi sebagai *priority queue*, di mana setiap simpul yang masuk pertama akan keluar lebih dahulu berdasarkan kriteria tertentu. *Closed list* berfungsi sebagai *stack* (tumpukan), di mana simpul yang terakhir masuk akan dikeluarkan pertama kali. Selain menyimpan simpul yang sudah dilewati, *closed list* juga membantu menemukan jarak terdekat ketika simpul tujuan telah tercapai.

Metode *A-Star* merupakan teknik pencarian jalur yang sering dimanfaatkan untuk menentukan rute tercepat dari simpul awal hingga simpul tujuan. Proses pencarian jalur pada graf banyak diaplikasikan dalam permainan berbasis simulasi (Mauliddiyah, 2021). Dari sisi matematis, evaluasi serta estimasi dalam metode *A-Star* dapat dirumuskan sebagai berikut (Putra et al., 2020):

$$f(n) = g(n) + h(n) \quad (2.1)$$

Keterangan:

$h(n)$: Nilai heuristik antara koordinat

$g(n)$: Jarak sebenarnya dari simpul asal ke simpul n

$f(n)$: Perkiraan solusi dengan jarak terpendek melalui n

Dalam penerapannya, Algoritma *A-Star* memiliki beberapa terminologi mendasar, seperti simpul awal, A, simpul, daftar terbuka, daftar tertutup, harga (F), dan simpul tujuan (Syihabuddin et al., 2022).

1. Simpul awal mengacu pada lokasi awal suatu titik.

2. A adalah titik yang sedang diproses dalam algoritma pencarian rute terpendek.
3. Simpul adalah representasi kecil dari area *pathfinding*, yang bisa berbentuk persegi, lingkaran, atau segitiga.
4. *Open list* adalah tempat untuk menyimpan data simpul yang dapat diakses dari simpul awal atau simpul yang sedang diproses.
5. *Closed list* adalah tempat menyimpan data simpul yang telah diproses sebelum A dan bagian dari rute terpendek.
6. Harga (F) adalah nilai yang diperoleh dari proses penjumlahan, mencakup nilai setiap simpul dalam rute terpendek dari *starting point* ke A , serta jumlah nilai perkiraan dari sebuah simpul ke simpul tujuan.
7. Simpul tujuan adalah simpul yang menjadi target pencarian.

Berikut adalah langkah-langkah algoritma A-Star:

1. Mulai (*Start*)

Algoritma dimulai dengan mempersiapkan semua variabel dan struktur yang diperlukan untuk proses pencarian rute terpendek.

2. Input graf

Masukkan representasi graf yang terdiri dari simpul dan sisi (*edge*), di mana setiap simpul mewakili lokasi, dan sisi mewakili jalan atau jalur penghubung antar simpul dengan bobot yang mencerminkan jarak tempuh.

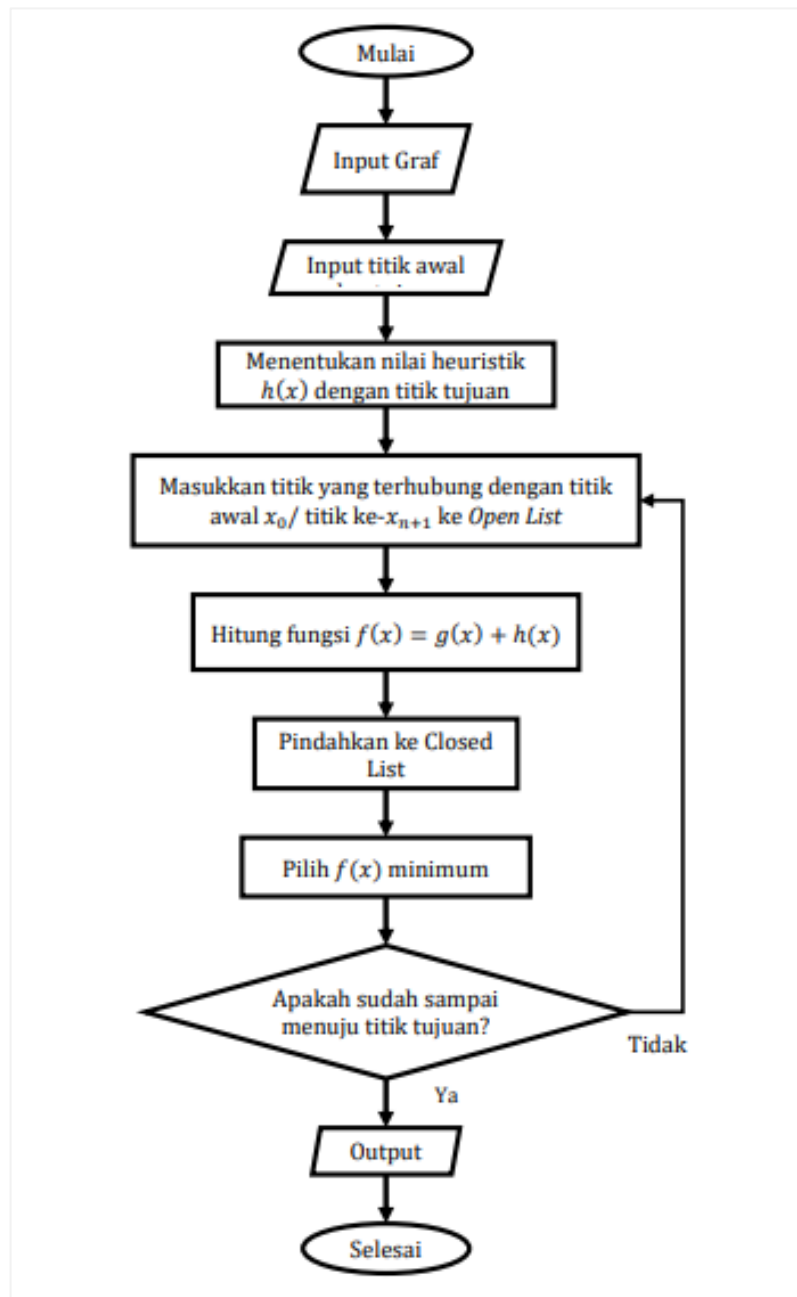
3. Input simpul awal

Tentukan simpul awal, yaitu lokasi dari mana pencarian rute akan dimulai.

4. Menentukan nilai heuristik $h(x)$

Untuk setiap simpul, hitung nilai heuristik $h(x)$, yang merupakan perkiraan

Berikut merupakan *Flowchart* Algoritma A-Star:



Gambar 2.7 *Flowchart* Algoritma A-Star
(Sumber: Syihabuddin et al., 2022)

jarak dari simpul tersebut menuju tujuan. Nilai heuristik digunakan sebagai panduan untuk memilih jalur yang paling efisien menuju tujuan.

5. Masukkan simpul yang terhubung dengan simpul awal ke *open list*

Semua simpul yang langsung terhubung dengan simpul awal dimasukkan ke dalam *open list*, yang merupakan daftar simpul yang akan dievaluasi lebih lanjut.

6. Hitung fungsi $f(x) = g(x) + h(x)$

Untuk setiap simpul dalam *open list*, hitung nilai $f(x)$, di mana:

- a. $g(x)$ adalah jarak sebenarnya dari simpul awal ke simpul x .
- b. $h(x)$ adalah nilai heuristik antara koordinat.

Kombinasi $g(x)$ dan $h(x)$ digunakan untuk memandu pencarian ke arah yang paling optimal.

7. Pindahkan simpul ke *closed list*

Simpul dengan nilai $f(x)$ terendah dipindahkan dari *open list*. *Closed list* berisi simpul-simpul yang sudah dievaluasi dan tidak perlu dievaluasi lagi.

8. Pilih simpul dengan $f(x)$ minimum

Dari *open list*, pilih simpul yang memiliki nilai $f(x)$ terkecil untuk dievaluasi lebih lanjut, karena simpul ini dianggap sebagai jalur terbaik untuk dilanjutkan.

9. Apakah sudah mencapai tujuan?

- a. Jika Ya, pencarian berhenti, dan rute terpendek yang ditemukan ditampilkan sebagai *output*. Algoritma selesai.
- b. Jika Tidak, kembali ke langkah ke 5 untuk mengevaluasi simpul berikutnya yang terhubung dengan simpul saat ini.

2.1.3 Fungsi Heuristik

Kata *Heuristic* berasal dari bahasa Yunani yaitu *heuriskein*, yang berarti “mencari” atau “menemukan”. Dalam dunia pemrograman, istilah “heuristik”

sering digunakan sebagai lawan dari istilah algoritmik. Menurut Ratnawati et al., (2012), heuristik merupakan suatu metode yang dapat memecahkan permasalahan, tetapi solusi yang diperoleh belum tentu selalu dapat ditemukan dengan pasti.

Dalam studi metode pencarian, heuristik didefinisikan sebagai sebuah fungsi yang menghasilkan estimasi atau perkiraan biaya terhadap suatu penyelesaian. *Heuristic* berperan dalam meningkatkan efisiensi proses pencarian (Yuliana et al., 2012). Teknik pencarian heuristik (*heuristic searching*) adalah metode yang mengarahkan proses pencarian secara selektif ke jalur-jalur yang memiliki peluang tertinggi untuk mencapai tujuan, meskipun terkadang mengorbankan kelengkapan (*completeness*).

Salah satu algoritma pencarian populer yang menggunakan fungsi heuristik adalah algoritma *A-Star*. Dalam algoritma ini, fungsi heuristik digunakan untuk memperkirakan seberapa dekat suatu simpul dengan tujuan akhir (*goal*), tanpa harus mengetahui secara pasti keseluruhan jalur. Fungsi ini sangat penting agar algoritma dapat mengambil keputusan yang efisien dalam memilih jalur terbaik.

Agar fungsi heuristik bekerja secara optimal dalam algoritma *A-Star*, fungsi ini harus memenuhi dua sifat penting, yaitu *admissible* dan *consistent*. Kedua sifat ini sangat krusial untuk menjamin bahwa pencarian jalur yang dilakukan oleh *A-Star* akan menemukan solusi yang benar dan efisien. Berikut penjelasan dari masing-masing sifat tersebut:

1. *Admissible*

Fungsi heuristik dikatakan *admissible* jika nilai heuristik tidak pernah melebihi-lebihkan biaya sebenarnya untuk mencapai tujuan dari simpul

mana pun dalam graf. Dengan kata lain, estimasi biaya yang diberikan oleh fungsi heuristik $h(n)$ selalu kurang dari atau sama dengan biaya aktual terendah untuk mencapai tujuan (Studi & Informatika, 2024).

2. *Consistent*

Fungsi heuristik disebut *consistent* apabila selain menjamin bahwa perkiraan biaya tidak melebihi biaya sebenarnya, juga memastikan bahwa selisih antara perkiraan biaya dari satu simpul ke simpul lain tidak lebih besar dari biaya sebenarnya untuk berpindah di antara kedua simpul tersebut. Dengan sifat ini, algoritma *A-Star* memiliki kemampuan yang lebih besar untuk menjelajahi ruang pencarian dengan lebih efisien.

Dalam penelitian ini, fungsi heuristik digunakan untuk memperkirakan jarak dari rumah sakit awal menuju rumah sakit tujuan. Estimasi ini penting untuk membantu algoritma dalam memilih rute terpendek dengan efisien. Dua jenis fungsi heuristik yang umum digunakan dalam perhitungan jarak geografis adalah *Heuristik* dan *Vincenty*. Berikut adalah penjelasan kedua fungsi tersebut:

1. Fungsi Heuristik *Haversine*

Fungsi heuristik yang digunakan dalam penelitian ini adalah *Haversine distance*. Fungsi ini memperhitungkan bentuk bumi yang melengkung. Oleh karena itu, fungsi ini lebih akurat untuk menghitung jarak antara dua lokasi geografis berdasarkan koordinat lintang dan bujur, seperti pada kasus penentuan rute antar rumah sakit di seluruh wilayah Bali.

Secara matematis, rumus *Haversine distance* (Arsyad et al., 2019) dinyatakan sebagai berikut:

$$h(n) = 2r \times \arcsin \left(\sqrt{\sin^2 \left(\frac{\Delta \varphi}{2} \right) + \cos(\varphi_1) \times \cos(\varphi_2) \times \sin^2 \left(\frac{\Delta \lambda}{2} \right)} \right), \quad (2.2)$$

dengan keterangan:

$h(n)$: Nilai heuristik untuk simpul n

r : Jari-jari bumi (rata-rata 6371 km)

φ_1, φ_2 : Lintang dari simpul awal dan simpul tujuan (dalam radian)

$\Delta \varphi$: Selisih lintang antara dua simpul

$\Delta \lambda$: Selisih bujur antara dua simpul

2. *Vincenty*

Menurut Windarni & Setiawan (2022) *Vincenty* digunakan dalam *geodesi* untuk menghitung jarak antara dua simpul di permukaan bumi, yang diasumsikan sebagai *ellipsoid*. Misalnya, ada dua simpul (A dan B), $A(\varphi_1, L_1)$ dan $B(\varphi_2, L_2)$. Diberikan koordinat dua simpul (φ_1, L_1) dan (φ_2, L_2) , masalah invers mencari *azimuth* α_1, α_2 dan jarak *elipsoid* s . Kemudian, U_1, U_2 , dan L dihitung, dan nilai awal $\lambda = L$ ditetapkan. Kemudian, evaluasi persamaan berikut secara iteratif hingga λ mendekati nol,

$$\sin \sigma = \sqrt{\frac{(\cos(U_2) \sin(\lambda))^2 + (\cos(U_1) \sin(U_2) - \sin(U_1) \cos(U_2) \cos(\lambda))^2}{\sin^2(U_1) \cos^2(U_2) \cos^2(\lambda)}}} \quad (2.3)$$

$$\cos \sigma = \sin(U_1) \sin(U_2) + \cos(U_1) \cos(U_2) \cos(\lambda) \quad (2.4)$$

$$\sigma = \arctan \left(\frac{\sin \sigma}{\cos \sigma} \right) \quad (2.5)$$

$$\sin \alpha = \frac{\cos(U_1) \cos(U_2) \sin(\lambda)}{\sin \sigma} \quad (2.6)$$

$$\cos(2\sigma_m) = \cos \sigma - \frac{2 \sin(U_1) \sin(U_2)}{\cos^2 \alpha} \quad (2.7)$$

$$C = \frac{f}{16} \cos^2 \alpha [4 + f(4 - 3 \cos^2 \alpha)] \quad (2.8)$$

$$\lambda = L(1 - C)f \sin \alpha \{ \sigma + C \sin \sigma [\cos(2\sigma_m) + C \cos \sigma (-1 + 2\cos^2(2\sigma_m))] \} \quad (2.9)$$

Ketika λ telah mencapai tingkat akurasi yang diinginkan (10^{-12} setara dengan sekitar 0,06 mm), evaluasi hal berikut:

$$u^2 = \cos^2 \alpha \left(\frac{a^2 - b^2}{b^2} \right) \quad (2.10)$$

$$A = 1 + \left(\frac{u^2}{16384} \right) \{ 4096 + u^2 [-768 + u^2 (320 - 175 u^2)] \} \quad (2.11)$$

$$B = \left(\frac{u^2}{1024} \right) \{ 256 + u^2 [-128 + u^2 (74 - 47 u^2)] \} \quad (2.12)$$

$$\Delta \sigma = B \sin \sigma \{ \cos(2\sigma_m) + \frac{1}{4} B [\cos \sigma (-1 + 2\cos^2(2\sigma_m)) - \frac{B}{6} \cos(2\sigma_m) (-3 + 4 \sin^2 \sigma) (-3 + 4 \cos^2(2\sigma_m))] \} \quad (2.13)$$

$$s = b A (\sigma - \Delta \sigma) \quad (2.14)$$

$$\alpha_1 = \arctan \left(\frac{\cos(U_2) \sin(\lambda)}{\cos(U_1) \sin(U_2) - \sin(U_1) \cos(U_2) \cos(\lambda)} \right) \quad (2.15)$$

$$\alpha_2 = \arctan \left(\frac{\cos(U_1) \sin(\lambda)}{-\sin(U_1) \cos(U_2) + \cos(U_1) \sin(U_2) \cos(\lambda)} \right) \quad (2.16)$$

dengan,

a : Panjang sumbu semi-mayor *elipsoid* (radius di ekuator)
(6378137,0 meter),

f : *Flattening* (perataan) *elipsoid*, yaitu $\left(\frac{1}{298,25722356} \right)$,

b : $(1 - f) a$, yaitu panjang sumbu semi-minor *elipsoid* (radius di kutub) (6356752,314245 meter),

φ_1, φ_2 : *Latitude* 1 dan *latitude* 2,

U_1 : $\arctan((1 - f) \tan \Phi_1)$,

U_2 : $\arctan((1 - f) \tan \Phi_2)$,

- L : $L_2 - L_1$, yaitu selisi bujur antara dua titik,
 λ_1, λ_2 : Bujur titik pada bola bantu (*additional sphere*),
 α_1, α_2 : *Azimut* maju pada titik-titik,
 s : Jarak *elipsoidal* antara dua titik,
 σ : Panjang busur antara dua titik pada bola bantu (*additional sphere*).

Kedua fungsi heuristik tersebut dibandingkan dalam penelitian ini untuk mengetahui pengaruhnya terhadap hasil dan kinerja algoritma A-*Star*. Dengan membandingkan hasil yang diperoleh dari fungsi *Haversine* dan *Vincenty*, dapat dianalisis sejauh mana perbedaan estimasi jarak mempengaruhi efektivitas pencarian rute terpendek antar rumah sakit. Pemilihan fungsi heuristik yang tepat diharapkan mampu mendukung pengambilan keputusan yang lebih efisien dan akurat dalam situasi darurat, dengan fokus pada penentuan rute berdasarkan jarak antar lokasi, bukan waktu tempuh.

2.1.4 Rute Terpendek

Rute terpendek merupakan jalur paling minimum yang harus ditempuh untuk mencapai suatu simpul tujuan dari simpul awal. Jalur ini dapat ditentukan dengan memanfaatkan struktur graf, yaitu himpunan titik-titik pada bidang dua dimensi yang saling terhubung melalui sisi (*edge*). Dalam sebuah graf, terdapat sejumlah simpul yang dihubungkan oleh sisi-sisi penghubung.

Menemukan jalur terpendek adalah salah satu masalah yang dapat diatasi dengan graf. Jalur terpendek adalah rute terkecil yang diperlukan untuk berpindah dari satu simpul awal ke simpul lainnya. Dalam proses pencarian rute terpendek, tantangan yang harus dilalui yaitu menentukan jalur manakah yang harus dilalui

untuk memperoleh rute terpendek dari satu simpul ke simpul lainnya (Yamin & Talai, 2015).

Terdapat beberapa jenis permasalahan rute terpendek, diantaranya: (Yamin & Talai, 2015).

1. Rute terpendek yang menghubungkan dua simpul tertentu (*a pair shortest path*).
2. Rute terpendek yang menghubungkan semua pasangan simpul (*all pairs shortest path*).
3. Rute terpendek dari satu simpul tertentu ke semua simpul lainnya (*single source shortest path*).
4. Rute terpendek antara dua simpul yang harus melalui beberapa simpul tertentu (*intermediate shortest path*).

2.1.5 Rumah Sakit di Bali

Dalam rangka memberikan pelayanan kesehatan baik dalam keadaan darurat maupun non darurat, rumah sakit merupakan fasilitas pelayanan kesehatan yang sangat vital. Di Bali, terdapat berbagai rumah sakit yang tersebar di berbagai kabupaten dan kota, mulai dari rumah sakit pemerintah hingga swasta, dengan layanan yang mencakup perawatan umum, spesialis, dan gawat darurat. Keberadaan rumah sakit sangat diperlukan, terutama terkait dengan tantangan dalam penanganan pasien, baik dari penduduk lokal maupun wisatawan.

Bali merupakan salah satu tujuan wisata populer di Indonesia juga memiliki tantangan tersendiri dalam hal penanganan pasien, baik penduduk lokal maupun wisatawan. Salah satu tantangan yang dihadapi adalah memastikan transportasi antar rumah sakit dapat berjalan dengan cepat dan efisien, terutama dalam kondisi

darurat di mana waktu sangat berharga. Mengingat kondisi geografis Bali yang terdiri dari wilayah perkotaan hingga pedesaan dan pegunungan, jaringan jalan di Bali cukup beragam dan sering kali mengalami kemacetan, terutama di daerah wisata dan pusat kota.

Dalam konteks ini, penting untuk menerapkan metode pencarian rute terpendek antar rumah sakit agar pasien dapat dipindahkan atau dirujuk dengan cepat, mengurangi risiko keterlambatan penanganan medis. Hingga saat ini, rute transportasi antar rumah sakit di Bali sebagian besar masih ditentukan berdasarkan pengetahuan lapangan dan pengalaman pengemudi ambulans tanpa adanya pendekatan terorganisir yang berbasis algoritma untuk mencari rute terpendek. Dengan banyaknya rumah sakit yang tersebar di berbagai lokasi dan tantangan geografis yang ada, diperlukan solusi berbasis teknologi untuk mengoptimalkan rute antar rumah sakit.

Menemukan jalur tercepat antar rumah sakit di Bali dan memaksimalkan layanan transportasi medis dalam keadaan darurat dapat dicapai dengan menerapkan algoritma *A-Star*. Algoritma ini akan sangat berguna dalam membantu pihak rumah sakit dan petugas transportasi untuk menentukan rute yang paling efisien, terutama dalam kondisi lalu lintas yang dinamis dan jaringan jalan yang kompleks.

2.2 Menjaga Keselamatan dalam Ajaran Islam

Islam sangat menekankan pentingnya menjaga keselamatan diri dan orang lain, terutama dalam upaya penyelamatan jiwa dan kemaslahatan umat. Setiap individu diajarkan untuk menjaga dirinya agar tidak terjerumus dalam bahaya, baik secara fisik maupun sosial. Ajaran Islam mendorong umatnya untuk berbuat baik dan

saling membantu demi kebaikan bersama. Salah satu wujud nyata yang dapat dilakukan adalah membantu mempercepat akses layanan kesehatan bagi yang membutuhkan, seperti pasien darurat, dengan memanfaatkan teknologi modern untuk menentukan rute terpendek menuju rumah sakit.

Keselamatan ini juga tercermin dalam ajaran Al-Qur'an yang mengingatkan umat untuk tidak membahayakan diri sendiri atau orang lain. Sebagaimana dijelaskan dalam QS. Al-Isra' [17]: 33:

وَلَا تَقْتُلُوا النَّفْسَ الَّتِي حَرَّمَ اللَّهُ إِلَّا بِالْحَقِّ ۚ وَمَنْ قُتِلَ مَظْلُومًا فَقَدْ جَعَلْنَا لَوْلِيٍّ سُلْطَانًا فَلَا يَسْرِفُ فِي الْقَتْلِ ۚ إِنَّهُ كَانَ مَنْصُورًا

(Kementerian Agama, 2022)

Artinya: *"Dan janganlah kamu membunuh jiwa yang diharamkan Allah melainkan dengan haq. Dan barang siapa dibunuh secara zalim, maka sesungguhnya Kami telah memberi kekuasaan kepada walinya, tetapi janganlah keluarganya melampaui batas dalam membunuh. Sesungguhnya ia adalah orang yang dimenangkan."* (QS. Al-Isra': 33)

Menurut Tafsir Al-Misbah (Shihab, 2002), ayat ini menekankan larangan keras terhadap tindakan yang dapat merugikan jiwa manusia tanpa alasan yang sah, sekaligus memberikan hak kepada pihak yang dirugikan untuk memperoleh keadilan dengan syarat tidak melampaui batas. Ajaran ini mengajarkan pentingnya keseimbangan antara keadilan dan kasih sayang dalam setiap tindakan, termasuk dalam usaha untuk melindungi dan menyelamatkan nyawa manusia. Ayat ini mendorong penguatan sistem perlindungan jiwa, seperti melalui pemanfaatan teknologi untuk mempercepat penyelamatan dalam situasi darurat. Islam tidak hanya melarang pembunuhan, tetapi juga mengajarkan umatnya untuk menjaga dan melindungi nyawa dengan berbagai cara, seperti mencegah bahaya, menjaga kesehatan, dan mempercepat respons dalam keadaan darurat.

Salah satu tujuan utama syariat Islam adalah menjaga keselamatan jiwa (*hifz an-nafs*), yang merupakan bagian dari *maqashid syariah* (Harahap, 2021). Ajaran Islam menegaskan pentingnya upaya penyelamatan jiwa, baik untuk diri sendiri maupun orang lain. Tidak hanya itu, Islam juga mendorong penggunaan berbagai sarana yang dapat mendukung upaya perlindungan kehidupan, termasuk teknologi modern. Sebagai contoh, algoritma A-Star yang digunakan untuk menentukan rute terpendek antar rumah sakit merupakan salah satu bentuk aplikasi teknologi yang dapat mempercepat respons dalam situasi darurat. Dengan menghasilkan rute tercepat, algoritma ini memungkinkan kendaraan darurat seperti ambulans mencapai tujuan lebih cepat, sehingga meningkatkan peluang untuk menyelamatkan nyawa. Hal ini sejalan dengan ajaran Islam yang mengutamakan upaya untuk mengurangi risiko terhadap kehidupan manusia sekaligus mewujudkan kemaslahatan bagi umat.

Dengan memperhatikan *maqashid syariah*, setiap tindakan yang dilakukan untuk melindungi jiwa, seperti memanfaatkan teknologi modern untuk meningkatkan efisiensi layanan kesehatan, merupakan bagian dari usaha untuk mencapai kemaslahatan umat dan mencegah kerusakan. Islam mengajarkan umatnya untuk berempati dan peduli terhadap orang lain, sebagaimana tercermin dalam firman Allah SWT:

وَمَنْ أَحْيَاهَا فَكَأَنَّمَا أَحْيَا النَّاسَ جَمِيعًا

(Kementerian Agama, 2022)

Artinya: “Barang siapa yang memelihara kehidupan seorang manusia, dia seakan-akan telah memelihara kehidupan semua manusia.” (QS. Al-Ma'idah: 32).

Ayat ini menegaskan pentingnya menjaga kehidupan dan keselamatan manusia, yang tidak hanya berhubungan dengan pertolongan fisik tetapi juga dengan pemenuhan hak-hak dasar seperti kesehatan, pendidikan, dan kebutuhan

lainnya. Tindakan untuk menyelamatkan satu jiwa, baik melalui upaya preventif seperti memastikan akses kesehatan yang lebih mudah, maupun melalui tindakan responsif seperti membantu mempercepat layanan medis dalam situasi darurat, memiliki peran besar dalam menjaga kehidupan umat manusia. Dalam Islam, setiap individu yang berusaha menyelamatkan jiwa orang lain akan memperoleh pahala yang sangat besar di sisi Allah SWT.

Dengan demikian, menjaga keselamatan jiwa menjadi bagian yang sangat penting dalam ajaran Islam. Hal ini mencakup berbagai upaya nyata, seperti memanfaatkan teknologi untuk kemaslahatan umat dan mencegah kerusakan. Sebagai umat Islam, kita diajarkan untuk selalu berusaha menjaga keselamatan diri dan orang lain melalui tindakan-tindakan yang bermanfaat bagi masyarakat dan umat manusia secara keseluruhan.

2.3 Kajian Topik dengan Teori Pendukung

Penerapan Algoritma *A-Star* bertujuan untuk menganalisis jalur terpendek antar rumah sakit yang tersebar di Provinsi Bali. Dengan menggunakan Algoritma *A-Star*, diharapkan dapat membantu pihak rumah sakit, layanan ambulans, maupun pihak terkait dalam mengoptimalkan perjalanan antar rumah sakit, terutama dalam kondisi darurat di mana kecepatan dan ketepatan dalam menentukan rute sangat penting. Implementasi algoritma ini diharapkan juga mampu mengurangi waktu tempuh dan meminimalkan biaya operasional yang dikeluarkan oleh rumah sakit atau layanan medis.

Algoritma *A-Star* merupakan teknik pencarian jalur terpendek yang efektif karena menggunakan pendekatan heuristik dalam proses pencariannya. Langkah-langkah penerapannya meliputi penentuan titik koordinat rumah sakit yang menjadi

tujuan, menghitung jarak antar titik, dan menerapkan nilai heuristik $h(n)$ untuk memperkirakan jarak terpendek dari setiap *node* menuju *node* tujuan. Selanjutnya, nilai $g(n)$ yang merupakan biaya aktual perjalanan dari titik awal ke titik tujuan juga diperhitungkan. Nilai $f(n)$ kemudian dihitung sebagai penjumlahan dari $g(n)$ dan $h(n)$, yang akan menentukan *node* mana yang akan diprioritaskan dalam pencarian rute.

Dalam penelitian ini, algoritma *A-Star* akan dikaji lebih lanjut untuk menentukan rute terpendek antar rumah sakit di Bali. Ada berbagai metode yang dapat digunakan untuk mencari jalur terpendek, namun Algoritma *A-Star* dipilih karena kemampuannya dalam mengeliminasi langkah-langkah yang tidak diperlukan sehingga solusi yang ditemukan lebih cepat dan efisien. Implementasi algoritma ini menjadi sangat penting, khususnya dalam kasus-kasus darurat seperti kebutuhan pengiriman pasien antar rumah sakit yang harus dilakukan dengan segera.

Salah satu aspek penting dalam algoritma *A-Star* adalah penggunaan fungsi heuristik untuk memperkirakan jarak dari suatu titik ke tujuan. Dalam penelitian ini, akan dilakukan perbandingan antara dua jenis fungsi heuristik, yaitu fungsi heuristik *Euclidean* dan *Haversine*, untuk melihat pengaruhnya terhadap jalur yang dihasilkan oleh algoritma. Perbandingan ini bertujuan untuk mengetahui fungsi heuristik mana yang lebih efektif dalam pencarian rute berdasarkan jarak dalam satuan kilometer, tanpa mempertimbangkan waktu tempuh, karena waktu perjalanan sering kali tidak konsisten akibat faktor seperti kemacetan lalu lintas.

Algoritma *A-Star* memiliki nilai heuristik yang didapatkan dari tiap simpul, yang membantu menentukan jalur terpendek. Metode ini juga memperhitungkan

node-node yang belum dikunjungi dan membandingkannya berdasarkan perkiraan jarak dan biaya aktual. Dengan menggunakan data koordinat rumah sakit dan jaringan jalan di Bali, eksperimen untuk menentukan rute terpendek dapat dilakukan. Data yang diperoleh dari eksperimen ini digunakan untuk membandingkan hasil dari Algoritma *A-Star* dengan rute-rute yang dihasilkan oleh aplikasi seperti *Google Maps*.

Eksperimen ini bertujuan untuk menilai sejauh mana Algoritma *A-Star* dapat memberikan solusi yang lebih optimal dalam menentukan rute terpendek antar rumah sakit di Bali. Graf yang dihasilkan dari proses eksperimen akan menggambarkan jaringan antar rumah sakit dan menunjukkan jalur-jalur mana yang paling efisien untuk dilalui. Hasil dari eksperimen ini diharapkan dapat meningkatkan layanan kesehatan di Bali secara signifikan, terutama dalam konteks penanganan kasus darurat yang membutuhkan efisiensi waktu dan tenaga.

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Penelitian ini merupakan penelitian terapan dengan pendekatan kuantitatif yang bertujuan untuk mengimplementasikan algoritma *A-Star* dalam menentukan rute terpendek antar rumah sakit di Bali sekaligus membandingkan dua jenis fungsi heuristik, yaitu *Haversine* dan *Vincenty*. Penelitian ini dilakukan melalui dua pendekatan utama, yaitu studi literatur dan studi lapangan. Studi literatur digunakan untuk memperkuat landasan teori mengenai algoritma *A-Star*, konsep graf, serta pemahaman teknis terkait perhitungan heuristik. Sementara itu, studi lapangan dilakukan secara tidak langsung dengan memanfaatkan teknologi digital seperti *Google Maps* dan *Google Earth* untuk memperoleh data spasial berupa koordinat lokasi rumah sakit serta jaringan jalan yang menghubungkan antar rumah sakit. Dengan pendekatan ini, penelitian tidak hanya bertujuan untuk menerapkan algoritma *A-Star*, tetapi juga untuk menganalisis efektivitas dan akurasi masing-masing fungsi heuristik dalam proses pencarian rute terpendek berdasarkan data geografis yang valid.

3.2 Data dan Sumber Data

Data yang digunakan dalam penelitian ini termasuk data sekunder, berupa koordinat geografis (*latitude* dan *longitude*) rumah sakit awal dan tujuan serta jaringan jalan yang menghubungkan antar rumah sakit. Data diperoleh secara tidak langsung melalui *platform digital Google Maps*, yang menyediakan informasi spasial berupa titik koordinat dan jarak tempuh antar titik. Data tersebut kemudian diolah dan direpresentasikan dalam bentuk graf berarah berbobot, di mana setiap

simpul merepresentasikan lokasi rumah sakit dan persimpangan jalan, sedangkan bobot pada setiap sisi graf menunjukkan jarak dalam satuan kilometer antara dua simpul yang terhubung. Data yang telah dikumpulkan selanjutnya digunakan untuk membangun graf berbobot sebagai dasar implementasi algoritma *A-Star* dan analisis perbandingan kedua fungsi heuristik tersebut.

3.3 Tahapan Penelitian

Tahapan penelitian menggunakan algoritma *A-Star* tersusun dari beberapa langkah sistematis. Tahapan-tahapan yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Pengambilan Data Rute

Langkah awal dalam penelitian ini adalah melakukan pengambilan data rute yang akan dianalisis menggunakan algoritma *A-Star*. Data rute diperoleh dari *Google Maps* dalam bentuk tangkapan layar (*screenshot*) yang menunjukkan jalur penghubung antar rumah sakit di wilayah Bali. Titik-titik yang menjadi simpul dalam graf meliputi rumah sakit awal, rumah sakit tujuan, serta persimpangan jalan di sepanjang rute. Sisi (*edge*) dalam graf merepresentasikan jalur jalan berarah yang menghubungkan titik-titik tersebut, dengan bobot jarak antara simpul-simpul yang terhubung. Data rute ini menjadi dasar untuk membangun graf berbobot yang digunakan dalam penerapan algoritma *A-Star*.

2. Identifikasi dan Pengumpulan Data Koordinat

Tahapan selanjutnya dalam penelitian ini adalah mengidentifikasi dan mengumpulkan data koordinat dari lokasi rumah sakit serta titik-titik persimpangan jalan yang relevan, sekaligus mengumpulkan jarak antar titik

yang diperoleh dari *Google Maps*. Proses ini dimulai dengan menentukan koordinat geografis dari setiap titik yang akan direpresentasikan sebagai simpul dalam graf, yang diperoleh dari nilai *latitude* dan *longitude*. Dalam penelitian ini, terdapat 3 pasangan rumah sakit yang dipilih secara acak sebagai simpul awal dan simpul tujuan, yaitu A_1 ke T_1 , A_2 ke T_2 , dan A_3 ke T_3 . Simpul awal terdiri dari 3 rumah sakit, yaitu A_1, A_2 , dan A_3 , sedangkan simpul tujuan terdiri dari 3 rumah sakit, yaitu T_1, T_2 , dan T_3 . Selain itu, terdapat 65 simpul persimpangan jalan yang dilambangkan sebagai $J_1, J_2, J_3, \dots, J_{13}$ untuk pasangan rumah sakit pertama, $J_{14}, J_{15}, J_{16}, \dots, J_{38}$ untuk pasangan rumah sakit kedua, dan $J_{39}, J_{40}, J_{41}, \dots, J_{65}$ untuk pasangan rumah sakit ketiga. sehingga total simpul dalam graf berjumlah 71 simpul yang terdiri dari 3 rumah sakit awal, 3 rumah sakit tujuan, dan 65 persimpangan jalan. Seluruh koordinat dari simpul-simpul tersebut akan digunakan sebagai dasar dalam perhitungan nilai heuristik yang dibutuhkan oleh algoritma *A-Star* untuk menentukan rute terpendek antar rumah sakit. Pengelompokan simbol ini dilakukan agar proses pemodelan graf dan analisis rute dapat dilakukan dengan lebih sistematis dan terstruktur.

3. Perhitungan Fungsi Heuristik

Perhitungan fungsi heuristik merupakan tahap penting dalam algoritma *A-Star* untuk menentukan jarak perkiraan antara satu simpul dengan simpul tujuan pada graf. Fungsi heuristik ini berperan sebagai estimasi jarak yang memandu algoritma dalam mencari rute terpendek secara efisien dan akurat. Dalam penelitian ini, digunakan dua jenis fungsi heuristik yang akan dibandingkan keakuratannya yaitu fungsi heuristik *Haversine* dan *Vincenty*.

4. Implementasi Algoritma A-*Star*

Setelah data koordinat rumah sakit dan titik-titik persimpangan jalan terkumpul, langkah selanjutnya adalah menerapkan algoritma A-*Star* untuk menemukan rute terpendek antar rumah sakit. Algoritma ini bekerja dengan memadukan dua komponen utama, yaitu jarak aktual yang telah ditempuh dari simpul awal $g(n)$ dan estimasi jarak dari simpul awal ke tujuan $h(n)$ berdasarkan fungsi heuristik.

Dalam penelitian ini, proses pencarian rute dilakukan untuk beberapa pasangan rumah sakit yang dipilih sebagai simpul awal dan tujuan. Algoritma A-*Star* dijalankan dua kali untuk setiap pasangan, masing-masing menggunakan dua fungsi heuristik yang berbeda, yaitu fungsi heuristik *Haversine* dan *Vincenty*.

Langkah-langkah dalam implementasi algoritma A-*Star* adalah sebagai berikut:

- a. Tentukan simpul awal dan simpul tujuan.
- b. Inisialisasi nilai $g(n)$, yaitu jarak yang telah ditempuh dari simpul awal ke simpul n .
- c. Hitung nilai heuristik $h(n)$ menggunakan salah satu dari dua fungsi heuristik (*Haversine* atau *Vincenty*).
- d. Hitung nilai $f(n)$ untuk setiap simpul yang terhubung menggunakan rumus pada persamaan (2.1).
- e. Tempatkan simpul awal ke *open list* (daftar simpul yang akan dievaluasi), dan kosongkan *closed list* (daftar simpul yang sudah dievaluasi).
- f. Selama *open list* tidak kosong, lakukan langkah berikut:

- (1) Pilih simpul dalam *open list* dengan nilai $f(n)$ terkecil.
- (2) Pindahkan simpul tersebut ke dalam *closed list*.
- (3) Untuk setiap tetangga dari simpul tersebut:
 - a) Jika tetangga sudah ada di *closed list*, abaikan.
 - b) Hitung nilai sementara $g(n)$ dari simpul awal ke tetangga melalui simpul saat ini.
 - c) Jika tetangga belum ada di *open list*, atau $g(n)$ baru lebih kecil, perbarui nilai $g(n)$, $h(n)$, dan $f(n)$ tetangga tersebut.
 - d) Tambahkan tetangga ke *open list* jika belum ada.
- g. Ulangi proses hingga simpul tujuan ditemukan atau *open list* kosong (menandakan tidak ada jalur yang tersedia).
- h. Simpan hasil pencarian berupa:
 - (1) Jalur rute terpendek ditemukan.
 - (2) Total jarak yang ditempuh berdasarkan nilai $f(n)$
 - (3) Jumlah simpul yang dilalui dalam proses pencarian.

Proses ini dilakukan dua kali untuk setiap pasangan rumah sakit: satu kali menggunakan fungsi heuristik *Haversine* dan satu kali menggunakan *Vincenty*. Hal ini memungkinkan perbandingan langsung hasil rute, jarak total, serta efisiensi pencarian dari kedua fungsi heuristik tersebut.

5. Perbandingan Fungsi Heuristik

Setelah implementasi algoritma *A-Star* menggunakan fungsi heuristik *Haversine* dan *Vincenty* dilakukan, langkah selanjutnya adalah membandingkan hasil dari masing-masing metode heuristik. Perbandingan

dilakukan untuk melihat seberapa efisien dan akurat masing-masing fungsi heuristik dalam menentukan rute terpendek antar rumah sakit di Bali.

Perbandingan ini dilakukan untuk mengevaluasi apakah perbedaan metode heuristik memberikan pengaruh signifikan terhadap hasil akhir rute terpendek dan efisiensi pencarian. Dengan membandingkan total jarak rute dan jumlah simpul yang dilalui, penelitian ini dapat menentukan fungsi heuristik mana yang lebih efektif dalam pencarian rute antar rumah sakit di Bali, tanpa mempertimbangkan waktu eksekusi karena batasan penelitian terkait kondisi lalu lintas yang tidak stabil.

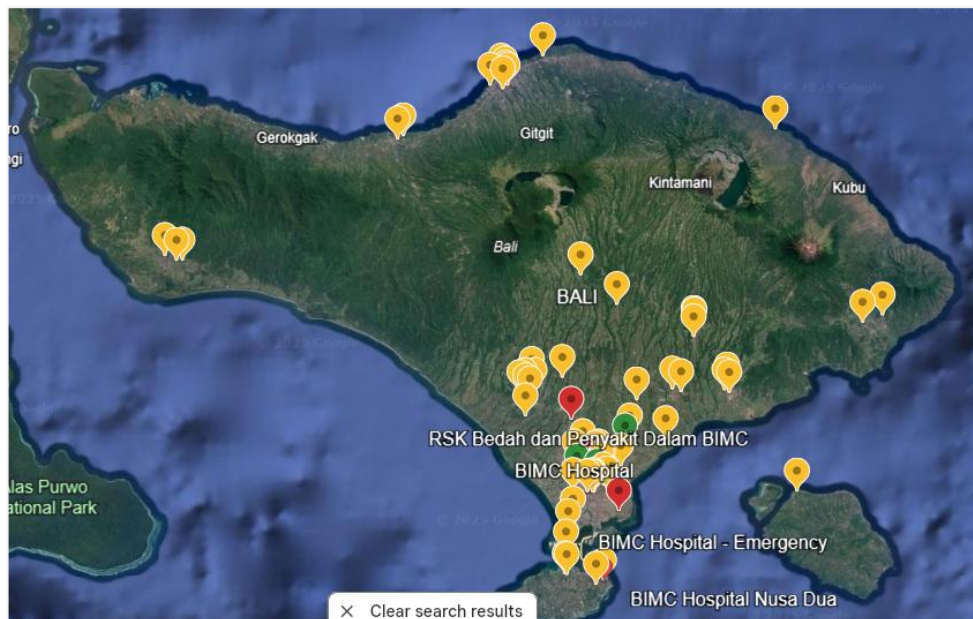
BAB IV

HASIL DAN PEMBAHASAN

4.1 Analisis Data

Penelitian ini menggunakan data koordinat geografis rumah sakit di wilayah Bali sebagai dasar dalam pemodelan graf untuk pencarian rute terpendek menggunakan algoritma A-*Star*. Di Bali terdapat sebanyak lebih dari 69 rumah sakit yang tersebar di berbagai kabupaten dan kota. Namun, dalam penelitian ini tidak semua rumah sakit digunakan sebagai titik awal dan titik tujuan, melainkan dipilih secara acak beberapa rumah sakit untuk dianalisis.

Berikut ini adalah daftar lengkap dari berbagai titik lokasi rumah sakit yang berada di wilayah Bali. Daftar ini diperoleh setelah melakukan pencarian mendetail dengan menggunakan *Google Maps* dan *Google Earth*, yang membantu dalam mengidentifikasi dan memastikan keberadaan lokasi-lokasi rumah sakit tersebut untuk keperluan penelitian ini.



Gambar 4.1 *Screenshoot* data lokasi rumah sakit di Bali

Dalam Gambar 4.1, titik-titik berwarna menunjukkan lokasi-lokasi rumah sakit di wilayah Bali. Titik-titik berwarna merah dan hijau merupakan rumah sakit yang telah dipilih untuk penelitian ini, sedangkan titik-titik berwarna kuning menunjukkan rumah sakit lain yang tidak termasuk dalam penelitian. Titik-titik merah menandai rumah sakit yang dijadikan sebagai simpul awal, sedangkan titik-titik hijau menandai rumah sakit yang dijadikan sebagai simpul tujuan. Penandaan warna ini memberikan gambaran yang jelas tentang cakupan lokasi rumah sakit yang digunakan dalam penelitian serta fokus analisis dalam konteks pencarian rute terpendek antar rumah sakit di Bali.

4.2 Simpul Awal dan Simpul Tujuan

Dalam penelitian ini, pemilihan simpul awal dan simpul tujuan dilakukan untuk menguji algoritma *A-Star* dalam pencarian rute terpendek antar rumah sakit di Bali. Dari sekian banyak rumah sakit yang ada di wilayah Bali, hanya diambil tiga pasangan rumah sakit secara acak sebagai sampel yang akan diteliti. Tidak semua rumah sakit dijadikan objek analisis karena keterbatasan ruang lingkup dan fokus penelitian. Pemilihan secara acak dilakukan tanpa mempertimbangkan kriteria tertentu, seperti tipe rumah sakit, tingkat urgensi, atau volume pasien, namun tetap memperhatikan penyebaran lokasi rumah sakit agar analisis tetap relevan. Meskipun jumlah pasangan rumah sakit yang digunakan terbatas, jumlah sampel tersebut telah disesuaikan dengan tujuan penelitian yang berfokus pada pengujian algoritma *A-Star* dalam menentukan rute terpendek, bukan untuk melakukan pemetaan lengkap terhadap seluruh rute antar rumah sakit di Bali.

Tabel 4.1 merupakan rumah sakit yang digunakan sebagai simpul awal dan simpul tujuan, yaitu sebagai berikut:

Tabel 4.1 Daftar Pasangan RS Awal dan RS Tujuan di Bali

No	RS Awal	Simbol RS Awal	RS Tujuan	Simbol RS Tujuan
1	RSU Balimed (Golongan C)	A_1	RSUD Wangaya (Golongan B)	T_1
2	RSU Premagana (Golongan C)	A_2	RSD Mangusada (Golongan B)	T_2
3	RS TK. II Udayana (Golongan C)	A_3	RSUD Bali Mandara (Golongan B)	T_3

Untuk mendukung proses perhitungan rute terpendek, diperlukan data koordinat geografis dari rumah sakit yang menjadi simpul awal dan tujuan. Koordinat ini akan digunakan sebagai simpul pada graf dalam algoritma *A-Star*. Tabel 4.2 adalah data *latitude* dan *longitude* dari rumah sakit awal dan tujuan.

Tabel 4.2 Koordinat Simpul Awal dan Simpul Tujuan

Simbol	Nama RS	<i>Latitude</i>	<i>Longitude</i>
A_1	RSU Balimed	-8,656791	115,190929
A_2	RSU Premagana	-8,614971	115,257351
A_3	RS TK. II Udayana	-8,663486	115,218641
T_1	RSUD Wangaya	-8,648779	115,213433
T_2	RSD Mangusada	-8,578390	115,182946
T_3	RSUD Bali Mandara	-8,703646	115,248806

Data koordinat pada Tabel 4.2 diperoleh dari *Google Maps* dan menjadi dasar dalam membangun graf berbobot serta perhitungan fungsi heuristik pada tahap selanjutnya.

4.3 Perhitungan Fungsi Heuristik dan Implementasi Algoritma *A-Star*

Pada tahap ini, dilakukan perhitungan fungsi heuristik dan penerapan algoritma *A-Star* untuk mencari rute terpendek antara pasangan simpul awal dan simpul tujuan yang telah ditentukan sebelumnya. Fungsi heuristik yang digunakan dalam penelitian ini adalah fungsi heuristik *Haversine* dan *Vincenty*, yang berfungsi untuk memperkirakan jarak antara simpul saat ini dengan simpul tujuan dalam graf.

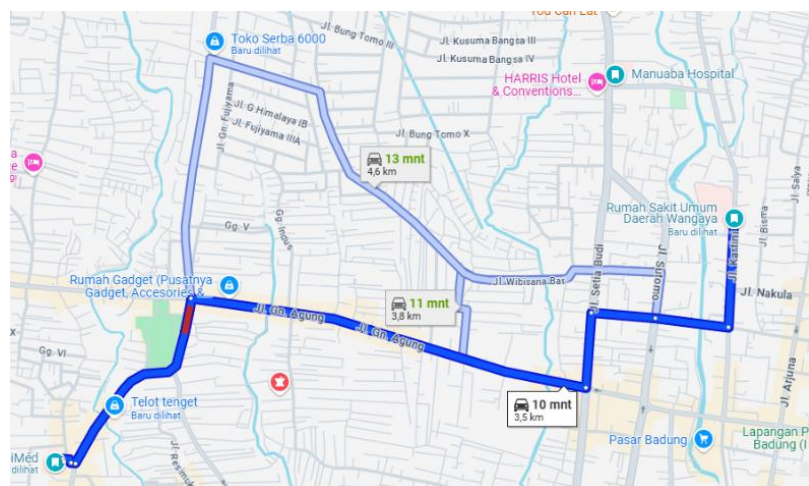
Penelitian ini menguji tiga pasangan rumah sakit sebagai sampel, yaitu:

1. Penentuan Rute $A_1 \rightarrow T_1$
2. Penentuan Rute $A_2 \rightarrow T_2$
3. Penentuan Rute $A_3 \rightarrow T_3$

Setiap rute akan dianalisis secara terperinci melalui tahapan berikut: pengumpulan data titik koordinat rumah sakit awal dan tujuan serta titik persimpangan jalan, perhitungan fungsi heuristik (metode *Haversine* dan *Vincenty*), implementasi algoritma *A-Star*, perbandingan algoritma *A-Star* menggunakan metode *Haversine* dan *Vincenty*, dan analisis hasil. Adapun perhitungan rute terpendek menggunakan algoritma *A-Star*, yaitu sebagai berikut:

1. Penentuan Rute $A_1 \rightarrow T_1$
 - a. Pengumpulan Data Titik dan Titik Persimpangan

Pada tahap awal, dilakukan visualisasi peta yang menampilkan lokasi rumah sakit awal (RSU Balimed), rumah sakit tujuan (RSUD Wangaya), serta titik-titik persimpangan jalan yang menjadi jalur penghubung antar kedua rumah sakit tersebut. Gambar 4.2 berikut menunjukkan peta rute yang menghubungkan RSU Balimed ke RSUD Wangaya beserta titik-titik persimpangan jalan utama yang dilalui.



Gambar 4.2 Peta Rute dari RSU Balimed ke RSUD Wangaya

Selanjutnya, koordinat *latitude* dan *longitude* dari rumah sakit awal, rumah sakit tujuan, serta titik-titik persimpangan jalan tersebut diidentifikasi dan disajikan dalam Tabel 4.3, yang digunakan sebagai dasar pemodelan graf dan perhitungan fungsi heuristik.

Tabel 4.3 RS Awal (RSU Balimed) ke RS Tujuan (RSUD Wangaya)

No	Nama Lokasi	Simbol	Latitude	Longitude
1	RSU Balimed	A_1	-8,656791	115,190929
2	RSUD Wangaya	T_1	-8,648779	115,213433

Selanjutnya, titik-titik koordinat persimpangan jalan beserta simbolnya seperti pada Tabel 4.4.

Tabel 4.4 Titik-titik Persimpangan Jalan (RSU Balimed ke RSUD Wangaya)

Simbol	Latitude	Longitude	Simbol	Latitude	Longitude
J_1	-8,656874	115,191602	J_8	-8,650567	115,208876
J_2	-8,651537	115,195444	J_9	-8,650636	115,210993
J_3	-8,643664	115,196089	J_{10}	-8,654456	115,208534
J_4	-8,653478	115,204416	J_{11}	-8,651968	115,208723
J_5	-8,644745	115,199734	J_{12}	-8,652119	115,210811
J_6	-8,650553	115,204403	J_{13}	-8,652438	115,213279
J_7	-8,650553	115,208147			

Daftar nama jalan yang menghubungkan simpul-simpul tersebut disajikan dalam Tabel 4.5 dan digunakan sebagai jalur dalam graf berarah berbobot.

Tabel 4.5 Nama Jalan Antar Persimpangan (RSU Balimed ke RSUD Wangaya)

Dari	Ke	Nama Jalan	Dari	Ke	Nama Jalan
A_1	J_1	Jl. Mahendradatta Utara	J_7	J_8	Jl. Wibisana
J_1	J_2	Jl. Mahendradatta Utara	J_8	J_9	Jl. Sri Rama
J_2	J_3	Jl. Buluh Indah	J_9	J_{12}	Jl. Sutomo
J_2	J_4	Jl. Gn. Agung	J_{10}	J_{11}	Jl. Setia Budi
J_3	J_5	Jl. Gn. Himalaya	J_{11}	J_8	Jl. Setia Budi
J_4	J_6	Jl. Merthayasa	J_{11}	J_{12}	Jl. Gabuh
J_4	J_{10}	Jl. Gn. Agung	J_{12}	J_{13}	Jl. Kumbakarna
J_5	J_6	Jl. Wibisana Bar	J_{13}	T_1	Jl. Kartini
J_6	J_7	Jl. Wibisana Bar			

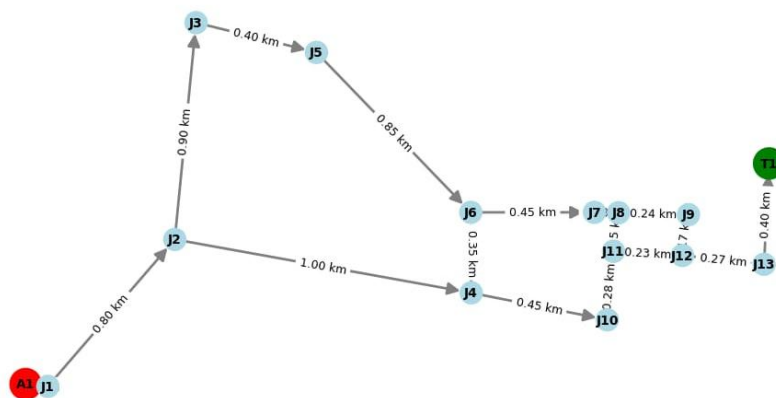
Tabel 4.6 adalah data jarak aktual ($g(n)$) dari RSU Balimed ke RSUD Wangaya berdasarkan *Google Maps*.

Tabel 4.6 Data Jarak Aktual ($g(n)$) (RSU Balimed ke RSUD Wangaya)

Titik 1	Titik 2	Jarak (km)	Titik 1	Titik 2	Jarak (km)
A_1	J_1	0,07	J_7	J_8	0,08
J_1	J_2	0,80	J_8	J_9	0,24
J_2	J_3	0,90	J_9	J_{12}	0,17
J_2	J_4	1,00	J_{10}	J_{11}	0,28
J_3	J_5	0,40	J_{11}	J_8	0,15
J_4	J_6	0,35	J_{11}	J_{12}	0,23
J_4	J_{10}	0,45	J_{12}	J_{13}	0,27
J_5	J_6	0,85	J_{13}	T_1	0,40
J_6	J_7	0,45			

b. Perhitungan Fungsi Heuristik

Berikut merupakan representasi dalam bentuk graf berarah berbobot yang dibentuk berdasarkan peta rute dari RSU Balimed menuju RSUD Wangaya. Pada visualisasi ini, semua titik persimpangan ditandai dengan warna biru, sementara titik rumah sakit awal yang dipilih diwarnai merah, sedangkan titik rumah sakit tujuannya diwarnai hijau. Garis-garis dengan panah menunjukkan jalur perjalanan dan menandakan graf tersebut adalah graf berarah.



Gambar 4.3 Graf Berarah Berbobot (RSU Balimed ke RSUD Wangaya)

Selain itu, pada graf juga ditampilkan bobot berupa jarak aktual ($g(n)$) dalam satuan kilometer yang merepresentasikan jarak antar titik pada setiap sisi graf, sehingga memudahkan pemahaman mengenai panjang rute antar simpul berdasarkan data nyata.

Perhitungan Heuristik *Haversine*

$$h(n) = 2r \times \arcsin \left(\sqrt{\sin^2 \left(\frac{\Delta \varphi}{2} \right) + \cos(\varphi_1) \times \cos(\varphi_2) \times \sin^2 \left(\frac{\Delta \lambda}{2} \right)} \right)$$

Diketahui:

$$A_1 \text{ (Simpul awal): } \varphi_1 = -8,656791, \lambda_1 = 115,190929$$

$$T_1 \text{ (Simpul tujuan): } \varphi_2 = -8,648779, \lambda_2 = 115,213433$$

Penyelesaian:

(1) Konversi derajat desimal ke radian

$$\text{a. } A_1: \varphi_1 = -8,656791 \times \frac{\pi}{180} = -0,15108951 \text{ rad}$$

$$\text{b. } A_1: \lambda_1 = 115,190929 \times \frac{\pi}{180} = 2,01046098 \text{ rad}$$

$$\text{c. } T_1: \varphi_2 = -8,648779 \times \frac{\pi}{180} = -0,15094967 \text{ rad}$$

$$\text{d. } T_1: \lambda_2 = 115,213433 \times \frac{\pi}{180} = 2,01085375 \text{ rad}$$

(2) Hitung perbedaan bujur ($\Delta \varphi$) dan perbedaan lintang ($\Delta \lambda$)

$$\begin{aligned} \text{a. } \Delta \varphi &= \varphi_2 - \varphi_1 = (-0,15094967) - (-0,15108951) = \\ &0,00013984 \text{ rad} \end{aligned}$$

$$\begin{aligned} \text{b. } \Delta \lambda &= \lambda_2 - \lambda_1 = (2,01085375) - (2,01046098) = \\ &0,00039277 \text{ rad} \end{aligned}$$

(3) Hitung $\sin^2\left(\frac{\Delta\varphi}{2}\right)$ dan $\sin^2\left(\frac{\Delta\lambda}{2}\right)$

$$\text{a. } \sin\left(\frac{\Delta\varphi}{2}\right) = \sin\left(\frac{0,00013984}{2}\right) = \sin(0,00006992)$$

$$\text{b. } \sin^2\left(\frac{\Delta\varphi}{2}\right) = (0,00006992)^2 = 4,89 \times 10^{-9}$$

$$\text{c. } \sin\left(\frac{\Delta\lambda}{2}\right) = \sin\left(\frac{0,00039277}{2}\right) = \sin(0,000196385)$$

$$\text{d. } \sin^2\left(\frac{\Delta\lambda}{2}\right) = (0,000196385)^2 = 3,86 \times 10^{-8}$$

(4) Hitung a dalam rumus *haversine*

$$\text{a. } \cos \varphi_1 = \cos(-8,656791) = 0,9886076774$$

$$\text{b. } \cos \varphi_2 = \cos(-8,648779) = 0,9886287152$$

$$\begin{aligned} \text{c. } a &= \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos \varphi_1 \times \cos \varphi_2 \times \sin^2\left(\frac{\Delta\lambda}{2}\right) \\ &= 4,89 \times 10^{-9} + 0,9886076774 \times 0,9886287152 \times \\ &\quad 3,86 \times 10^{-8} \\ &= 4,89 \times 10^{-9} + 3,77 \times 10^{-8} \\ &= 4,26 \times 10^{-8} \end{aligned}$$

(5) Hitung c

$$\begin{aligned} c &= 2 \times \arcsin(\sqrt{a}) \\ &= 2 \times \arcsin\left(\sqrt{4,26 \times 10^{-8}}\right) \\ &= 2 \times \arcsin(0,00020639767) \\ &= 2 \times 0,00020639767 \\ &= 0,00041279534 \text{ rad} \end{aligned}$$

(6) Hitung jarak d dengan radius bumi $R = 6371 \text{ km}$

$$d = 6371 \times c = 6371 \times 0,00041279534 = 2,63 \text{ km}$$

Tabel 4.7 adalah data jarak antar titik menggunakan fungsi heuristik *Haversine* yang dihitung menggunakan *google collab*.

Tabel 4.7 Data Heuristik Menggunakan Formula Haversine (RSU Balimed ke RSUD Wangaya)

Titik 1	Titik 2	Jarak (km)	Titik 1	Titik 2	Jarak (km)
A_1	T_1	2,63	J_7	T_1	0,61
J_1	T_1	2,56	J_8	T_1	0,53
J_2	T_1	2,00	J_9	T_1	0,33
J_3	T_1	1,98	J_{10}	T_1	0,82
J_4	T_1	1,12	J_{11}	T_1	0,62
J_5	T_1	1,57	J_{12}	T_1	0,47
J_6	T_1	1,01	J_{13}	T_1	0,40

Berdasarkan perhitungan jarak menggunakan rumus *Haversine* yang ditampilkan pada Tabel 4.7, langkah selanjutnya adalah membuktikan bahwa nilai heuristik tersebut memenuhi sifat *consistent* dan *admissible*.

Pembuktian Consistency Heuristik *Haversine*

Heuristik dikatakan *consistent* jika untuk setiap pasangan simpul bertetangga n dan n' , nilai heuristik pada simpul n tidak melebihi jumlah biaya pindah dari n ke n' , secara matematis:

$$h(n) \leq c(n, n') + h(n')$$

Untuk membuktikan sifat konsistensi heuristik *Haversine*, dilakukan perbandingan nilai heuristik pada setiap simpul dengan penjumlahan jarak aktual antar simpul bertetangga dan nilai heuristik simpul tetangga, sebagaimana disajikan pada Tabel 4.8. Dari Tabel 4.8, semua nilai $h(n)$ memenuhi kondisi konsistensi $h(n) \leq c(n, n') + h(n')$. Dengan demikian, heuristik ini konsisten (*consistent*).

Tabel 4.8 Perhitungan Konsistensi Heuristik *Haversine* (RSU Balimed ke RSUD Wangaya)

Pasangan Simpul ($n \rightarrow n'$)	$h(n)$ (km)	$c(n, n')$ (km)	$h(n')$ (km)	$c(n, n') + h(n')$ (km)	Konsisten? (Apakah $h(n) \leq c(n, n') + h(n')$?)
$A_1 \rightarrow J_1$	2,63	0,07	2,56	2,63	Ya, $2,63 \leq 2,63$
$J_1 \rightarrow J_2$	2,56	0,80	2,00	2,80	Ya, $2,56 \leq 2,80$
$J_2 \rightarrow J_3$	2,00	0,90	1,98	2,88	Ya, $2,00 \leq 2,88$
$J_2 \rightarrow J_4$	2,00	1,00	1,12	2,12	Ya, $2,00 \leq 2,12$
$J_3 \rightarrow J_5$	1,98	0,40	1,58	1,98	Ya, $1,98 \leq 1,98$
$J_4 \rightarrow J_6$	1,12	0,35	1,01	1,36	Ya, $1,12 \leq 1,36$
$J_4 \rightarrow J_{10}$	1,12	0,45	0,82	1,27	Ya, $1,12 \leq 1,27$
$J_5 \rightarrow J_6$	1,58	0,85	1,01	1,86	Ya, $1,58 \leq 1,86$
$J_6 \rightarrow J_7$	1,01	0,45	0,61	1,06	Ya, $1,01 \leq 1,06$
$J_7 \rightarrow J_8$	0,61	0,08	0,53	0,61	Ya, $0,61 \leq 0,61$
$J_8 \rightarrow J_9$	0,53	0,24	0,33	0,57	Ya, $0,53 \leq 0,57$
$J_9 \rightarrow J_{12}$	0,33	0,17	0,47	0,64	Ya, $0,33 \leq 0,64$
$J_{10} \rightarrow J_{11}$	0,82	0,28	0,62	0,90	Ya, $0,82 \leq 0,90$
$J_{11} \rightarrow J_8$	0,62	0,15	0,53	0,68	Ya, $0,62 \leq 0,68$
$J_{11} \rightarrow J_{12}$	0,62	0,23	0,47	0,70	Ya, $0,62 \leq 0,70$
$J_{12} \rightarrow J_{13}$	0,47	0,27	0,40	0,67	Ya, $0,47 \leq 0,67$
$J_{13} \rightarrow T_1$	0,40	0,40	0,00	0,40	Ya, $0,40 \leq 0,40$

Pembuktian *Admissibility* Heuristik *Haversine*

Setelah pembuktian bahwa heuristik *Haversine* memenuhi sifat konsistensi, langkah berikutnya adalah membuktikan bahwa heuristik tersebut juga *admissible*. Heuristik dikatakan *admissible* jika nilai heuristik pada setiap simpul tidak pernah melebihi jarak aktual terpendek dari simpul tersebut ke tujuan akhir, yaitu:

$$h(n) \leq g^*(n)$$

di mana $g^*(n)$ adalah jarak aktual terpendek dari simpul n ke tujuan. Untuk membuktikan *admissibility* heuristik *Haversine*, dilakukan perbandingan nilai heuristik dengan jarak aktual terpendek yang dihitung berdasarkan data jarak antar simpul bertetangga dan rute terpendek menuju tujuan T_1 .

Perbandingan antara nilai heuristik dan jarak aktual terpendek dari setiap simpul ke simpul tujuan disajikan pada Tabel 4.9.

Tabel 4.9 Perhitungan *Admissible* Heuristik *Haversine* (RSU Balimed ke RSUD Wangaya)

Pasangan Simpul	Heuristik <i>Haversine</i> $h(n)$ (km)	Jarak Aktual Terpendek $g^*(n)$ (km)	<i>Admissible?</i> (Apakah $h(n) \leq g^*(n)$?)
$A_1 \rightarrow T_1$	2,63	3,50	Ya, $2,63 \leq 3,50$
$J_1 \rightarrow T_1$	2,56	3,43	Ya, $2,56 \leq 3,43$
$J_2 \rightarrow T_1$	2,00	2,63	Ya, $2,00 \leq 2,63$
$J_3 \rightarrow T_1$	1,98	2,86	Ya, $1,98 \leq 2,86$
$J_4 \rightarrow T_1$	1,12	1,63	Ya, $1,12 \leq 1,63$
$J_5 \rightarrow T_1$	1,58	2,46	Ya, $1,58 \leq 2,46$
$J_6 \rightarrow T_1$	1,01	1,61	Ya, $1,01 \leq 1,61$
$J_7 \rightarrow T_1$	0,61	1,16	Ya, $0,61 \leq 1,16$
$J_8 \rightarrow T_1$	0,53	1,08	Ya, $0,53 \leq 1,08$
$J_9 \rightarrow T_1$	0,33	0,84	Ya, $0,33 \leq 0,84$
$J_{10} \rightarrow T_1$	0,82	1,18	Ya, $0,82 \leq 1,18$
$J_{11} \rightarrow T_1$	0,62	0,90	Ya, $0,62 \leq 0,90$
$J_{12} \rightarrow T_1$	0,47	0,67	Ya, $0,47 \leq 0,67$
$J_{13} \rightarrow T_1$	0,40	0,40	Ya, $0,40 \leq 0,40$

Dari Tabel 4.9, terlihat bahwa untuk semua simpul, nilai heuristik *Haversine* tidak pernah melebihi jarak aktual terpendek ke tujuan. Dengan demikian, heuristik ini memenuhi syarat *admissible*.

Jadi, nilai heuristik *Haversine* yang digunakan pada algoritma pencarian rute ini terbukti *consistent* (Tabel 4.8) dan *admissible* (Tabel 4.9). Kedua sifat ini menjamin bahwa algoritma A-Star yang menggunakan heuristik ini dapat menemukan jalur terpendek secara optimal dan efisien.

Perhitungan Heuristik *Vincenty*

Diketahui:

$$A_1 \text{ (Simpul awal): } \varphi_1 = -8,656791, \lambda_1 = 115,190929$$

$$T_1 \text{ (Simpul tujuan): } \varphi_2 = -8,648779, \lambda_2 = 115,213433$$

Penyelesaian:

(1) Parameter *Ellipsoid*

$$a = 6378137,0 \text{ m}$$

$$f = \frac{1}{298,257223563} \approx 0,0033528$$

$$b = a \times (1 - f) = 6356752,314245 \text{ m}$$

(2) Konversi ke radian

$$\varphi_1 = -8,656791 \times \frac{\pi}{180} = -0,1511036033 \text{ rad}$$

$$\lambda_1 = 115,190929 \times \frac{\pi}{180} = 2,0101354906 \text{ rad}$$

$$\varphi_2 = -8,648779 \times \frac{\pi}{180} = -0,1509650123 \text{ rad}$$

$$\lambda_2 = 115,213433 \times \frac{\pi}{180} = 2,0105295467 \text{ rad}$$

$$L = \lambda_2 - \lambda_1 = 2,0105295467 - 2,0101354906 = 0,0003940561 \text{ rad}$$

(3) Hitung *reduce latitude*

$$\begin{aligned} U_1 &= \arctan((1 - f) \times \tan \varphi_1) \\ &= \arctan(0,9966471893 \times \tan(-0,1511036033)) \\ &= -0,1505905675 \text{ rad} \end{aligned}$$

$$\begin{aligned} U_2 &= \arctan((1 - f) \times \tan \varphi_2) \\ &= \arctan(0,9966471893 \times \tan(-0,1509650123)) \\ &= -0,1504511795 \text{ rad} \end{aligned}$$

(4) Iterasi pertama untuk λ

$$\lambda = L = 0,0003940561 \text{ rad}$$

Langkah 1: Hitung $\sin U_1, \sin U_2, \cos U_1, \cos U_2$

$$\sin U_1 = \sin(-0,1505905675) = -0,149996025$$

$$\sin U_2 = \sin(-0,1504511795) = -0,149857358$$

$$\cos U_1 = \cos(-0,1505905675) = 0,98868040$$

$$\cos U_2 = \cos(-0,1504511795) = 0,988706473$$

Langkah 2: Hitung $\sin \lambda, \cos \lambda$

$$\sin \lambda = \sin(0,0003940561) = 0,0003940560$$

$$\cos \lambda = \cos(0,0003940561) = 0,9999999224$$

Langkah 3: Hitung $\sin \sigma$

$$\begin{aligned} \sin^2 \sigma &= (\cos U_2 \times \sin \lambda)^2 + (\cos U_1 \times \sin U_2 - \sin U_1 \times \cos U_2 \times \cos \lambda)^2 \\ &= (0,988706473 \times 0,0003940560)^2 + (0,98868040 \times (-0,149857358) - (-0,149996025) \times 0,988706473 \times 0,9999999224)^2 \\ &= (0,000389504)^2 + (0,000090997)^2 \\ &= 1,5990 \times 10^{-7} \end{aligned}$$

$$\sin \sigma = \sqrt{1,5990 \times 10^{-7}} = 0,0004125864$$

Langkah 4: Hitung $\cos \sigma$

$$\begin{aligned} \cos \sigma &= \sin U_1 \times \sin U_2 + \cos U_1 \times \cos U_2 \times \cos \lambda \\ &= (-0,149996025 \times (-0,149857358)) + (0,98868040 \times 0,988706473 \times 0,9999999224) \\ &= 0,022457806 + 0,977542109 \\ &= 0,999999915 \end{aligned}$$

Langkah 5: Hitung σ

$$\begin{aligned} \sigma &= \arctan\left(\frac{\sin \sigma}{\cos \sigma}\right) \\ &= \arctan\left(\frac{0,0004125864}{0,999999915}\right) \end{aligned}$$

$$= 0,0004125865 \text{ rad}$$

Langkah 6: Hitung $\sin \alpha$

$$\begin{aligned} \sin \alpha &= \frac{\cos U_1 \times \cos U_2 \times \sin \lambda}{\sin \sigma} \\ &= \frac{0,98868040 \times 0,988706473 \times 0,0003940560}{0,0004125864} \\ &= \frac{0,000385846}{0,0004125864} \\ &= 0,930561597 \end{aligned}$$

Langkah 7: $\cos^2 \alpha$

$$\cos^2 \alpha = 1 - \sin^2 \alpha = 1 - (0,930561597)^2 = 0,133055114$$

Langkah 8: Hitung $\cos(2\sigma_m)$

$$\begin{aligned} \cos(2\sigma_m) &= \cos \sigma - \frac{2 \times \sin U_1 \times \sin U_2}{\cos^2 \alpha} \\ &= 0,999999915 - \frac{2 \times (-0,149996025) \times (-0,149857358)}{0,133055114} \\ &= 0,999999915 - \frac{0,044915}{0,113055114} \\ &= 0,66452687 \end{aligned}$$

Langkah 9: Hitung C

$$\begin{aligned} C &= \frac{f}{16} \times \cos^2 \alpha \times (4 + f \times (4 - 3 \times \cos^2 \alpha)) \\ &= \frac{0,0033528107}{16} \times 0,133055114 \times (4 + 0,0033528107 \times (4 \\ &\quad - 3 \times 0,133055114)) \\ &= 0,000112704 \end{aligned}$$

(5) Iterasi 2: Perbarui nilai λ

$$\begin{aligned} \lambda_{baru} &= L + (1 - C) \times f \times \sin \alpha \times [\sigma + C \times \sin \sigma \times (\cos(2\sigma_m) \\ &\quad + C \times \cos \sigma \times (-1 + 2 \times \cos^2(2\sigma_m)))] \\ &= 0,0003940561 + (1 - 0,000112704) \times \end{aligned}$$

$$\begin{aligned}
& 0,0033528107 \times 0,930561597 \times [0,0004125865 + \\
& 0,000112704 \times 0,0004125864 \times (0,66452687 + \\
& 0,000112704 \times 0,999999915 \times (-1 + 2 \times \\
& (0,66452687)^2))] \\
& = 0,0003940561 + (1 - 0,000112704) \times \\
& 0,0033528107 \times 0,930561597 \times 0,0004126174 \\
& = 0,0003940561 + 0,00000128796 \\
& = 0,000394060 \text{ rad}
\end{aligned}$$

Hasil iterasi 2:

$$\lambda_{baru} = 0,0003940561 \text{ rad}$$

Cek perbedaan dari iterasi sebelumnya:

$$\begin{aligned}
|\lambda_{baru} - \lambda_{lama}| &= |0,0003940561 - 0,0003940561| \\
&= 1,288 \times 10^{-6} \text{ rad}
\end{aligned}$$

Masih lebih besar dari toleransi 10^{-12} , jadi belum konvergen.

(6) Iterasi 3: Perbarui Nilai λ

Hitung C

$$\begin{aligned}
C &= \frac{f}{16} \times \cos^2 \alpha \times (4 + f \times (4 - 3 \times \cos^2 \alpha)) \\
&= \frac{0,0033528107}{16} \times 0,133055114 \times (4 + 0,0033528107 \times (4 \\
&\quad - 3 \times 0,133055114)) \\
&= 0,00020955 \\
\lambda_{baru} &= L + (1 - C) \times f \times \sin \alpha \times [\sigma + C \times \sin \sigma \times (\cos(2\sigma_m) \\
&\quad + C \times \cos \sigma \times (-1 + 2 \times \cos^2(2\sigma_m)))] \\
&= 0,0003940561 + (1 - 0,00020955) \times
\end{aligned}$$

$$\begin{aligned}
& 0,0033528107 \times 0,930561597 \times [0,0004125865 + \\
& 0,000112704 \times 0,0004125864 \times (0,66452687 + \\
& 0,00020955 \times 0,999999915 \times (-1 + 2 \times \\
& (0,66452687)^2))] \\
& = 0,0003940561 + (1 - 0,00020955) \times \\
& 0,0033528107 \times 0,930561597 \times 0,0004126439 \\
& = 0,0003940561 + 0,0000012867 \\
& = 0,0003940574 \text{ rad}
\end{aligned}$$

(7) Iterasi 4

Ulangi langkah sama dengan nilai $\lambda_{baru} = 0,0003940574$

$$\begin{aligned}
|\lambda_{baru} - \lambda_{lama}| &= |0,0003940574 - 0,0003940604| \\
&= 3 \times 10^{-9} \text{ rad}
\end{aligned}$$

Selisih nilai λ pada iterasi ke 4 sudah sangat kecil, yaitu 3×10^{-9} , yang berarti iterasi sudah hampir konvergen (toleransi biasanya 10^{-12} , tapi ini sudah sangat kecil untuk aplikasi praktis).

(8) Hitung jarak akhir s

$$\begin{aligned}
u^2 &= \cos^2 \alpha \times \frac{a^2 - b^2}{b^2} \\
&= 0,133055114 \times \frac{(6378137,0)^2 - (6356752,314245)^2}{(6356752,314245)^2} \\
&= 0,133055114 \times \frac{2,8 \times 10^{11}}{4,040 \times 10^{13}} \\
&= 0,133055114 \times 0,000694 \\
&= 0,000923
\end{aligned}$$

Hitung:

$$A = 1 + \frac{u^2}{16384} \times (4096 + u^2 \times (-768 + u^2 \times (320 - 175u^2)))$$

$$= 1,00023$$

$$B = \frac{u^2}{1024} \times (256 + u^2 \times (-128 + u^2 \times (74 - 47u^2)))$$

$$= 0,0009$$

Hitung $\Delta\sigma$:

$$\begin{aligned} \Delta\sigma = B \times \sin \sigma \times (\cos(2\sigma_m) + \frac{B}{4} \times (\cos \sigma \times (-1 + \\ 2 \cos^2(2\sigma_m)) - \frac{B}{6} \times \cos(2\sigma_m) \times (-3 + 4 \sin^2 \sigma) \times (-3 + \\ 4 \cos^2(2\sigma_m)))) \end{aligned}$$

$$s = b \times A \times (\sigma - \Delta\sigma)$$

$$= 6356752,314245 \times 1,00023 \times 0,0004125$$

$$= 2623,4 \text{ meter}$$

$$= 2,6234 \text{ km}$$

Tabel 4.10 adalah data jarak antar titik menggunakan fungsi heuristik *Vincenty* yang dihitung menggunakan *google collab*.

Tabel 4.10 Data Heuristik Menggunakan Formula *Vincenty* (RSU Balimed ke RSUD Wangaya)

Titik 1	Titik 2	Jarak (km)	Titik 1	Titik 2	Jarak (km)
A_1	T_1	2,63	J_7	T_1	0,61
J_1	T_1	2,56	J_8	T_1	0,53
J_2	T_1	2,00	J_9	T_1	0,33
J_3	T_1	1,98	J_{10}	T_1	0,82
J_4	T_1	1,12	J_{11}	T_1	0,62
J_5	T_1	1,58	J_{12}	T_1	0,46
J_6	T_1	1,01	J_{13}	T_1	0,40

Berdasarkan perhitungan jarak menggunakan rumus *Vincenty* yang ditampilkan pada Tabel 4.10, langkah selanjutnya adalah membuktikan bahwa nilai heuristik tersebut memenuhi sifat *consistent* dan *admissible*.

Pembuktian *Consistency* Heuristik *Vincenty*

Berikut akan dibuktikan bahwa nilai heuristik *vincenty* pada Tabel 4.10 terbukti *consistent*.

Tabel 4.11 Perhitungan Konsistensi Heuristik *Vincenty* (RSU Balimed ke RSUD Wangaya)

Pasangan Simpul ($n \rightarrow n'$)	$h(n)$ (km)	$c(n, n')$ (km)	$h(n')$ (km)	$c(n, n') + h(n')$ (km)	Konsisten? (Apakah $h(n) \leq c(n, n') + h(n')$?)
$A_1 \rightarrow J_1$	2,63	0,07	2,56	2,63	Ya, $2,63 \leq 2,63$
$J_1 \rightarrow J_2$	2,56	0,80	2,00	2,80	Ya, $2,56 \leq 2,80$
$J_2 \rightarrow J_3$	2,00	0,90	1,98	2,88	Ya, $2,00 \leq 2,88$
$J_2 \rightarrow J_4$	2,00	1,00	1,12	2,12	Ya, $2,00 \leq 2,12$
$J_3 \rightarrow J_5$	1,98	0,40	1,58	1,98	Ya, $1,98 \leq 1,98$
$J_4 \rightarrow J_6$	1,12	0,35	1,01	1,36	Ya, $1,12 \leq 1,36$
$J_4 \rightarrow J_{10}$	1,12	0,45	0,82	1,27	Ya, $1,12 \leq 1,27$
$J_5 \rightarrow J_6$	1,58	0,85	1,01	1,86	Ya, $1,58 \leq 1,86$
$J_6 \rightarrow J_7$	1,01	0,45	0,61	1,06	Ya, $1,01 \leq 1,06$
$J_7 \rightarrow J_8$	0,61	0,08	0,53	0,61	Ya, $0,61 \leq 0,61$
$J_8 \rightarrow J_9$	0,53	0,24	0,33	0,57	Ya, $0,53 \leq 0,57$
$J_9 \rightarrow J_{12}$	0,33	0,17	0,46	0,63	Ya, $0,33 \leq 0,63$
$J_{10} \rightarrow J_{11}$	0,82	0,28	0,62	0,90	Ya, $0,82 \leq 0,90$
$J_{11} \rightarrow J_8$	0,62	0,15	0,53	0,68	Ya, $0,62 \leq 0,68$
$J_{11} \rightarrow J_{12}$	0,62	0,23	0,46	0,69	Ya, $0,62 \leq 0,69$
$J_{12} \rightarrow J_{13}$	0,46	0,27	0,40	0,67	Ya, $0,46 \leq 0,67$
$J_{13} \rightarrow T_1$	0,40	0,40	0,00	0,40	Ya, $0,40 \leq 0,40$

Dari Tabel 4.11 semua nilai $h(n)$ memenuhi kondisi konsistensi $h(n) \leq c(n, n') + h(n')$. Dengan demikian, heuristik ini konsisten (*consistent*).

Pembuktian *Admissibility* Heuristik *Vincenty*

Langkah berikutnya adalah membuktikan bahwa heuristik tersebut juga *admissible*. Dari Tabel 4.12, terlihat bahwa untuk semua simpul, nilai heuristik *Vincenty* tidak pernah melebihi jarak aktual terpendek ke tujuan. Dengan demikian, heuristik ini memenuhi syarat *admissible*. Jadi, nilai heuristik *Vincenty* yang digunakan pada algoritma pencarian rute ini terbukti *consistent* (Tabel 4.11) dan *admissible* (Tabel 4.12).

Tabel 4.12 Perhitungan *Admissible* Heuristik *Vincenty* (RSU Balimed ke RSUD Wangaya)

Pasangan Simpul	Heuristik <i>Vincenty</i> $h(n)$ (km)	Jarak Aktual Terpendek $g^*(n)$ (km)	<i>Admissible?</i> (Apakah $h(n) \leq g^*(n)$?)
$A_1 \rightarrow T_1$	2,63	3,50	Ya, $2,63 \leq 3,50$
$J_1 \rightarrow T_1$	2,56	3,43	Ya, $2,56 \leq 3,43$
$J_2 \rightarrow T_1$	2,00	2,63	Ya, $2,00 \leq 2,63$
$J_3 \rightarrow T_1$	1,98	2,86	Ya, $1,98 \leq 2,86$
$J_4 \rightarrow T_1$	1,12	1,63	Ya, $1,12 \leq 1,63$
$J_5 \rightarrow T_1$	1,58	2,46	Ya, $1,58 \leq 2,46$
$J_6 \rightarrow T_1$	1,01	1,61	Ya, $1,01 \leq 1,61$
$J_7 \rightarrow T_1$	0,61	1,16	Ya, $0,61 \leq 1,16$
$J_8 \rightarrow T_1$	0,53	1,08	Ya, $0,53 \leq 1,08$
$J_9 \rightarrow T_1$	0,33	0,84	Ya, $0,33 \leq 0,84$
$J_{10} \rightarrow T_1$	0,82	1,18	Ya, $0,82 \leq 1,18$
$J_{11} \rightarrow T_1$	0,62	0,90	Ya, $0,62 \leq 0,90$
$J_{12} \rightarrow T_1$	0,46	0,67	Ya, $0,46 \leq 0,67$
$J_{13} \rightarrow T_1$	0,40	0,40	Ya, $0,40 \leq 0,40$

Kedua sifat ini menjamin bahwa algoritma *A-Star* yang menggunakan heuristik ini dapat menemukan jalur terpendek secara optimal dan efisien.

c. Implementasi Algoritma *A-Star*

Fungsi Heuristik *Haversine*

A_1 merupakan simpul awal untuk menuju RS Tujuan, sehingga A_1 dapat dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [A_1, 0]$$

Karena hanya J_1 yang sejalur dilewati, maka J_1 dimasukkan ke dalam *List tutup*.

$$List\ Tutup = [A_1, J_1, 0,07]$$

Karena hanya J_2 yang sejalur dilewati, maka J_2 dimasukkan ke dalam *List tutup*.

$$List\ Tutup = [A_1, J_1, J_2, 0,87]$$

Selanjutnya, akan ditentukan titik-titik mana saja yang memiliki jalur utama yang terhubung dengan J_2 , serta dilakukan perhitungan nilai heuristik untuk masing-masing titik tersebut. Setelah perhitungan, diperoleh hasil berupa panjang rute dan nilai heuristik sebagai berikut:

$$\text{Terpilih} = \{J_3: (0,90, 1,98), J_4: (1,00, 1,12)\}$$

Selanjutnya, kita melakukan seleksi melalui perhitungan untuk memilih nilai yang terkecil.

$$\begin{aligned} f(J_3) &= g(J_3) + h(J_3) \\ &= (0,07 + 0,80 + 0,90) + 1,98 \\ &= 1,77 + 1,98 \\ &= 3,75 \end{aligned}$$

$$\begin{aligned} f(J_4) &= g(J_4) + h(J_4) \\ &= (0,07 + 0,80 + 1,00) + 1,12 \\ &= 1,87 + 1,12 \\ &= 2,99 \end{aligned}$$

Hasil perhitungan yang didapat menunjukkan bahwa J_4 memiliki nilai terendah, sehingga dapat dimasukkan ke dalam *list* tutup sebagai berikut:

$$\text{List Tutup} = [A_1, J_1, J_2, J_4, 1,87]$$

Sama seperti dalam proses perhitungan sebelumnya, akan dicari titik-titik yang jalur utamanya terhubung dengan J_4 , dan hasilnya adalah sebagai berikut:

$$\text{Terpilih} = \{J_6: (0,33, 1,01), J_{10}: (0,47, 0,82)\}$$

$$f(J_6) = g(J_6) + h(J_6)$$

$$\begin{aligned}
&= (0,07 + 0,80 + 1,00 + 0,35) + 1,01 \\
&= 2,22 + 1,01 \\
&= 3,33
\end{aligned}$$

$$\begin{aligned}
f(J_{10}) &= g(J_{10}) + h(J_{10}) \\
&= (0,07 + 0,80 + 1,00 + 0,45) + 0,82 \\
&= 2,32 + 0,82 \\
&= 3,14
\end{aligned}$$

Dengan menggunakan proses yang sama. Diperoleh hasil bahwa titik dengan perhitungan terendah adalah J_{10} . Oleh karena itu, J_{10} dimasukkan ke dalam *List* tutup.

$$List \text{ Tutup} = [A_1, J_1, J_2, J_4, J_{10}], 2,32]$$

Karena hanya J_{11} yang sejalur dilewati, maka J_{11} dimasukkan ke dalam *List* tutup.

$$List \text{ Tutup} = [A_1, J_1, J_2, J_4, J_{10}, J_{11}], 2,60]$$

Sama seperti dalam proses perhitungan sebelumnya, akan dicari titik-titik yang jalur utamanya terhubung dengan J_{11} , dan hasilnya adalah sebagai berikut:

$$Terpilih = \{J_8: (0,15, 0,53), J_{12}: (0,23, 0,47)\}$$

$$\begin{aligned}
f(J_8) &= g(J_8) + h(J_8) \\
&= (0,07 + 0,80 + 1,00 + 0,45 + 0,28 + 0,15) + 0,53 \\
&= 2,75 + 0,53 \\
&= 3,28
\end{aligned}$$

$$\begin{aligned}
f(J_{12}) &= g(J_{12}) + h(J_{12}) \\
&= (0,07 + 0,80 + 1,00 + 0,45 + 0,28 + 0,23) + 0,47
\end{aligned}$$

$$= 2,83 + 0,47$$

$$= 3,30$$

Dengan menggunakan proses yang sama. Diperoleh hasil bahwa titik dengan perhitungan terendah adalah J_8 . Oleh karena itu, J_8 dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8], 2,75]$$

Karena hanya J_9 yang sejalur dilewati, maka J_9 dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8, J_9], 2,99]$$

Hanya J_{12} dan J_{13} yang sejalur ke jalur RS akhir, maka J_{12} dan J_{13} dimasukkan ke dalam *List* tutup sekaligus.

$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8, J_9, J_{12}, J_{13}], 3,43]$$

Sama halnya, karena hanya T_1 yang sejalur dilewati dan merupakan jalur RS tujuan yang menandakan telah sampainya dilokasi tujuan, maka T_1 dimasukkan ke dalam *List* tutup sekaligus didapatkan rute terpendeknya:

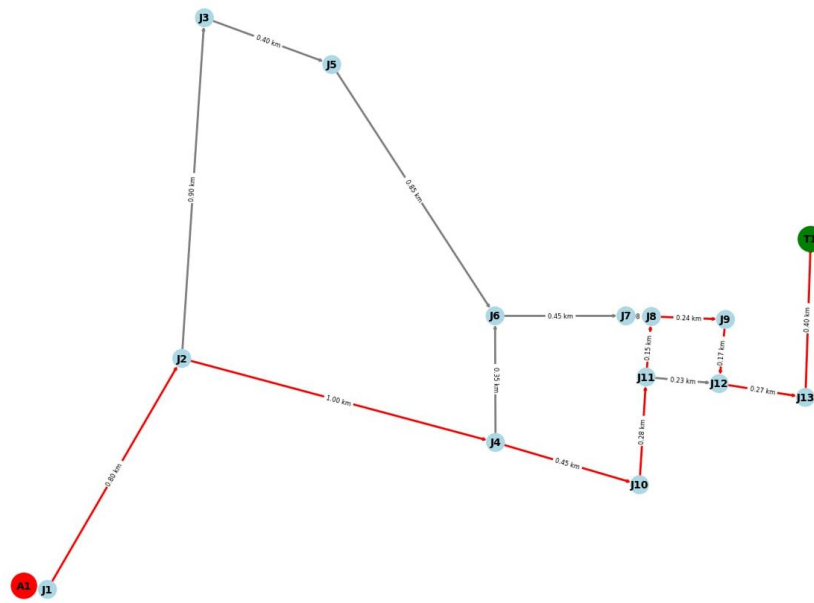
$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8, J_9, J_{12}, J_{13}, T_1], 3,83]$$

Berikut adalah hasil rute terpendek dari RSU Balimed ke RSUD Wangaya menggunakan fungsi heuristik *Haversine*, yaitu sebagai berikut:

$$A_1 \rightarrow J_1 \rightarrow J_2 \rightarrow J_4 \rightarrow J_{10} \rightarrow J_{11} \rightarrow J_8 \rightarrow J_9 \rightarrow J_{12} \rightarrow J_{13} \rightarrow T_1$$

Diperoleh dengan jarak dari A_1 menuju T_1 yaitu 3,83 km.

Berikut representasi graf berarah berbobot yang ditampilkan pada Gambar 4.4, menunjukkan hasil rute terpendek dari RSU Balimed ke RSUD Wangaya menggunakan fungsi heuristik *Haversine*.



Gambar 4.4 Graf Hasil Rute Terpendek dari RSU Balimed ke RSUD Wangaya (Heuristik *Haversine*)

Fungsi Heuristik *Vincenty*

A_1 merupakan simpul awal untuk menuju RS Tujuan, sehingga A_1 dapat dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_1], 0]$$

Karena hanya J_1 yang sejalur dilewati, maka J_1 dimasukkan ke dalam *List tutup*.

$$List\ Tutup = [[A_1, J_1], 0,07]$$

Karena hanya J_2 yang sejalur dilewati, maka J_2 dimasukkan ke dalam *List tutup*.

$$List\ Tutup = [[A_1, J_1, J_2], 0,87]$$

Selanjutnya, akan ditentukan titik-titik mana saja yang memiliki jalur utama yang terhubung dengan J_2 , serta dilakukan perhitungan nilai heuristik untuk masing-masing titik tersebut. Setelah perhitungan, diperoleh hasil berupa panjang rute dan nilai heuristik sebagai berikut:

$$\text{Terpilih} = \{J_3: (0,90, 1,98), J_4: (1,00, 1,12)\}$$

Selanjutnya, kita melakukan seleksi melalui perhitungan untuk memilih nilai yang terkecil.

$$\begin{aligned} f(J_3) &= g(J_3) + h(J_3) \\ &= (0,07 + 0,80 + 0,90) + 1,98 \\ &= 1,77 + 1,98 \\ &= 3,75 \end{aligned}$$

$$\begin{aligned} f(J_4) &= g(J_4) + h(J_4) \\ &= (0,07 + 0,80 + 1,00) + 1,12 \\ &= 1,87 + 1,12 \\ &= 2,99 \end{aligned}$$

Hasil perhitungan yang didapat menunjukkan bahwa J_4 memiliki nilai terendah, sehingga dapat dimasukkan ke dalam *list* tutup sebagai berikut:

$$\text{List Tutup} = [A_1, J_1, J_2, J_4, 1,87]$$

Sama seperti dalam proses perhitungan sebelumnya, akan dicari titik-titik yang jalur utamanya terhubung dengan J_4 , dan hasilnya adalah sebagai berikut:

$$\text{Terpilih} = \{J_6: (0,33, 1,01), J_{10}: (0,47, 0,82)\}$$

$$\begin{aligned} f(J_6) &= g(J_6) + h(J_6) \\ &= (0,07 + 0,80 + 1,00 + 0,35) + 1,01 \\ &= 2,22 + 1,01 \\ &= 3,33 \end{aligned}$$

$$f(J_{10}) = g(J_{10}) + h(J_{10})$$

$$\begin{aligned}
&= (0,07 + 0,80 + 1,00 + 0,45) + 0,82 \\
&= 2,32 + 0,82 \\
&= 3,14
\end{aligned}$$

Dengan menggunakan proses yang sama. Diperoleh hasil bahwa titik dengan perhitungan terendah adalah J_{10} . Oleh karena itu, J_{10} dimasukkan ke dalam *List* tutup.

$$List \text{ Tutup} = [A_1, J_1, J_2, J_4, J_{10}, 2,32]$$

Karena hanya J_{11} yang sejalur dilewati, maka J_{11} dimasukkan ke dalam *List* tutup.

$$List \text{ Tutup} = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, 2,60]$$

Sama seperti dalam proses perhitungan sebelumnya, akan dicari titik-titik yang jalur utamanya terhubung dengan J_{11} , dan hasilnya adalah sebagai berikut:

$$Terpilih = \{J_8: (0,15, 0,53), J_{12}: (0,23, 0,46)\}$$

$$\begin{aligned}
f(J_8) &= g(J_8) + h(J_8) \\
&= (0,07 + 0,80 + 1,00 + 0,45 + 0,28 + 0,15) + 0,53 \\
&= 2,75 + 0,53 \\
&= 3,28
\end{aligned}$$

$$\begin{aligned}
f(J_{12}) &= g(J_{12}) + h(J_{12}) \\
&= (0,07 + 0,80 + 1,00 + 0,45 + 0,28 + 0,23) + 0,46 \\
&= 2,83 + 0,46 \\
&= 3,29
\end{aligned}$$

Dengan menggunakan proses yang sama. Diperoleh hasil bahwa titik dengan perhitungan terendah adalah J_8 . Oleh karena itu, J_8 dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8], 2,75]$$

Karena hanya J_9 yang sejalur dilewati, maka J_9 dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8, J_9], 2,99]$$

Hanya J_{12} dan J_{13} yang sejalur ke jalur RS akhir, maka J_{12} dan J_{13} dimasukkan ke dalam *List* tutup sekaligus.

$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8, J_9, J_{12}, J_{13}], 3,43]$$

Sama halnya, karena hanya T_1 yang sejalur dilewati dan merupakan jalur RS tujuan yang menandakan telah sampainya dilokasi tujuan, maka T_1 dimasukkan ke dalam *List* tutup sekaligus didapatkan rute terpendeknya:

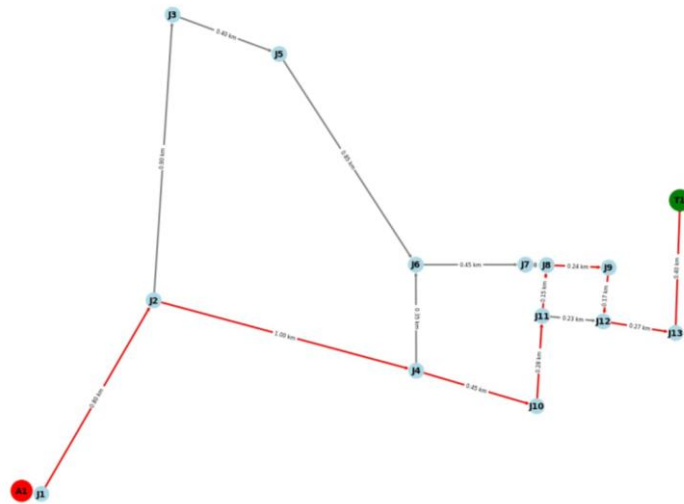
$$List\ Tutup = [A_1, J_1, J_2, J_4, J_{10}, J_{11}, J_8, J_9, J_{12}, J_{13}, T_1], 3,83]$$

Berikut adalah hasil rute terpendek dari RSU Balimed ke RSUD Wangaya menggunakan fungsi heuristik *Vincenty*, yaitu sebagai berikut:

$$A_1 \rightarrow J_1 \rightarrow J_2 \rightarrow J_4 \rightarrow J_{10} \rightarrow J_{11} \rightarrow J_8 \rightarrow J_9 \rightarrow J_{12} \rightarrow J_{13} \rightarrow T_1$$

Diperoleh dengan jarak dari A_1 menuju T_1 yaitu 3,83 km.

Berikut representasi graf berarah berbobot yang ditampilkan pada Gambar 4.5, menunjukkan hasil rute terpendek dari RSU Balimed ke RSUD Wangaya meggunakan fungsi heuristik *Vincenty*.

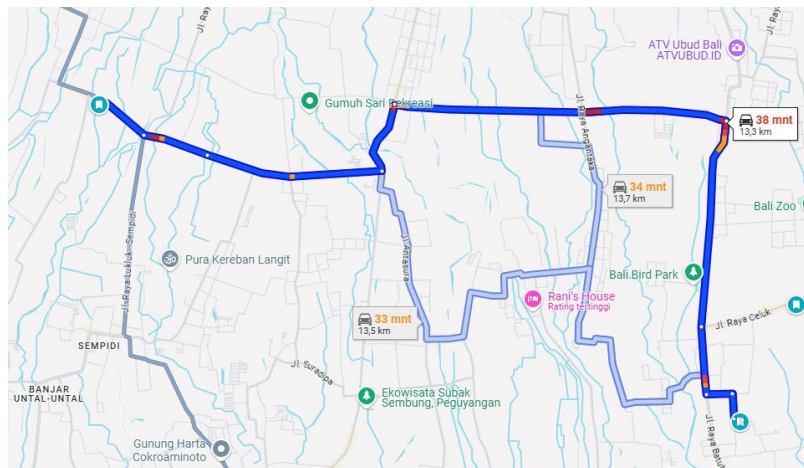


Gambar 4.5 Graf Hasil Rute Terpendek dari RSU Balimed ke RSUD Wangaya (Heuristik *Vincenty*)

2. Penentuan Rute $A_2 \rightarrow T_2$

a. Pengumpulan Data Titik dan Titik Persimpangan

Pada tahap awal, dilakukan visualisasi peta yang menampilkan lokasi rumah sakit awal (RSU Pregamagana), rumah sakit tujuan (RSD Mangusada) serta titik-titik persimpangan jalan yang menjadi jalur penghubung antar kedua rumah sakit tersebut. Gambar 4.6, yang menunjukkan peta rute yang menghubungkan RSU Premagana ke RSD Mangusada beserta titik persimpangan jalan utama yang dilalui.



Gambar 4.6 Peta Rute dari RSU Premagana ke RSD Mangusada

Selanjutnya, koordinat *latitude* dan *longitude* dari rumah sakit awal, rumah sakit tujuan, serta titik-titik persimpangan jalan tersebut diidentifikasi dan disajikan dalam tabel berikut sebagai dasar pemodelan graf dan perhitungan fungsi heuristik. Tabel 4.13 menyajikan pasangan rumah sakit yang digunakan, yaitu RSU Premagana sebagai simpul awal dan RSD Mangusada sebagai simpul tujuan.

Tabel 4.13 RS Awal (RSU Premagana) ke RS Tujuan (RSD Mangusada)

No	Nama Lokasi	Simbol	Latitude	Longitude
1	RSU Premagana	A_2	-8,614971	115,257351
2	RSD Mangusada	T_2	-8,578390	115,182946

Selanjutnya pada Tabel 4.14, terdapat titik-titik koordinat persimpangan jalan beserta simbolnya.

Tabel 4.14 Titik-titik Persimpangan Jalan (RSU Premagana ke RSD Mangusada)

Simbol	Latitude	Longitude	Simbol	Latitude	Longitude
J_{14}	-8,612028	115,256445	J_{27}	-8,597862	115,239032
J_{15}	-8,612087	115,253438	J_{28}	-8,598339	115,233520
J_{16}	-8,609951	115,253211	J_{29}	-8,598185	115,231838
J_{17}	-8,580757	115,255757	J_{30}	-8,598524	115,229876
J_{18}	-8,579747	115,234218	J_{31}	-8,600865	115,230031
J_{19}	-8,611198	115,250540	J_{32}	-8,605431	115,225293
J_{20}	-8,611569	115,249189	J_{33}	-8,605820	115,220960
J_{21}	-8,612891	115,249219	J_{34}	-8,588622	115,215536
J_{22}	-8,613277	115,243947	J_{35}	-8,578828	115,217234
J_{23}	-8,605956	115,238509	J_{36}	-8,586480	115,215763
J_{24}	-8,597654	115,239891	J_{37}	-8,584637	115,195506
J_{25}	-8,590968	115,240876	J_{38}	-8,582390	115,188158
J_{26}	-8,583681	115,239761			

Daftar nama jalan yang menghubungkan simpul-simpul tersebut disajikan dalam Tabel 4.15, dan digunakan sebagai jalur dalam graf berarah berbobot.

Tabel 4.15 Nama Jalan Antar Persimpangan (RSU Premagana ke RSD Mangusada)

Dari	Ke	Nama Jalan	Dari	Ke	Nama Jalan
A_2	J_{14}	Jl. Baka	J_{25}	J_{26}	Jl. Raya Angantaka
J_{14}	J_{15}	Jl. Smki	J_{26}	J_{18}	Jl. Raya Angantaka
J_{15}	J_{16}	Jl. Raya Batubulan	J_{27}	J_{28}	Jl. Trenggana
J_{16}	J_{17}	Jl. Raya Singapadu	J_{28}	J_{29}	Jl. Sekargadung
J_{16}	J_{19}	Jl. Kapten Japa	J_{29}	J_{30}	Jl. Sekargadung
J_{17}	J_{18}	Jl. Raya Angantaka – Sibang Gede	J_{30}	J_{31}	Jl. Cekomar
J_{18}	J_{35}	Jl. Raya Angantaka – Sibang Gede	J_{31}	J_{32}	Jl. Cekomaria
J_{19}	J_{20}	Jl. Pagutan	J_{32}	J_{33}	Jl. Raya Pangus
J_{20}	J_{21}	Jl. Pagutan	J_{33}	J_{34}	Jl. Antasura
J_{21}	J_{22}	Jl. Ida Bagus Japa	J_{34}	J_{36}	Jl. Raya Darmasaba
J_{22}	J_{23}	Jl. Nagasari	J_{35}	J_{36}	Jl. Raya Sibanggede
J_{23}	J_{24}	Jl. Nagasari	J_{36}	J_{37}	Jl. Raya Darmasaba
J_{24}	J_{25}	Jl. Raya Jagapati	J_{37}	J_{38}	Jl. Raya Perang
J_{24}	J_{27}	Jl. Raya Jagapati	J_{38}	T_2	Jl. Raya Kapal

Tabel 4.16 adalah data jarak aktual ($g(n)$) dari RSU Premagana ke RSD

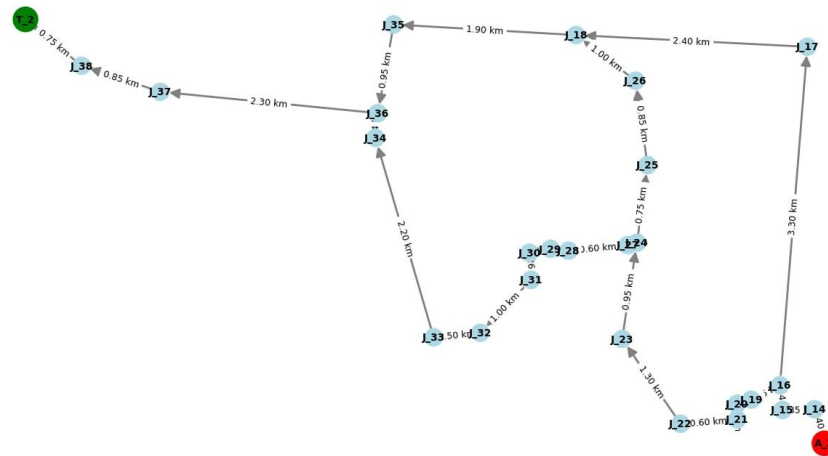
Mangusada menggunakan *Google Maps*.

Tabel 4.16 Data Jarak Aktual ($g(n)$) (RSU Premagana ke RSD Mangusada)

Titik 1	Titik 2	Jarak (km)	Titik 1	Titik 2	Jarak (km)
A_2	J_{14}	0,40	J_{25}	J_{26}	0,85
J_{14}	J_{15}	0,35	J_{26}	J_{18}	1,00
J_{15}	J_{16}	0,24	J_{27}	J_{28}	0,60
J_{16}	J_{17}	3,30	J_{28}	J_{29}	0,21
J_{16}	J_{19}	0,35	J_{29}	J_{30}	0,21
J_{17}	J_{18}	2,40	J_{30}	J_{31}	0,26
J_{18}	J_{35}	1,90	J_{31}	J_{32}	1,00
J_{19}	J_{20}	0,16	J_{32}	J_{33}	0,50
J_{20}	J_{21}	0,15	J_{33}	J_{34}	2,20
J_{21}	J_{22}	0,60	J_{34}	J_{36}	0,24
J_{22}	J_{23}	1,30	J_{35}	J_{36}	0,95
J_{23}	J_{24}	0,95	J_{36}	J_{37}	2,30
J_{24}	J_{25}	0,75	J_{37}	J_{38}	0,85
J_{24}	J_{27}	0,10	J_{38}	T_2	0,75

b. Perhitungan Fungsi Heuristik

Berikut Gambar 4.7, merupakan representasi dalam bentuk graf berbobot berarah yang dibentuk berdasarkan peta rute dari RSU Premagana menuju RSD Mangusada.



Gambar 4.7 Graf Berarah Berbobot (RSU Premagana ke RSD Mangusada)

Perhitungan Heuristik *Haversine*

Berikut Tabel 4.17, yaitu data jarak antar simpul menggunakan fungsi heuristik *Haversine* yang dihitung menggunakan *google collab*.

Tabel 4.17 Data Heuristik Menggunakan Formula Haversine (RSU Premagana ke RSD Mangusada)

Titik 1	Titik 2	Jarak (km)	Titik 1	Titik 2	Jarak (km)
A_2	T_2	9,13	J_{26}	T_2	6,27
J_{14}	T_2	8,90	J_{27}	T_2	6,53
J_{15}	T_2	8,60	J_{28}	T_2	5,98
J_{16}	T_2	8,48	J_{29}	T_2	5,80
J_{17}	T_2	8,01	J_{30}	T_2	5,62
J_{18}	T_2	5,63	J_{31}	T_2	5,74
J_{19}	T_2	8,27	J_{32}	T_2	5,54
J_{20}	T_2	8,16	J_{33}	T_2	5,17
J_{21}	T_2	8,23	J_{34}	T_2	3,76
J_{22}	T_2	7,74	J_{35}	T_2	3,77
J_{23}	T_2	6,83	J_{36}	T_2	3,71
J_{24}	T_2	6,61	J_{37}	T_2	1,54
J_{25}	T_2	6,52	J_{38}	T_2	0,73

Berdasarkan perhitungan jarak menggunakan rumus *Haversine* yang ditampilkan pada Tabel 4.17, langkah selanjutnya adalah membuktikan bahwa nilai heuristik tersebut memenuhi sifat *consistent* dan *admissible*.

Pembuktian *Consistency* Heuristik *Haversine*

Berikut akan dibuktikan bahwa nilai heuristik *Haversine* diatas terbukti *consistent*.

Tabel 4.18 Perhitungan Konsistensi Heuristik *Haversine* (RSU Premagana ke RSD Mangusada)

Pasangan Simpul ($n \rightarrow n'$)	$h(n)$ (km)	$c(n, n')$ (km)	$h(n')$ (km)	$c(n, n') + h(n')$ (km)	Konsisten? (Apakah $h(n) \leq c(n, n') + h(n')$?)
$A_2 \rightarrow J_{14}$	9,13	0,40	8,90	9,30	Ya, $9,13 \leq 9,30$
$J_{14} \rightarrow J_{15}$	8,90	0,35	8,60	8,95	Ya, $8,90 \leq 8,95$
$J_{15} \rightarrow J_{16}$	8,60	0,24	8,48	8,72	Ya, $8,60 \leq 8,72$
$J_{16} \rightarrow J_{17}$	8,48	3,30	8,01	11,31	Ya, $8,48 \leq 11,31$
$J_{16} \rightarrow J_{19}$	8,48	0,35	8,27	8,63	Ya, $8,48 \leq 8,63$
$J_{17} \rightarrow J_{18}$	8,01	2,40	5,64	8,04	Ya, $8,01 \leq 8,04$
$J_{18} \rightarrow J_{35}$	5,64	1,90	3,77	5,67	Ya, $5,64 \leq 5,67$
$J_{19} \rightarrow J_{20}$	8,27	0,16	8,16	8,32	Ya, $8,27 \leq 8,32$
$J_{20} \rightarrow J_{21}$	8,16	0,15	8,23	8,38	Ya, $8,16 \leq 8,38$
$J_{21} \rightarrow J_{22}$	8,23	0,60	7,74	8,34	Ya, $8,23 \leq 8,34$
$J_{22} \rightarrow J_{23}$	7,74	1,30	6,83	8,13	Ya, $7,74 \leq 8,13$
$J_{23} \rightarrow J_{24}$	6,83	0,95	6,61	7,56	Ya, $6,83 \leq 0,63$
$J_{24} \rightarrow J_{25}$	6,61	0,75	6,52	7,27	Ya, $6,61 \leq 7,27$
$J_{24} \rightarrow J_{27}$	6,61	0,10	6,53	6,63	Ya, $6,61 \leq 6,63$
$J_{25} \rightarrow J_{26}$	6,52	0,85	6,27	7,12	Ya, $6,52 \leq 7,12$
$J_{26} \rightarrow J_{18}$	6,27	1,00	5,64	6,64	Ya, $6,27 \leq 6,64$
$J_{27} \rightarrow J_{28}$	6,53	0,60	5,98	6,58	Ya, $6,53 \leq 6,58$
$J_{28} \rightarrow J_{29}$	5,98	0,21	5,80	6,01	Ya, $5,98 \leq 6,01$
$J_{29} \rightarrow J_{30}$	5,80	0,21	5,62	5,83	Ya, $5,80 \leq 5,83$
$J_{30} \rightarrow J_{31}$	5,62	0,26	5,74	6,00	Ya, $5,62 \leq 6,00$
$J_{31} \rightarrow J_{32}$	5,74	1,00	5,54	6,54	Ya, $5,74 \leq 6,54$
$J_{32} \rightarrow J_{33}$	5,54	0,50	5,17	5,67	Ya, $5,54 \leq 5,67$
$J_{33} \rightarrow J_{34}$	5,17	2,20	3,76	5,96	Ya, $5,17 \leq 5,96$
$J_{34} \rightarrow J_{36}$	3,76	0,24	3,71	3,95	Ya, $3,76 \leq 3,95$
$J_{35} \rightarrow J_{36}$	3,77	0,95	3,71	4,66	Ya, $3,77 \leq 4,66$
$J_{36} \rightarrow J_{37}$	3,71	2,30	1,54	3,84	Ya, $3,71 \leq 3,84$
$J_{37} \rightarrow J_{38}$	1,54	0,85	0,73	1,58	Ya, $1,54 \leq 1,58$
$J_{38} \rightarrow T_2$	0,73	0,75	0,00	0,75	Ya, $0,73 \leq 0,75$

Dari Tabel 4.18 semua nilai $h(n)$ memenuhi kondisi konsistensi $h(n) \leq c(n, n') + h(n')$. Dengan demikian, heuristik ini konsisten (*consistent*).

Pembuktian *Admissibility* Heuristik *Haversine*

Langkah berikutnya adalah membuktikan bahwa heuristik tersebut juga *admissible*.

Tabel 4.19 Perhitungan *Admissible* Heuristik *Haversine* (RSU Premagana ke RSD Mangusada)

Pasangan Simpul	Heuristik <i>Haversine</i> $h(n)$ (km)	Jarak Aktual Terpendek $g^*(n)$ (km)	<i>Admissible?</i> (Apakah $h(n) \leq g^*(n)$?)
$A_2 \rightarrow T_2$	9,13	13,44	Ya, $9,13 \leq 13,44$
$J_{14} \rightarrow T_2$	8,90	13,04	Ya, $8,90 \leq 13,04$
$J_{15} \rightarrow T_2$	8,60	12,69	Ya, $8,60 \leq 12,69$
$J_{16} \rightarrow T_2$	8,48	12,45	Ya, $8,48 \leq 12,45$
$J_{17} \rightarrow T_2$	8,01	9,15	Ya, $8,01 \leq 9,15$
$J_{18} \rightarrow T_2$	5,63	6,75	Ya, $5,63 \leq 6,75$
$J_{19} \rightarrow T_2$	8,27	12,38	Ya, $8,27 \leq 12,38$
$J_{20} \rightarrow T_2$	8,16	12,22	Ya, $8,16 \leq 12,22$
$J_{21} \rightarrow T_2$	8,23	12,07	Ya, $8,23 \leq 12,07$
$J_{22} \rightarrow T_2$	7,74	11,47	Ya, $7,74 \leq 11,47$
$J_{23} \rightarrow T_2$	6,83	10,17	Ya, $6,83 \leq 10,17$
$J_{24} \rightarrow T_2$	6,61	9,22	Ya, $6,61 \leq 9,22$
$J_{25} \rightarrow T_2$	6,52	8,60	Ya, $6,52 \leq 8,60$
$J_{26} \rightarrow T_2$	6,27	7,75	Ya, $6,27 \leq 7,75$
$J_{27} \rightarrow T_2$	6,53	9,12	Ya, $6,53 \leq 9,12$
$J_{28} \rightarrow T_2$	5,98	8,52	Ya, $5,98 \leq 8,52$
$J_{29} \rightarrow T_2$	5,80	8,31	Ya, $5,80 \leq 8,31$
$J_{30} \rightarrow T_2$	5,62	8,10	Ya, $5,62 \leq 8,10$
$J_{31} \rightarrow T_2$	5,74	7,84	Ya, $5,74 \leq 7,84$
$J_{32} \rightarrow T_2$	5,54	6,84	Ya, $5,54 \leq 6,84$
$J_{33} \rightarrow T_2$	5,17	6,34	Ya, $5,17 \leq 6,34$
$J_{34} \rightarrow T_2$	3,76	4,14	Ya, $3,76 \leq 4,14$
$J_{35} \rightarrow T_2$	3,77	4,85	Ya, $3,77 \leq 4,85$
$J_{36} \rightarrow T_2$	3,71	3,90	Ya, $3,71 \leq 3,90$
$J_{37} \rightarrow T_2$	1,54	1,60	Ya, $1,54 \leq 1,60$
$J_{38} \rightarrow T_2$	0,73	0,75	Ya, $0,73 \leq 0,75$

Dari Tabel 4.19 terlihat bahwa untuk semua simpul, nilai heuristik *Haversine* tidak pernah melebihi jarak aktual terpendek ke tujuan. Dengan demikian, heuristik ini memenuhi syarat *admissible*.

Jadi, nilai heuristik *Haversine* yang digunakan pada algoritma pencarian rute ini terbukti *consistent* (Tabel 4.18) dan *admissible* (Tabel 4.19). Kedua sifat ini menjamin bahwa algoritma A-*Star* yang menggunakan heuristik ini dapat menemukan jalur terpendek secara optimal dan efisien.

Perhitungan Heuristik *Vincenty*

Berikut adalah data jarak antar titik menggunakan fungsi heuristik *Vincenty* yang dihitung menggunakan *google collab*. Mengingat banyak titik yang perlu dihitung dalam pencarian rute, perhitungan jarak antara setiap titik tersebut akan lebih efisien dengan bantuan *Google Colab*.

Tabel 4.20 Data Heuristik Menggunakan Formula *Vincenty* (RSU Premagana ke RSD Mangusada)

Titik 1	Titik 2	Jarak (KM)	Titik 1	Titik 2	Jarak (KM)
A_2	T_2	9,13	J_{26}	T_2	6,28
J_{14}	T_2	8,90	J_{27}	T_2	6,53
J_{15}	T_2	8,60	J_{28}	T_2	5,98
J_{16}	T_2	8,48	J_{29}	T_2	5,81
J_{17}	T_2	8,01	J_{30}	T_2	5,62
J_{18}	T_2	5,64	J_{31}	T_2	5,74
J_{19}	T_2	8,27	J_{32}	T_2	5,53
J_{20}	T_2	8,16	J_{33}	T_2	5,16
J_{21}	T_2	8,23	J_{34}	T_2	3,76
J_{22}	T_2	7,74	J_{35}	T_2	3,77
J_{23}	T_2	6,83	J_{36}	T_2	3,72
J_{24}	T_2	6,62	J_{37}	T_2	1,54
J_{25}	T_2	6,52	J_{38}	T_2	0,72

Berdasarkan perhitungan jarak menggunakan rumus *Vincenty* yang ditampilkan pada Tabel 4.20, langkah selanjutnya adalah membuktikan bahwa nilai heuristik tersebut memenuhi sifat *consistent* dan *admissible*.

Pembuktian *Consistency* Heuristik *Vincenty*

Berikut akan dibuktikan bahwa nilai heuristik *Vincenty* terbukti *consistent*.

Tabel 4.21 Perhitungan Konsistensi Heuristik *Vincenty* (RSU Premagana ke RSD Mangusada)

Pasangan Simpul ($n \rightarrow n'$)	$h(n)$ (km)	$c(n, n')$ (km)	$h(n')$ (km)	$c(n, n') + h(n')$ (km)	Konsisten? (Apakah $h(n) \leq c(n, n') + h(n')$?)
$A_2 \rightarrow J_{14}$	9,13	0,40	8,90	9,30	Ya, $9,13 \leq 9,30$
$J_{14} \rightarrow J_{15}$	8,90	0,35	8,60	8,95	Ya, $8,90 \leq 8,95$
$J_{15} \rightarrow J_{16}$	8,60	0,24	8,48	8,72	Ya, $8,60 \leq 8,72$
$J_{16} \rightarrow J_{17}$	8,48	3,30	8,01	11,31	Ya, $8,48 \leq 11,31$
$J_{16} \rightarrow J_{19}$	8,48	0,35	8,27	8,62	Ya, $8,48 \leq 8,62$
$J_{17} \rightarrow J_{18}$	8,01	2,40	5,64	8,04	Ya, $8,01 \leq 8,04$
$J_{18} \rightarrow J_{35}$	5,64	1,90	3,77	5,67	Ya, $5,64 \leq 5,67$
$J_{19} \rightarrow J_{20}$	8,27	0,16	8,16	8,32	Ya, $8,27 \leq 8,32$
$J_{20} \rightarrow J_{21}$	8,16	0,15	8,23	8,38	Ya, $8,16 \leq 8,38$
$J_{21} \rightarrow J_{22}$	8,23	0,60	7,74	8,34	Ya, $8,23 \leq 8,34$
$J_{22} \rightarrow J_{23}$	7,74	1,30	6,83	8,13	Ya, $7,74 \leq 8,13$
$J_{23} \rightarrow J_{24}$	6,83	0,95	6,62	7,56	Ya, $6,83 \leq 7,56$
$J_{24} \rightarrow J_{25}$	6,62	0,75	6,52	7,25	Ya, $6,62 \leq 7,25$
$J_{24} \rightarrow J_{27}$	6,62	0,10	6,53	6,63	Ya, $6,62 \leq 6,63$
$J_{25} \rightarrow J_{26}$	6,52	0,85	6,28	7,13	Ya, $6,52 \leq 7,13$
$J_{26} \rightarrow J_{18}$	6,28	1,00	5,64	6,64	Ya, $6,28 \leq 6,64$
$J_{27} \rightarrow J_{28}$	6,53	0,60	5,98	6,58	Ya, $6,53 \leq 6,58$
$J_{28} \rightarrow J_{29}$	5,98	0,21	5,81	6,02	Ya, $5,98 \leq 6,02$
$J_{29} \rightarrow J_{30}$	5,81	0,21	5,62	5,83	Ya, $5,81 \leq 5,83$
$J_{30} \rightarrow J_{31}$	5,62	0,26	5,74	6,00	Ya, $5,62 \leq 6,00$
$J_{31} \rightarrow J_{32}$	5,74	1,00	5,53	6,53	Ya, $5,74 \leq 6,53$
$J_{32} \rightarrow J_{33}$	5,53	0,50	5,16	5,66	Ya, $5,53 \leq 5,66$
$J_{33} \rightarrow J_{34}$	5,16	2,20	3,76	5,96	Ya, $5,16 \leq 5,96$
$J_{34} \rightarrow J_{36}$	3,76	0,24	3,72	3,96	Ya, $3,76 \leq 3,96$
$J_{35} \rightarrow J_{36}$	3,77	0,95	3,72	4,67	Ya, $3,77 \leq 4,67$
$J_{36} \rightarrow J_{37}$	3,72	2,30	1,54	3,84	Ya, $3,72 \leq 3,84$
$J_{37} \rightarrow J_{38}$	1,54	0,85	0,72	1,57	Ya, $1,54 \leq 1,57$
$J_{38} \rightarrow T_2$	0,72	0,75	0,00	0,75	Ya, $0,72 \leq 0,75$

Dari Tabel 4.21, semua nilai $h(n)$ memenuhi kondisi kondisi konsistensi $h(n) \leq c(n, n') + h(n')$. Dengan demikian, heuristik ini konsisten (*consistent*).

Pembuktian *Admissibility* Heuristik *Vincenty*

Langkah berikutnya adalah membuktikan bahwa heuristik tersebut juga *admissible*.

Tabel 4.22 Perhitungan *Admissible* Heuristik *Vincenty* (RSU Premagana ke RSD Mangusada)

Pasangan Simpul	Heuristik <i>Vincenty</i> $h(n)$ (km)	Jarak Aktual Terpendek $g^*(n)$ (km)	<i>Admissible?</i> (Apakah $h(n) \leq g^*(n)$?)
$A_2 \rightarrow T_2$	9,13	13,44	Ya, $9,13 \leq 13,44$
$J_{14} \rightarrow T_2$	8,90	13,04	Ya, $8,90 \leq 13,04$
$J_{15} \rightarrow T_2$	8,60	12,69	Ya, $8,60 \leq 12,69$
$J_{16} \rightarrow T_2$	8,48	12,45	Ya, $8,48 \leq 12,45$
$J_{17} \rightarrow T_2$	8,01	9,15	Ya, $8,01 \leq 9,15$
$J_{18} \rightarrow T_2$	5,64	6,75	Ya, $5,64 \leq 6,75$
$J_{19} \rightarrow T_2$	8,27	12,38	Ya, $8,27 \leq 12,38$
$J_{20} \rightarrow T_2$	8,16	12,22	Ya, $8,16 \leq 12,22$
$J_{21} \rightarrow T_2$	8,23	12,07	Ya, $8,23 \leq 12,07$
$J_{22} \rightarrow T_2$	7,74	11,47	Ya, $7,74 \leq 11,47$
$J_{23} \rightarrow T_2$	6,83	10,17	Ya, $6,83 \leq 10,17$
$J_{24} \rightarrow T_2$	6,62	9,22	Ya, $6,61 \leq 9,22$
$J_{25} \rightarrow T_2$	6,52	8,60	Ya, $6,52 \leq 8,60$
$J_{26} \rightarrow T_2$	6,28	7,75	Ya, $6,28 \leq 7,75$
$J_{27} \rightarrow T_2$	6,53	9,12	Ya, $6,53 \leq 9,12$
$J_{28} \rightarrow T_2$	5,98	8,52	Ya, $5,98 \leq 8,52$
$J_{29} \rightarrow T_2$	5,81	8,31	Ya, $5,81 \leq 8,31$
$J_{30} \rightarrow T_2$	5,62	8,10	Ya, $5,62 \leq 8,10$
$J_{31} \rightarrow T_2$	5,74	7,84	Ya, $5,74 \leq 7,84$
$J_{32} \rightarrow T_2$	5,53	6,84	Ya, $5,53 \leq 6,84$
$J_{33} \rightarrow T_2$	5,16	6,34	Ya, $5,16 \leq 6,34$
$J_{34} \rightarrow T_2$	3,76	4,14	Ya, $3,76 \leq 4,14$
$J_{35} \rightarrow T_2$	3,77	4,85	Ya, $3,77 \leq 4,85$
$J_{36} \rightarrow T_2$	3,72	3,90	Ya, $3,72 \leq 3,90$
$J_{37} \rightarrow T_2$	1,54	1,60	Ya, $1,54 \leq 1,60$
$J_{38} \rightarrow T_2$	0,72	0,75	Ya, $0,72 \leq 0,75$

Dari Tabel 4.22 terlihat bahwa untuk semua simpul, nilai heuristik *Vincenty* tidak pernah melebihi jarak aktual terpendek ke tujuan. Dengan demikian, heuristik ini memenuhi syarat *admissible*.

Jadi, nilai heuristik *Haversine* yang digunakan pada algoritma pencarian rute ini terbukti *consistent* (Tabel 4.21) dan *admissible* (Tabel 4.22). Kedua sifat ini menjamin bahwa algoritma A-Star yang menggunakan heuristik ini dapat menemukan jalur terpendek secara optimal dan efisien.

c. Implementasi Algoritma A-Star

Fungsi Heuristik *Haversine*

A_2 merupakan simpul awal untuk menuju RS Tujuan, sehingga A_2 dapat dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2], 0]$$

Karena hanya J_{14} yang sejalur dilewati, maka J_{14} dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2, J_{14}], 0,40]$$

Karena hanya J_{15} yang sejalur dilewati, maka J_{15} dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2, J_{14}, J_{15}], 0,75]$$

Karena hanya J_{16} yang sejalur dilewati, maka J_{16} dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}], 0,99]$$

Selanjutnya, akan ditentukan titik-titik mana saja yang memiliki jalur utama yang terhubung dengan J_{16} , serta dilakukan perhitungan nilai heuristik untuk masing-masing titik tersebut. Setelah perhitungan, diperoleh hasil berupa panjang rute dan nilai heuristik sebagai berikut:

$$Terpilih = \{J_{17}: (3,30, 8,01), J_{19}: (0,35, 8,27)\}$$

Selanjutnya, kita melakukan seleksi melalui perhitungan untuk memilih nilai yang terkecil.

$$\begin{aligned}
 f(J_{17}) &= g(J_{17}) + h(J_{17}) \\
 &= (0,40 + 0,35 + 0,24 + 3,30) + 8,01 \\
 &= 4,29 + 8,01 \\
 &= 12,18
 \end{aligned}$$

$$\begin{aligned}
 f(J_{19}) &= g(J_{19}) + h(J_{19}) \\
 &= (0,40 + 0,35 + 0,24 + 0,35) + 8,27 \\
 &= 1,34 + 8,27 \\
 &= 9,61
 \end{aligned}$$

Hasil perhitungan yang didapat menunjukkan bahwa J_{19} memiliki nilai terendah, sehingga dapat dimasukkan ke dalam *list* tutup sebagai berikut:

$$List\ Tutup = [A_2, J_{14}, J_{15}, J_{16}, J_{19}, 1,34]$$

Karena hanya J_{20} yang sejalur dilewati, maka J_{20} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, 1,50]$$

J_{20} hanya terhubung dengan J_{21} , sedangkan J_{21} terhubung langsung dengan J_{22} , J_{22} terhubung langsung dengan J_{23} , dan J_{23} terhubung langsung dengan J_{24} . Sehingga $J_{21}, J_{22}, J_{23}, J_{24}$ langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$List\ Tutup = [A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, 4,50]$$

Sama seperti dalam proses perhitungan sebelumnya, akan dicari titik-titik yang jalur utamanya terhubung dengan J_{24} , dan hasilnya adalah sebagai berikut.

$$\text{Terpilih} = \{J_{25}: (0,75, 6,52), J_{27}: (0,10, 6,53)\}$$

$$\begin{aligned} f(J_{25}) &= g(J_{25}) + h(J_{25}) \\ &= (0,40 + 0,35 + 0,24 + 0,35 + 0,16 + 0,15 + 0,60 + 1,30 + \\ &\quad 0,95 + 0,75) + 6,52 \\ &= 5,25 + 6,52 \\ &= 11,77 \end{aligned}$$

$$\begin{aligned} f(J_{27}) &= g(J_{27}) + h(J_{27}) \\ &= (0,40 + 0,35 + 0,24 + 0,35 + 0,16 + 0,15 + 0,60 + 1,30 + \\ &\quad 0,95 + 0,10) + 6,53 \\ &= 4,60 + 6,53 \\ &= 11,13 \end{aligned}$$

Dengan menggunakan proses yang sama. Diperoleh hasil bahwa titik dengan perhitungan terendah adalah J_{27} . Oleh karena itu, J_{27} dimasukkan ke dalam *List* tutup.

$$\text{List Tutup} = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}], 4,60]$$

J_{27} hanya terhubung dengan J_{28} , sedangkan J_{28} terhubung langsung dengan J_{29} , dan J_{29} terhubung langsung dengan J_{30} . Sehingga J_{28}, J_{29}, J_{30} langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$\begin{aligned} \text{List Tutup} &= [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, \\ &\quad J_{30}], 5,62] \end{aligned}$$

J_{30} hanya terhubung dengan J_{31} , sedangkan J_{31} terhubung langsung dengan J_{32} , J_{32} terhubung langsung dengan J_{33} , J_{33} terhubung langsung dengan J_{34} dan J_{34} terhubung langsung dengan J_{36} . Sehingga $J_{31}, J_{32}, J_{33}, J_{34}, J_{36}$ langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, J_{32}, J_{33}, J_{34}, J_{36}], 9, 82]$$

Karena hanya J_{37} yang sejalur dilewati, maka J_{37} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, J_{32}, J_{33}, J_{34}, J_{36}, J_{37}], 12, 12]$$

Karena hanya J_{38} yang sejalur dilewati, maka J_{38} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, J_{32}, J_{33}, J_{34}, J_{36}, J_{37}, J_{38}], 12, 97]$$

Sama halnya, karena hanya T_2 yang sejalur dilewati dan merupakan jalur RS tujuan yang menandakan telah sampainya dilokasi tujuan, maka T_2 dimasukkan ke dalam *List* tutup sekaligus didapatkan rute terpendeknya:

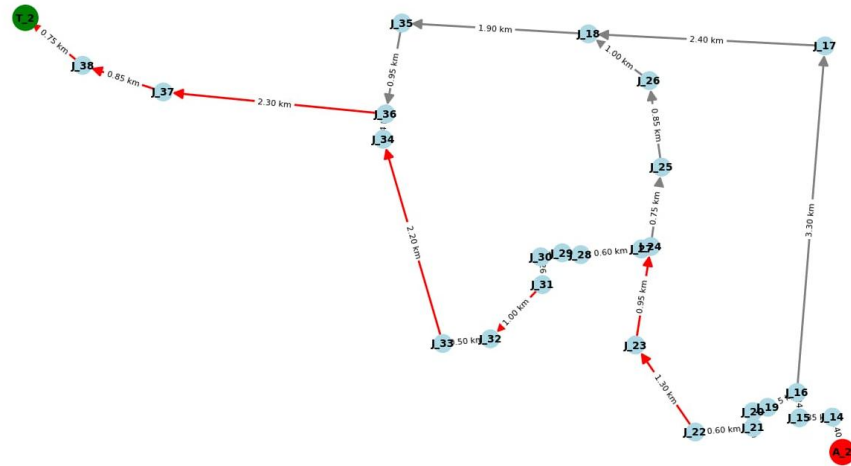
$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, J_{32}, J_{33}, J_{34}, J_{36}, J_{37}, J_{38}, T_2], 13, 72].$$

Berikut adalah hasil rute terpendek dari RSU Premagana ke RSD Mangusada menggunakan fungsi heuristik *Haversine*, yaitu sebagai berikut:

$$A_2 \rightarrow J_{14} \rightarrow J_{15} \rightarrow J_{16} \rightarrow J_{19} \rightarrow J_{20} \rightarrow J_{21} \rightarrow J_{22} \rightarrow J_{23} \rightarrow J_{24} \rightarrow J_{27} \rightarrow J_{28} \rightarrow J_{29} \rightarrow J_{30} \rightarrow J_{31} \rightarrow J_{32} \rightarrow J_{33} \rightarrow J_{34} \rightarrow J_{36} \rightarrow J_{37} \rightarrow J_{38} \rightarrow T_2$$

diperoleh dengan jarak dari A_2 menuju T_2 yaitu 13,72 km.

Berikut representasi graf berarah berbobot yang ditampilkan pada Gambar 4.8, menunjukkan hasil rute terpendek dari RSUD Premagana ke RSD Mangusada menggunakan fungsi heuristik *Haversine*.



Gambar 4.8 Graf Hasil Rute Terpendek dari RSUD Premagana ke RSD Mangusada (*Haversine*)

Fungsi Heuristik *Vincenty*

A_2 merupakan simpul awal untuk menuju RS Tujuan, sehingga A_2 dapat dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2], 0]$$

Karena hanya J_{14} yang sejalur dilewati, maka J_{14} dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2, J_{14}], 0,40]$$

Karena hanya J_{15} yang sejalur dilewati, maka J_{15} dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_2, J_{14}, J_{15}], 0,75]$$

Karena hanya J_{16} yang sejalur dilewati, maka J_{16} dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [A_2, J_{14}, J_{15}, J_{16}], 0,99]$$

Selanjutnya, akan ditentukan titik-titik mana saja yang memiliki jalur utama yang terhubung dengan J_{16} , serta dilakukan perhitungan nilai heuristik untuk masing-masing titik tersebut. Setelah perhitungan, diperoleh hasil berupa panjang rute dan nilai heuristik sebagai berikut:

$$Terpilih = \{J_{17}: (3,30, 8,02), J_{19}: (0,35, 8,28)\}$$

Selanjutnya, kita melakukan seleksi melalui perhitungan untuk memilih nilai yang terkecil.

$$\begin{aligned} f(J_{17}) &= g(J_{17}) + h(J_{17}) \\ &= (0,40 + 0,35 + 0,24 + 3,30) + 8,02 \\ &= 4,29 + 8,02 \\ &= 12,31 \end{aligned}$$

$$\begin{aligned} f(J_{19}) &= g(J_{19}) + h(J_{19}) \\ &= (0,40 + 0,35 + 0,24 + 0,35) + 8,28 \\ &= 1,34 + 8,22 \\ &= 9,56 \end{aligned}$$

Hasil perhitungan yang didapat menunjukkan bahwa J_{19} memiliki nilai terendah, sehingga dapat dimasukkan ke dalam *list* tutup sebagai berikut:

$$List\ Tutup = [A_2, J_{14}, J_{15}, J_{16}, J_{19}], 1,34]$$

Karena hanya J_{20} yang sejalur dilewati, maka J_{20} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}], 1,50]$$

J_{20} hanya terhubung dengan J_{21} , sedangkan J_{21} terhubung langsung dengan J_{22} , J_{22} terhubung langsung dengan J_{23} , dan J_{23} terhubung langsung dengan J_{24} . Sehingga $J_{21}, J_{22}, J_{23}, J_{24}$ langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}], 4,50]$$

Sama seperti dalam proses perhitungan sebelumnya, akan dicari titik-titik yang jalur utamanya terhubung dengan J_{24} , dan hasilnya adalah sebagai berikut.

$$Terpilih = \{J_{25}: (0,75, 6,52), J_{27}: (0,10, 6,53)\}$$

$$\begin{aligned} f(J_{25}) &= g(J_{25}) + h(J_{25}) \\ &= (0,40 + 0,35 + 0,24 + 0,35 + 0,16 + 0,15 + 0,60 + 1,30 + \\ &\quad 0,95 + 0,75) + 6,52 \\ &= 5,25 + 6,52 \\ &= 11,77 \end{aligned}$$

$$\begin{aligned} f(J_{27}) &= g(J_{27}) + h(J_{27}) \\ &= (0,40 + 0,35 + 0,24 + 0,35 + 0,16 + 0,15 + 0,60 + 1,30 + \\ &\quad 0,95 + 0,10) + 6,53 \\ &= 4,60 + 6,53 \\ &= 11,13 \end{aligned}$$

Dengan menggunakan proses yang sama. Diperoleh hasil bahwa titik dengan perhitungan terendah adalah J_{27} . Oleh karena itu, J_{27} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}], 4,60]$$

J_{27} hanya terhubung dengan J_{28} , sedangkan J_{28} terhubung langsung dengan J_{29} , dan J_{29} terhubung langsung dengan J_{30} . Sehingga J_{28}, J_{29}, J_{30} langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$\text{List Tutup} = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}], \\ 5, 62]$$

J_{30} hanya terhubung dengan J_{31} , sedangkan J_{31} terhubung langsung dengan J_{32} , J_{32} terhubung langsung dengan J_{33} , J_{33} terhubung langsung dengan J_{34} dan J_{34} terhubung langsung dengan J_{36} . Sehingga $J_{31}, J_{32}, J_{33}, J_{34}, J_{36}$ langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$\text{List Tutup} = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, \\ J_{32}, J_{33}, J_{34}, J_{36}], 9, 82]$$

Karena hanya J_{37} yang sejalur dilewati, maka J_{37} dimasukkan ke dalam *List* tutup.

$$\text{List Tutup} = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, \\ J_{32}, J_{33}, J_{34}, J_{36}, J_{37}], 12, 12]$$

Karena hanya J_{38} yang sejalur dilewati, maka J_{38} dimasukkan ke dalam *List* tutup.

$$\text{List Tutup} = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31}, \\ J_{32}, J_{33}, J_{34}, J_{36}, J_{37}, J_{38}], 12, 97]$$

Sama halnya, karena hanya T_2 yang sejalur dilewati dan merupakan jalur RS tujuan yang menandakan telah sampainya dilokasi tujuan, maka T_2 dimasukkan ke dalam *List* tutup sekaligus didapatkan rute terpendeknya:

$$\text{List Tutup} = [[A_2, J_{14}, J_{15}, J_{16}, J_{19}, J_{20}, J_{21}, J_{22}, J_{23}, J_{24}, J_{27}, J_{28}, J_{29}, J_{30}, J_{31},$$

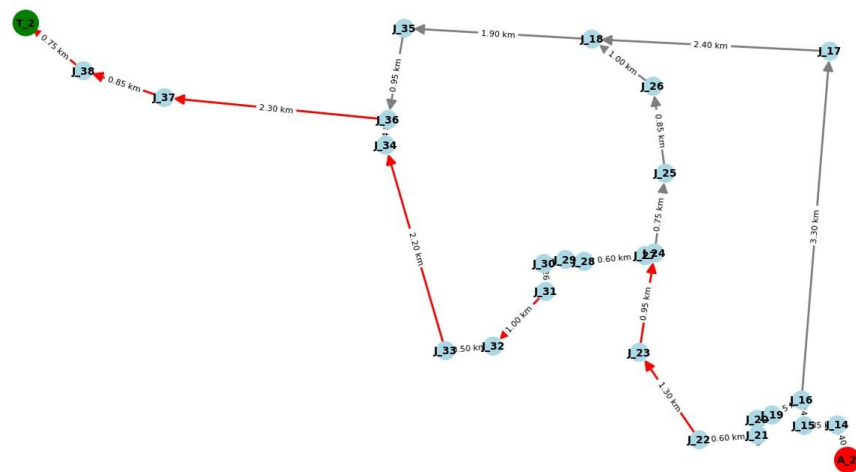
$$J_{32}, J_{33}, J_{34}, J_{36}, J_{37}, J_{38}, T_2], 13,72].$$

Berikut adalah hasil rute terpendek dari RSU Premagana ke RSD Mangusada menggunakan fungsi heuristik *Haversine*, yaitu sebagai berikut:

$$A_2 \rightarrow J_{14} \rightarrow J_{15} \rightarrow J_{16} \rightarrow J_{19} \rightarrow J_{20} \rightarrow J_{21} \rightarrow J_{22} \rightarrow J_{23} \rightarrow J_{24} \rightarrow J_{27} \rightarrow J_{28} \\ \rightarrow J_{29} \rightarrow J_{30} \rightarrow J_{31} \rightarrow J_{32} \rightarrow J_{33} \rightarrow J_{34} \rightarrow J_{36} \rightarrow J_{37} \rightarrow J_{38} \rightarrow T_2$$

Diperoleh dengan jarak dari A_2 menuju T_2 yaitu 13,72 km.

Berikut representasi graf berarah berbobot yang ditampilkan pada Gambar 4.9, menunjukkan hasil rute terpendek dari RSU Premagana ke RSD Mangusada menggunakan fungsi heuristik *Vincenty*.



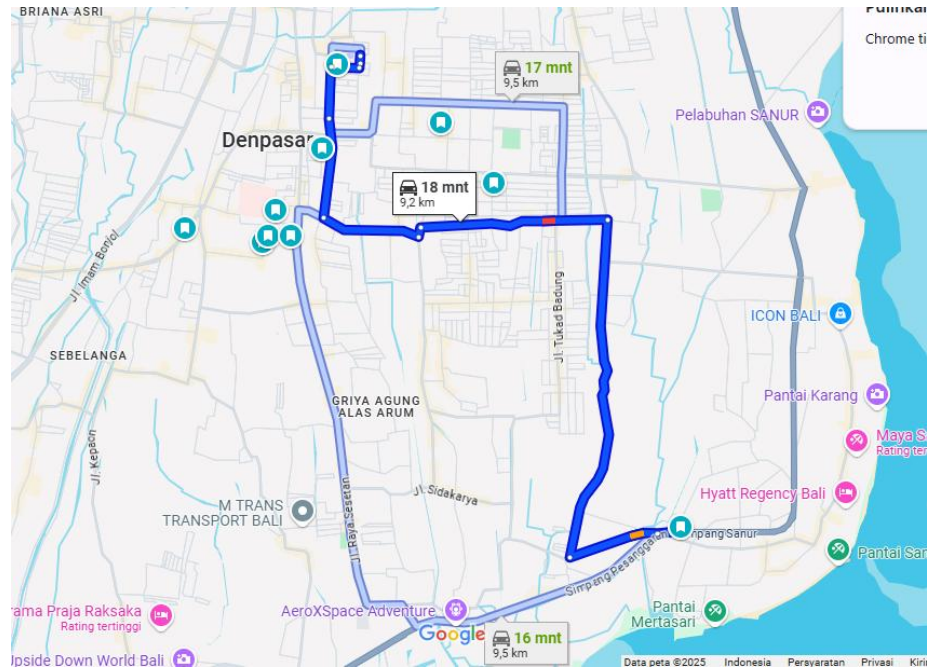
Gambar 4.9 Graf Hasil Rute Terpendek dari RSU Premagana ke RSD Mangusada (*Vincenty*)

3. Penentuan Rute $A_3 \rightarrow T_3$

a. Pengumpulan Data Titik dan Titik Persimpangan

Pada tahap awal, dilakukan visualisasi peta yang menampilkan lokasi rumah sakit awal (RS TK. II Udayana), rumah sakit tujuan (RSUD Bali Mandara) serta titik-titik persimpangan jalan yang menjadi jalur penghubung antar kedua rumah sakit tersebut. Berikut Gambar 4.10, merupakan peta rute yang

menghubungkan RS TK. II Udayana ke RSUD Bali Mandara beserta titik persimpangan jalan utama yang dilalui.



Gambar 4.10 Peta Rute dari RS TK. II Udayana ke RSUD Bali Mandara

Selanjutnya, koordinat *latitude* dan *longitude* dari rumah sakit awal, rumah sakit tujuan, serta titik-titik persimpangan jalan tersebut diidentifikasi dan disajikan dalam tabel berikut sebagai dasar pemodelan graf dan perhitungan fungsi heuristik.

Tabel 4.23 RS Awal (RS TK. II Udayana) ke RS Tujuan (RSUD Bali Mandara)

No	Nama Lokasi	Simbol	<i>Latitude</i>	<i>Longitude</i>
1	RS TK. II Udayana	A_3	-8,663486	115,218641
2	RSUD Bali Mandara	T_3	-8,703646	115,248806

Tabel 4.23, menyajikan pasangan rumah sakit yang digunakan, yaitu RS TK. II Udayana sebagai simpul awal dan RSUD Bali Mandara sebagai simpul tujuan.

Selanjutnya pada Tabel 4.24, terdapat titik koordinat persimpangan jalan beserta simbolnya.

Tabel 4.24 Titik-titik Persimpangan Jalan (RS TK. II Udayana ke RSUD Bali Mandara)

Simbol	Latitude	Longitude	Simbol	Latitude	Longitude
J_{39}	-8,662906	115,220683	J_{53}	-8,677176	115,238378
J_{40}	-8,662283	115,220729	J_{54}	-8,677040	115,217468
J_{41}	-8,662120	115,218221	J_{55}	-8,678651	115,225794
J_{42}	-8,663580	115,220653	J_{56}	-8,677801	115,225957
J_{43}	-8,664049	115,220609	J_{57}	-8,676315	115,215292
J_{44}	-8,663902	115,218105	J_{58}	-8,677369	115,215232
J_{45}	-8,668413	115,217942	J_{59}	-8,710465	115,221285
J_{46}	-8,669307	115,218046	J_{60}	-8,710658	115,221405
J_{47}	-8,669453	115,221809	J_{61}	-8,710554	115,224349
J_{48}	-8,666905	115,224476	J_{62}	-8,712424	115,225626
J_{49}	-8,667329	115,238521	J_{63}	-8,677157	115,242373
J_{50}	-8,668867	115,238491	J_{64}	-8,706521	115,239054
J_{51}	-8,671899	115,238402	J_{65}	-8,704405	115,246748
J_{52}	-8,673115	115,238343			

Daftar nama jalan yang menghubungkan simpul-simpul tersebut disajikan dalam Tabel 4.25, dan digunakan sebagai jalur dalam graf berarah berbobot.

Tabel 4.25 Nama Jalan Antar Persimpangan (RS TK. II Udayana ke RSUD Bali Mandara)

Dari	Ke	Nama Jalan	Dari	Ke	Nama Jalan
A_3	J_{39}	Jl. P.B. Sudirman	J_{52}	J_{53}	Jl. Pemuda
J_{39}	J_{40}	Jl. P.B. Sudirman III	J_{53}	J_{63}	Jl. Tukad Yeh Aya
J_{39}	J_{42}	Jl. P.B. Sudirman Tim	J_{54}	J_{55}	Jl. Waturenggong
J_{40}	J_{41}	Jl. Taman Kanak-kanak Kartika	J_{54}	J_{57}	Jl. Waturenggong
J_{41}	J_{44}	Jl. P.B. Sudirman	J_{55}	J_{56}	Jl. Tukad Ijo Gading
J_{42}	J_{43}	Jl. Sudirman VII	J_{56}	J_{53}	Jl. Tukad Yeh Aya
J_{43}	J_{44}	Jl. RSAD	J_{57}	J_{58}	Jl. Diponegoro
J_{44}	J_{45}	Jl. P.B. Sudirman	J_{58}	J_{59}	Jl. Raya Sesetan
J_{45}	J_{46}	Jl. P.B. Sudirman	J_{59}	J_{60}	Jl. Raya Sesetan
J_{46}	J_{47}	Jl. Raya Puputan	J_{60}	J_{61}	Jl. Suwung Batan Kendal
J_{46}	J_{54}	Jl. P.B. Sudirman	J_{61}	J_{62}	Jl. Pamelisan

Dari	Ke	Nama Jalan	Dari	Ke	Nama Jalan
J_{47}	J_{48}	Jl. Letda Tantular	J_{62}	J_{65}	Jl. Bypass Ngurah Rai
J_{48}	J_{49}	Jl. Cok Agung Tresna	J_{63}	J_{64}	Jl. Tukad Balian
J_{49}	J_{50}	Jl. Prof Moh. Yamin	J_{64}	J_{65}	Jl. Danau Tempe
J_{50}	J_{51}	Jl. Prof Moh. Yamin	J_{65}	T_3	Jl. Bypass Ngurah Rai
J_{51}	J_{52}	Jl. Prof Moh. Yamin			

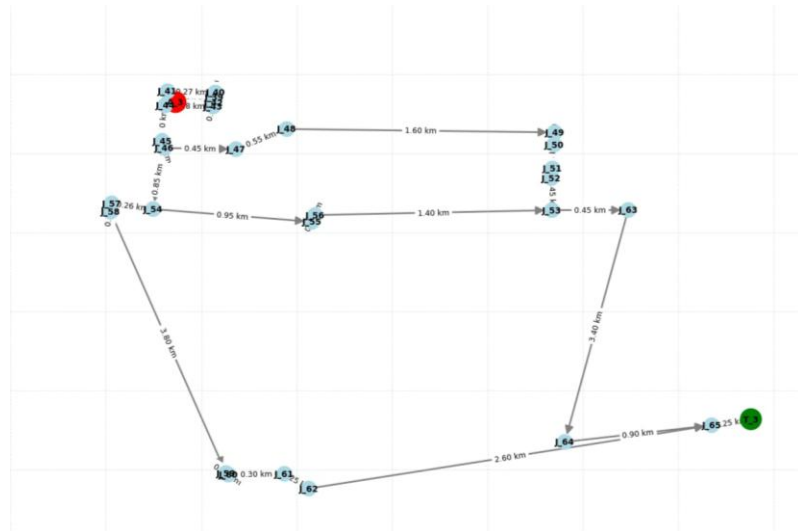
Tabel 4.26 adalah data jarak aktual ($g(n)$) dari RS TK. II Udayana ke RSUD Bali Mandara menggunakan *Google Maps*.

Tabel 4.26 Data Jarak Aktual ($g(n)$) (RS TK. II Udayana ke RSUD Bali Mandara)

Titik 1	Titik 2	Jarak (km)	Titik 1	Titik 2	Jarak (km)
A_3	J_{39}	0,10	J_{52}	J_{53}	0,45
J_{39}	J_{40}	0,07	J_{53}	J_{63}	0,45
J_{39}	J_{42}	0,08	J_{54}	J_{55}	0,95
J_{40}	J_{41}	0,27	J_{54}	J_{57}	0,26
J_{41}	J_{44}	0,21	J_{55}	J_{56}	0,10
J_{42}	J_{43}	0,05	J_{56}	J_{53}	1,40
J_{43}	J_{44}	0,28	J_{57}	J_{58}	0,12
J_{44}	J_{45}	0,50	J_{58}	J_{59}	3,80
J_{45}	J_{46}	0,10	J_{59}	J_{60}	0,02
J_{46}	J_{47}	0,45	J_{60}	J_{61}	0,30
J_{46}	J_{54}	0,85	J_{61}	J_{62}	0,25
J_{47}	J_{48}	0,55	J_{62}	J_{65}	2,60
J_{48}	J_{49}	1,60	J_{63}	J_{64}	3,40
J_{49}	J_{50}	0,17	J_{64}	J_{65}	0,90
J_{50}	J_{51}	0,35	J_{65}	T_3	0,25
J_{51}	J_{52}	0,13			

b. Perhitungan Fungsi Heuristik

Berikut Gambar 4.11, merupakan representasi dalam bentuk graf berarah berbobot yang dibentuk berdasarkan peta rute dari RS TK. II Udayana menuju RSUD Bali Mandara.



Gambar 4.11 Graf Berarah Berbobot (RS TK. II Udayana ke RSUD Bali Mandara)

Perhitungan Heuristik *Haversine*

Berikut Tabel 4.27, yaitu data jarak antar simpul menggunakan fungsi heuristik *Haversine* yang dihitung menggunakan *google collab*. Berdasarkan perhitungan jarak menggunakan rumus *Haversine* yang ditampilkan pada Tabel 4.27, langkah selanjutnya adalah membuktikan bahwa nilai heuristik tersebut memenuhi sifat *consistent* dan *admissible*.

Tabel 4.27 Data Heuristik Menggunakan Formula *Haversine* (RS TK. II Udayana ke RSUD Bali Mandara)

Titik 1	Titik 2	Jarak (KM)	Titik 1	Titik 2	Jarak (KM)
A_3	T_3	5,56	J_{52}	T_3	3,58
J_{39}	T_3	5,48	J_{53}	T_3	3,15
J_{40}	T_3	5,53	J_{54}	T_3	4,54
J_{41}	T_3	5,71	J_{55}	T_3	3,75
J_{42}	T_3	5,42	J_{56}	T_3	3,81
J_{43}	T_3	5,38	J_{57}	T_3	4,77
J_{44}	T_3	5,56	J_{58}	T_3	4,70
J_{45}	T_3	5,18	J_{59}	T_3	3,11
J_{46}	T_3	5,10	J_{60}	T_3	3,10
J_{47}	T_3	4,82	J_{61}	T_3	2,80
J_{48}	T_3	4,88	J_{62}	T_3	2,72
J_{49}	T_3	4,19	J_{63}	T_3	3,02
J_{50}	T_3	4,03	J_{64}	T_3	1,11
J_{51}	T_3	3,71	J_{65}	T_3	0,24

Pembuktian *Consistency* Heuristik *Haversine*

Berikut akan dibuktikan bahwa nilai heuristik *Haversine* pada Tabel 4.27 terbukti *consistent*.

Tabel 4.28 Perhitungan Konsistensi Heuristik *Haversine* (RS TK. II Udayana ke RSUD Bali Mandara)

Pasangan Simpul ($n \rightarrow n'$)	$h(n)$ (km)	$c(n, n')$ (km)	$h(n')$ (km)	$c(n, n') + h(n')$ (km)	Konsisten? (Apakah $h(n) \leq c(n, n') + h(n')$?)
$A_3 \rightarrow J_{39}$	5,56	0,10	5,48	5,58	Ya, $5,56 \leq 5,58$
$J_{39} \rightarrow J_{40}$	5,48	0,07	5,53	5,60	Ya, $5,48 \leq 5,60$
$J_{39} \rightarrow J_{42}$	5,48	0,08	5,42	5,50	Ya, $5,48 \leq 5,50$
$J_{40} \rightarrow J_{41}$	5,53	0,27	5,71	5,98	Ya, $5,54 \leq 5,98$
$J_{41} \rightarrow J_{44}$	5,71	0,21	5,56	5,77	Ya, $5,71 \leq 5,77$
$J_{42} \rightarrow J_{43}$	5,42	0,05	5,38	5,43	Ya, $5,42 \leq 5,43$
$J_{43} \rightarrow J_{44}$	5,38	0,28	5,56	5,84	Ya, $5,38 \leq 5,84$
$J_{44} \rightarrow J_{45}$	5,56	0,50	5,18	5,68	Ya, $5,56 \leq 5,68$
$J_{45} \rightarrow J_{46}$	5,18	0,10	5,10	5,20	Ya, $5,18 \leq 5,20$
$J_{46} \rightarrow J_{47}$	5,10	0,45	4,82	5,27	Ya, $5,10 \leq 5,27$
$J_{46} \rightarrow J_{54}$	5,10	0,85	4,54	5,39	Ya, $5,10 \leq 5,39$
$J_{47} \rightarrow J_{48}$	4,82	0,55	4,88	5,43	Ya, $4,82 \leq 5,43$
$J_{48} \rightarrow J_{49}$	4,88	1,60	4,19	5,79	Ya, $4,88 \leq 5,79$
$J_{49} \rightarrow J_{50}$	4,19	0,17	4,03	4,20	Ya, $4,19 \leq 4,20$
$J_{50} \rightarrow J_{51}$	4,03	0,35	3,71	4,06	Ya, $4,03 \leq 4,06$
$J_{51} \rightarrow J_{52}$	3,71	0,13	3,58	3,71	Ya, $3,71 \leq 3,71$
$J_{52} \rightarrow J_{53}$	3,58	0,45	3,15	3,59	Ya, $3,58 \leq 3,59$
$J_{53} \rightarrow J_{63}$	3,15	0,45	3,02	3,47	Ya, $3,15 \leq 3,47$
$J_{54} \rightarrow J_{55}$	4,54	0,95	3,75	4,70	Ya, $4,54 \leq 4,70$
$J_{54} \rightarrow J_{57}$	4,54	0,26	4,77	5,03	Ya, $4,54 \leq 5,03$
$J_{55} \rightarrow J_{56}$	3,75	0,10	3,81	3,91	Ya, $3,75 \leq 3,91$
$J_{56} \rightarrow J_{53}$	3,81	1,40	3,15	4,55	Ya, $3,81 \leq 4,55$
$J_{57} \rightarrow J_{58}$	4,77	0,12	4,70	4,80	Ya, $4,77 \leq 4,80$
$J_{58} \rightarrow J_{59}$	4,70	3,80	3,11	6,91	Ya, $4,70 \leq 6,91$
$J_{59} \rightarrow J_{60}$	3,11	0,02	3,10	3,12	Ya, $3,11 \leq 3,12$
$J_{60} \rightarrow J_{61}$	3,10	0,30	2,80	3,10	Ya, $3,10 \leq 3,10$
$J_{61} \rightarrow J_{62}$	2,80	0,25	2,72	2,97	Ya, $2,80 \leq 2,97$
$J_{62} \rightarrow J_{65}$	2,72	2,60	0,24	2,84	Ya, $2,72 \leq 2,84$
$J_{63} \rightarrow J_{64}$	3,02	3,40	1,12	4,52	Ya, $3,02 \leq 4,52$
$J_{64} \rightarrow J_{65}$	1,12	0,90	0,24	1,14	Ya, $1,12 \leq 1,14$
$J_{65} \rightarrow T_3$	0,24	0,25	0,00	0,25	Ya, $0,24 \leq 0,25$

Dari Tabel 4.28, semua nilai $h(n)$ memenuhi kondisi konsistensi $h(n) \leq c(n, n') + h(n')$. Dengan demikian, heuristik ini konsisten (*consistent*).

Pembuktian *Admissibility* Heuristik *Haversine*

Langkah berikutnya adalah membuktikan bahwa heuristik pada Tabel 4.27 juga *admissible*.

Tabel 4.29 Perhitungan *Admissible* Heuristik *Haversine* (RS TK. II Udayana ke RSUD Bali Mandara)

Pasangan Simpul	Heuristik <i>Haversine</i> $h(n)$ (km)	Jarak Aktual Terpendek $g^*(n)$ (km)	<i>Admissible?</i> (Apakah $h(n) \leq g^*(n)$?)
A_3	5,56	9,41	Ya, $5,56 \leq 9,41$
J_{39}	5,48	9,31	Ya, $5,48 \leq 9,31$
J_{40}	5,53	9,38	Ya, $5,53 \leq 9,38$
J_{41}	5,71	9,11	Ya, $5,71 \leq 9,11$
J_{42}	5,42	9,23	Ya, $5,42 \leq 9,23$
J_{43}	5,38	8,18	Ya, $5,38 \leq 8,18$
J_{44}	5,56	8,90	Ya, $5,56 \leq 8,90$
J_{45}	5,18	8,40	Ya, $5,18 \leq 8,40$
J_{46}	5,10	8,30	Ya, $5,10 \leq 8,30$
J_{47}	4,82	8,25	Ya, $4,82 \leq 8,25$
J_{48}	4,88	7,70	Ya, $4,88 \leq 7,70$
J_{49}	4,19	6,10	Ya, $4,19 \leq 6,10$
J_{50}	4,03	5,93	Ya, $4,03 \leq 5,93$
J_{51}	3,71	5,58	Ya, $3,71 \leq 5,58$
J_{52}	3,58	5,45	Ya, $3,58 \leq 5,45$
J_{53}	3,15	5,00	Ya, $3,15 \leq 5,00$
J_{54}	4,54	7,45	Ya, $4,54 \leq 7,45$
J_{55}	3,75	6,50	Ya, $3,75 \leq 6,50$
J_{56}	3,81	6,40	Ya, $3,81 \leq 6,40$
J_{57}	4,77	7,34	Ya, $4,77 \leq 7,34$
J_{58}	4,70	7,22	Ya, $4,70 \leq 7,22$
J_{59}	3,11	3,42	Ya, $3,11 \leq 3,42$
J_{60}	3,10	3,40	Ya, $3,10 \leq 3,40$
J_{61}	2,80	3,10	Ya, $2,80 \leq 3,10$
J_{62}	2,72	2,85	Ya, $2,72 \leq 2,85$
J_{63}	3,02	4,55	Ya, $3,02 \leq 4,55$
J_{64}	1,11	1,15	Ya, $1,11 \leq 1,15$
J_{65}	0,24	0,25	Ya, $0,24 \leq 0,25$

Dari Tabel 4.29, terlihat bahwa untuk semua simpul, nilai heuristik *Haversine* tidak pernah melebihi jarak aktual terpendek ke tujuan. Dengan demikian, heuristik ini memenuhi syarat *admissible*.

Jadi, nilai heuristik *Haversine* yang digunakan pada algoritma pencarian rute ini terbukti *consistent* (Tabel 4.28) dan *admissible* (Tabel 4.29). Kedua sifat ini menjamin bahwa algoritma A-*Star* yang menggunakan heuristik ini dapat menemukan jalur terpendek secara optimal dan efisien.

Perhitungan Heuristik *Vincenty*

Berikut Tabel 4.30, adalah data jarak antar titik menggunakan fungsi heuristik *Vincenty* yang dihitung menggunakan *google collab*.

Tabel 4.30 Data Heuristik Menggunakan Formula *Vincenty* (RS TK. II Udayana ke RSUD Bali Mandara)

Titik 1	Titik 2	Jarak (KM)	Titik 1	Titik 2	Jarak (KM)
A_3	T_3	5,54	J_{52}	T_3	3,56
J_{39}	T_3	5,46	J_{53}	T_3	3,14
J_{40}	T_3	5,52	J_{54}	T_3	4,53
J_{41}	T_3	5,69	J_{55}	T_3	3,74
J_{42}	T_3	5,40	J_{56}	T_3	3,80
J_{43}	T_3	5,36	J_{57}	T_3	4,76
J_{44}	T_3	5,54	J_{58}	T_3	4,70
J_{45}	T_3	5,16	J_{59}	T_3	3,12
J_{46}	T_3	5,08	J_{60}	T_3	3,11
J_{47}	T_3	4,80	J_{61}	T_3	2,79
J_{48}	T_3	4,86	J_{62}	T_3	2,72
J_{49}	T_3	4,17	J_{63}	T_3	3,01
J_{50}	T_3	4,01	J_{64}	T_3	1,11
J_{51}	T_3	3,69	J_{65}	T_3	0,24

Berdasarkan perhitungan jarak menggunakan rumus *Vincenty* yang ditampilkan pada Tabel 4.30, langkah selanjutnya adalah membuktikan bahwa nilai heuristik tersebut memenuhi sifat *consistent* dan *admissible*.

Pembuktian *Consistency* Heuristik *Vincenty*

Berikut akan dibuktikan bahwa nilai heuristik *Vincenty* pada Tabel 4.30 terbukti *consistent*.

Tabel 4.31 Perhitungan Konsistensi Heuristik *Vincenty* (RS TK. II Udayana ke RSUD Bali Mandara)

Pasangan Simpul ($n \rightarrow n'$)	$h(n)$ (km)	$c(n, n')$ (km)	$h(n')$ (km)	$c(n, n') + h(n')$ (km)	Konsisten? (Apakah $h(n) \leq c(n, n') + h(n')$?)
$A_3 \rightarrow J_{39}$	5,54	0,10	5,46	5,56	Ya, $5,54 \leq 5,56$
$J_{39} \rightarrow J_{40}$	5,46	0,07	5,52	5,59	Ya, $5,46 \leq 5,59$
$J_{39} \rightarrow J_{42}$	5,46	0,08	5,40	5,48	Ya, $5,46 \leq 5,48$
$J_{40} \rightarrow J_{41}$	5,52	0,27	5,69	5,96	Ya, $5,52 \leq 5,96$
$J_{41} \rightarrow J_{44}$	5,69	0,21	5,54	5,75	Ya, $5,69 \leq 5,75$
$J_{42} \rightarrow J_{43}$	5,40	0,05	5,36	5,41	Ya, $5,40 \leq 5,41$
$J_{43} \rightarrow J_{44}$	5,36	0,28	5,54	5,82	Ya, $5,36 \leq 5,82$
$J_{44} \rightarrow J_{45}$	5,54	0,50	5,16	5,66	Ya, $5,54 \leq 5,66$
$J_{45} \rightarrow J_{46}$	5,16	0,10	5,08	5,18	Ya, $5,16 \leq 5,18$
$J_{46} \rightarrow J_{47}$	5,08	0,45	4,80	5,25	Ya, $5,08 \leq 5,25$
$J_{46} \rightarrow J_{54}$	5,08	0,85	4,53	5,38	Ya, $5,08 \leq 5,38$
$J_{47} \rightarrow J_{48}$	4,80	0,55	4,86	5,41	Ya, $4,80 \leq 5,41$
$J_{48} \rightarrow J_{49}$	4,86	1,60	4,17	5,77	Ya, $4,86 \leq 5,77$
$J_{49} \rightarrow J_{50}$	4,17	0,17	4,01	4,18	Ya, $4,17 \leq 4,18$
$J_{50} \rightarrow J_{51}$	4,01	0,35	3,69	4,04	Ya, $4,01 \leq 4,04$
$J_{51} \rightarrow J_{52}$	3,69	0,13	3,56	3,69	Ya, $3,69 \leq 3,69$
$J_{52} \rightarrow J_{53}$	3,56	0,45	3,14	3,59	Ya, $3,56 \leq 3,59$
$J_{53} \rightarrow J_{63}$	3,14	0,45	3,01	3,46	Ya, $3,14 \leq 3,46$
$J_{54} \rightarrow J_{55}$	4,53	0,95	3,74	4,69	Ya, $4,53 \leq 4,69$
$J_{54} \rightarrow J_{57}$	4,53	0,26	4,76	5,02	Ya, $4,53 \leq 5,02$
$J_{55} \rightarrow J_{56}$	3,74	0,10	3,80	3,90	Ya, $3,74 \leq 3,90$
$J_{56} \rightarrow J_{53}$	3,80	1,40	3,14	4,54	Ya, $3,80 \leq 4,54$
$J_{57} \rightarrow J_{58}$	4,76	0,12	4,70	4,82	Ya, $4,76 \leq 4,82$
$J_{58} \rightarrow J_{59}$	4,70	3,80	3,12	6,92	Ya, $4,70 \leq 6,92$
$J_{59} \rightarrow J_{60}$	3,12	0,02	3,11	3,13	Ya, $3,12 \leq 3,13$
$J_{60} \rightarrow J_{61}$	3,11	0,30	2,79	3,09	Ya, $3,11 \leq 3,09$
$J_{61} \rightarrow J_{62}$	2,79	0,25	2,72	2,97	Ya, $2,79 \leq 2,97$
$J_{62} \rightarrow J_{65}$	2,72	2,60	0,24	2,84	Ya, $2,72 \leq 2,84$
$J_{63} \rightarrow J_{64}$	3,01	3,40	1,11	4,51	Ya, $3,02 \leq 4,51$
$J_{64} \rightarrow J_{65}$	1,11	0,90	0,24	1,14	Ya, $1,12 \leq 1,14$
$J_{65} \rightarrow T_3$	0,24	0,25	0,00	0,25	Ya, $0,24 \leq 0,25$

Dari Tabel 4.31, semua nilai $h(n)$ memenuhi kondisi kondisi konsistensi $h(n) \leq c(n, n') + h(n')$. Dengan demikian, heuristik ini konsisten (*consistent*).

Pembuktian *Admissibility* Heuristik *Vincenty*

Langkah berikutnya adalah membuktikan bahwa heuristik pada Tabel 4.30 juga *admissible*.

Tabel 4.32 Perhitungan *Admissible* Heuristik *Vincenty* (RS TK. II Udayana ke RSUD Bali Mandara)

Pasangan Simpul	Heuristik <i>Vincenty</i> $h(n)$ (km)	Jarak Aktual Terpendek $g^*(n)$ (km)	<i>Admissible?</i> (Apakah $h(n) \leq g^*(n)$?)
A_3	5,54	9,41	Ya, $5,54 \leq 9,41$
J_{39}	5,46	9,31	Ya, $5,46 \leq 9,31$
J_{40}	5,52	9,38	Ya, $5,52 \leq 9,38$
J_{41}	5,69	9,11	Ya, $5,69 \leq 9,11$
J_{42}	5,40	9,23	Ya, $5,40 \leq 9,23$
J_{43}	5,36	8,18	Ya, $5,36 \leq 8,18$
J_{44}	5,54	8,90	Ya, $5,54 \leq 8,90$
J_{45}	5,16	8,40	Ya, $5,16 \leq 8,40$
J_{46}	5,08	8,30	Ya, $5,08 \leq 8,30$
J_{47}	4,80	8,25	Ya, $4,80 \leq 8,25$
J_{48}	4,86	7,70	Ya, $4,86 \leq 7,70$
J_{49}	4,17	6,10	Ya, $4,17 \leq 6,10$
J_{50}	4,01	5,93	Ya, $4,01 \leq 5,93$
J_{51}	3,69	5,58	Ya, $3,69 \leq 5,58$
J_{52}	3,56	5,45	Ya, $3,56 \leq 5,45$
J_{53}	3,14	5,00	Ya, $3,14 \leq 5,00$
J_{54}	4,53	7,45	Ya, $4,53 \leq 7,45$
J_{55}	3,74	6,50	Ya, $3,74 \leq 6,50$
J_{56}	3,80	6,40	Ya, $3,80 \leq 6,40$
J_{57}	4,76	7,34	Ya, $4,76 \leq 7,34$
J_{58}	4,70	7,22	Ya, $4,70 \leq 7,22$
J_{59}	3,12	3,42	Ya, $3,12 \leq 3,42$
J_{60}	3,11	3,40	Ya, $3,11 \leq 3,40$
J_{61}	2,79	3,10	Ya, $2,79 \leq 3,10$
J_{62}	2,72	2,85	Ya, $2,72 \leq 2,85$
J_{63}	3,01	4,55	Ya, $3,01 \leq 4,55$
J_{64}	1,11	1,15	Ya, $1,11 \leq 1,15$
J_{65}	0,24	0,25	Ya, $0,24 \leq 0,25$

Dari Tabel 4.32, terlihat bahwa untuk semua simpul, nilai heuristik *Vincenty* tidak pernah melebihi jarak aktual terpendek ke tujuan. Dengan demikian, heuristik ini memenuhi syarat *admissible*.

Jadi, nilai heuristik *Vincenty* yang digunakan pada algoritma pencarian rute ini terbukti *consistent* (Tabel 4.31) dan *admissible* (Tabel 4.32). Kedua sifat ini menjamin bahwa algoritma A-Star yang menggunakan heuristik ini dapat menemukan jalur terpendek secara optimal dan efisien.

c. Implementasi Algoritma A-Star

Fungsi Heuristik *Haversine*

A_3 Merupakan simpul awal sehingga A_3 kita masukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_3], 0]$$

A_3 hanya terhubung dengan J_{39} sehingga J_{39} langsung kita masukkan ke *list* tutup menjadi:

$$List\ Tutup = [[A_3, J_{39}], 0, 10]$$

Selanjutnya, akan ditentukan titik-titik mana saja yang memiliki jalur utama yang terhubung dengan J_{39} , serta dilakukan perhitungan nilai heuristik untuk masing-masing titik tersebut. Setelah perhitungan, diperoleh hasil berupa panjang rute dan nilai heuristik sebagai berikut:

$$Terpilih = \{J_{40}: (0,07, 5,53), J_{42}: (0,08, 5,42)\}$$

Selanjutnya, kita melakukan seleksi melalui perhitungan untuk memilih nilai yang terkecil.

$$\begin{aligned} f(J_{40}) &= g(J_{40}) + h(J_{40}) \\ &= (0,10 + 0,07) + 5,54 \end{aligned}$$

$$= 0,17 + 5,54$$

$$= 5,71$$

$$f(J_{42}) = g(J_{42}) + h(J_{42})$$

$$= (0,10 + 0,08) + 5,42$$

$$= 0,18 + 5,42$$

$$= 5,60$$

Hasil perhitungan yang didapat menunjukkan bahwa J_{42} memiliki nilai terendah, sehingga dapat dimasukkan ke dalam *list* tutup sebagai berikut:

$$List\ Tutup = [A_3, J_{39}, J_{42}, 0,18]$$

Karena hanya J_{43} yang sejalur dilewati, maka J_{43} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, 0,23]$$

Karena hanya J_{44} yang sejalur dilewati, maka J_{44} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, 0,51]$$

J_{44} terhubung langsung dengan J_{45} , sehingga J_{45} langsung dimasukkan ke *list* tutup. Selanjutnya kita tuliskan:

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, 1,01]$$

Sama halnya dengan proses sebelumnya, karna J_{45} hanya terhubung langsung dengan J_{46} , maka J_{46} dimasukkan ke *list* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, 1,11]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan J_{46} , kemudian didapatkan hasil:

$$\text{Terpilih} = \{J_{47}: (0,45, 4,80), J_{54}: (0,85, 4,53)\}$$

$$f(J_{47}) = g(J_{47}) + h(J_{47})$$

$$= (0,10 + 0,07 + 0,05 + 0,28 + 0,50 + 0,10 + 0,45) + 4,80$$

$$= 1,55 + 4,80$$

$$= 6,35$$

$$f(J_{54}) = g(J_{54}) + h(J_{54})$$

$$= (0,10 + 0,07 + 0,05 + 0,28 + 0,50 + 0,10 + 0,85) + 4,53$$

$$= 1,95 + 4,53$$

$$= 6,48$$

Hasil yang diperoleh menunjukkan bahwa J_{47} memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$\text{List Tutup} = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}], 1,55]$$

J_{47} terhubung langsung dengan J_{48} , sehingga J_{48} langsung dimasukkan ke *list* tutup. Selanjutnya kita tuliskan:

$$\text{List Tutup} = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}], 2,10]$$

Sama halnya dengan proses sebelumnya, karna J_{48} hanya terhubung langsung dengan J_{49} , maka J_{49} dimasukkan ke *list* tutup.

$$\text{List Tutup} = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}], 3,70]$$

J_{49} hanya terhubung dengan J_{50} , sedangkan J_{50} terhubung langsung dengan J_{51} , J_{51} terhubung langsung dengan J_{52} , dan J_{52} terhubung langsung dengan J_{53} . Sehingga $J_{50}, J_{51}, J_{52}, J_{53}$ langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$\text{List Tutup} = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}], 4,80]$$

Karena hanya J_{63} yang sejalur dilewati, maka J_{63} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}], 5, 25]$$

J_{63} terhubung langsung dengan J_{64} , sehingga J_{64} langsung dimasukkan ke *list* tutup. Selanjutnya kita tuliskan:

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}, J_{64}], 8, 65]$$

Sama halnya dengan proses sebelumnya, karna J_{64} hanya terhubung langsung dengan J_{65} , maka J_{65} dimasukkan ke *list* tutup.

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}, J_{64}, J_{65}], 9, 55]$$

J_{65} terhubung langsung dengan T_3 , yang menandakan bahwa telah sampainya ke lokasi tujuan, maka T_3 dimasukkan ke dalam *List* tutup sekaligus didapatkan rute terpendeknya:

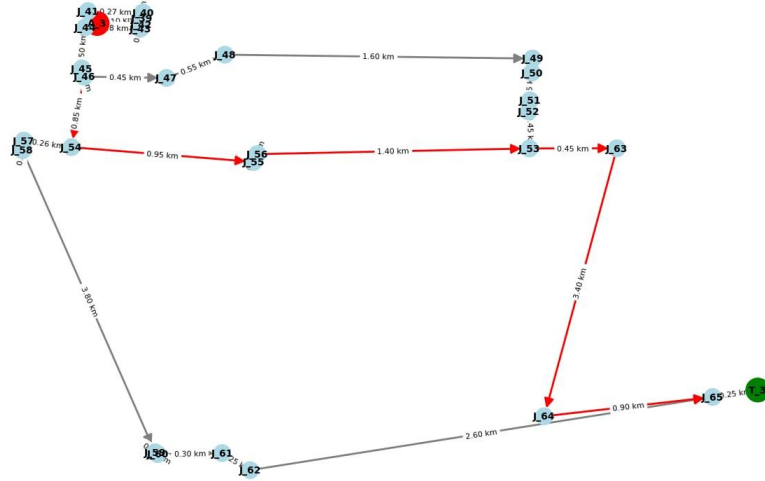
$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}, J_{64}, J_{65}, T_3], 9, 80]$$

Berikut adalah hasil rute terpendek dari RS TK. II Udayana ke RSUD Bali Mandara menggunakan fungsi heuristik *Haversine*, yaitu sebagai berikut:

$$A_3 \rightarrow J_{39} \rightarrow J_{42} \rightarrow J_{43} \rightarrow J_{44} \rightarrow J_{45} \rightarrow J_{46} \rightarrow J_{47} \rightarrow J_{48} \rightarrow J_{49} \rightarrow J_{50} \rightarrow J_{51} \\ \rightarrow J_{52} \rightarrow J_{53} \rightarrow J_{63} \rightarrow J_{64} \rightarrow J_{65} \rightarrow T_3$$

Diperoleh dengan jarak dari A_3 menuju T_3 yaitu 9,80 km.

Berikut representasi graf berarah berbobot yang ditampilkan pada Gambar 4.12, menunjukkan hasil rute terpendek dari RS TK. II Udayana ke RSUD Bali Mandara menggunakan fungsi heuristik *Haversine*.



Gambar 4.12 Graf Hasil Rute Terpendek dari RS TK. II Udayana ke RSUD Bali Mandara (*Haversine*)

Fungsi Heuristik *Vincenty*

A_3 merupakan simpul awal sehingga A_3 kita masukkan ke dalam *List Tutup*.

$$List\ Tutup = [[A_3], 0]$$

A_3 hanya terhubung dengan J_{39} sehingga J_{39} langsung kita masukkan ke *list tutup* menjadi:

$$List\ Tutup = [[A_3, J_{39}], 0, 10]$$

Selanjutnya, akan ditentukan titik-titik mana saja yang memiliki jalur utama yang terhubung dengan J_{39} , serta dilakukan perhitungan nilai heuristik untuk masing-masing titik tersebut. Setelah perhitungan, diperoleh hasil berupa panjang rute dan nilai heuristik sebagai berikut:

$$Terpilih = \{J_{40}: (0,07, 5,52), J_{42}: (0,08, 5,40)\}$$

Selanjutnya, kita melakukan seleksi melalui perhitungan untuk memilih nilai yang terkecil.

$$\begin{aligned}
 f(J_{40}) &= g(J_{40}) + h(J_{40}) \\
 &= (0,10 + 0,07) + 5,52 \\
 &= 0,17 + 5,52 \\
 &= 5,69
 \end{aligned}$$

$$\begin{aligned}
 f(J_{42}) &= g(J_{42}) + h(J_{42}) \\
 &= (0,10 + 0,08) + 5,40 \\
 &= 0,18 + 5,40 \\
 &= 5,58
 \end{aligned}$$

Hasil perhitungan yang didapat menunjukkan bahwa J_{42} memiliki nilai terendah, sehingga dapat dimasukkan ke dalam *list* tutup sebagai berikut:

$$List\ Tutup = [A_3, J_{39}, J_{42}], 0,18]$$

Karena hanya J_{43} yang sejalur dilewati, maka J_{43} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}], 0,23]$$

Karena hanya J_{44} yang sejalur dilewati, maka J_{44} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}], 0,51]$$

J_{44} terhubung langsung dengan J_{45} , sehingga J_{45} langsung dimasukkan ke *list* tutup. Selanjutnya kita tuliskan:

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}], 1,01]$$

Sama halnya dengan proses sebelumnya, karna J_{45} hanya terhubung langsung dengan J_{46} , maka J_{46} dimasukkan ke *list* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}], 1,11]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan J_{46} , kemudian didapatkan hasil:

$$Terpilih = \{J_{47}: (0,45, 4,80), J_{54}: (0,85, 4,53)\}$$

$$\begin{aligned} f(J_{47}) &= g(J_{47}) + h(J_{47}) \\ &= (0,10 + 0,07 + 0,05 + 0,28 + 0,50 + 0,10 + 0,45) + 4,80 \\ &= 1,55 + 4,80 \\ &= 6,35 \end{aligned}$$

$$\begin{aligned} f(J_{54}) &= g(J_{54}) + h(J_{54}) \\ &= (0,10 + 0,07 + 0,05 + 0,28 + 0,50 + 0,10 + 0,85) + 4,53 \\ &= 1,95 + 4,53 \\ &= 6,48 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa J_{47} memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}], 1,55]$$

J_{47} terhubung langsung dengan J_{48} , sehingga J_{48} langsung dimasukkan ke *list* tutup. Selanjutnya kita tuliskan:

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}], 2,10]$$

Sama halnya dengan proses sebelumnya, karna J_{48} hanya terhubung langsung dengan J_{49} , maka J_{49} dimasukkan ke *list* tutup.

$$List\ Tutup = [A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}], 3,70]$$

J_{49} hanya terhubung dengan J_{50} , sedangkan J_{50} terhubung langsung dengan J_{51} , J_{51} terhubung langsung dengan J_{52} , dan J_{52} terhubung langsung dengan J_{53} . Sehingga $J_{50}, J_{51}, J_{52}, J_{53}$ langsung dapat dimasukkan ke *list* tutup, yaitu sebagai berikut:

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}], 4, 80]$$

Karena hanya J_{63} yang sejalur dilewati, maka J_{63} dimasukkan ke dalam *List* tutup.

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}], 5, 25]$$

J_{63} terhubung langsung dengan J_{64} , sehingga J_{64} langsung dimasukkan ke *list* tutup. Selanjutnya kita tuliskan:

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}, J_{64}], 8, 65]$$

Sama halnya dengan proses sebelumnya, karna J_{64} hanya terhubung langsung dengan J_{65} , maka J_{65} dimasukkan ke *list* tutup.

$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}, J_{64}, J_{65}], 9, 55]$$

J_{65} terhubung langsung dengan T_3 , yang menandakan bahwa telah sampainya ke lokasi tujuan, maka T_3 dimasukkan ke dalam *List* tutup sekaligus didapatkan rute terpendeknya:

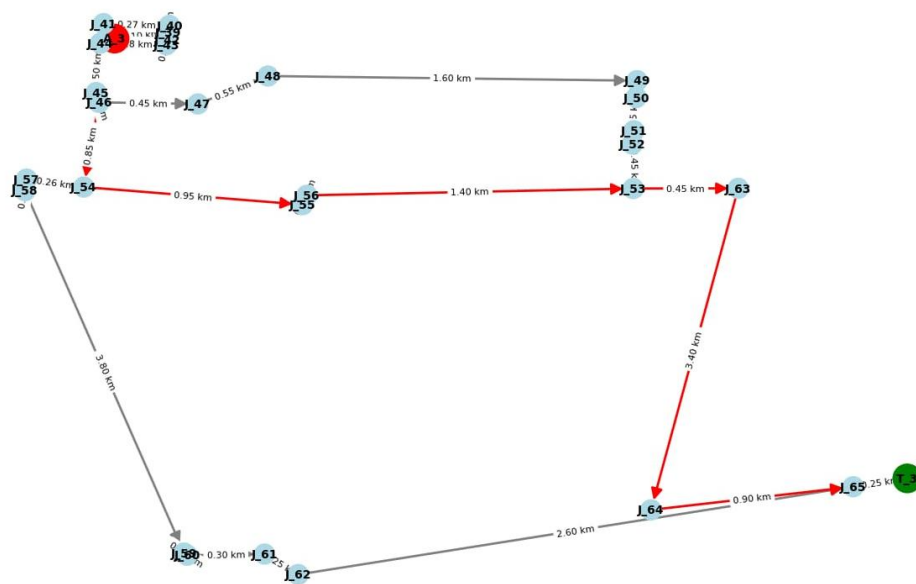
$$List\ Tutup = [[A_3, J_{39}, J_{42}, J_{43}, J_{44}, J_{45}, J_{46}, J_{47}, J_{48}, J_{49}, J_{50}, J_{51}, J_{52}, J_{53}, J_{63}, J_{64}, J_{65}, T_3], 9, 80]$$

Berikut adalah hasil rute terpendek dari RS TK. II Udayana ke RSUD Bali Mandara menggunakan fungsi heuristik *vincenty*, yaitu sebagai berikut:

$$A_3 \rightarrow J_{39} \rightarrow J_{42} \rightarrow J_{43} \rightarrow J_{44} \rightarrow J_{45} \rightarrow J_{46} \rightarrow J_{47} \rightarrow J_{48} \rightarrow J_{49} \rightarrow J_{50} \rightarrow J_{51} \\ \rightarrow J_{52} \rightarrow J_{53} \rightarrow J_{63} \rightarrow J_{64} \rightarrow J_{65} \rightarrow T_3$$

Diperoleh dengan jarak dari A_3 menuju T_3 yaitu 9,80 km.

Berikut representasi graf berarah berbobot yang ditampilkan pada Gambar 4.13, menunjukkan hasil rute terpendek dari RS TK. II Udayana ke RSUD Bali Mandara menggunakan fungsi heuristik *Vincenty*.



Gambar 4.13 Graf Hasil Rute Terpendek dari RS TK. II Udayana ke RSUD Bali Mandara (*Vincenty*)

4.4 Analisis Perbandingan Implementasi Algoritma A-Star dengan Menggunakan Metode *Haversine* dan *Vincenty*

Pada bagian ini dilakukan analisis terhadap hasil implementasi algoritma A-Star yang menggunakan dua fungsi heuristik, yaitu metode *Haversine* dan *Vincenty*, dalam menentukan rute terpendek antar rumah sakit di Bali. Tujuan dari analisis ini

adalah untuk membandingkan keakuratan dan efektivitas kedua heuristik berdasarkan hasil rute dan jarak yang diperoleh.

1. Hasil Perhitungan Algoritma A-Star dengan metode *Haversine* dan *Vincenty*

Berikut pada Tabel 4.33, merupakan hasil jalur dan jarak tempuh yang dihasilkan dari implementasi algoritma A-Star menggunakan metode *Haversine* dan *Vincenty*.

Tabel 4.33 Rute dan Jarak Hasil Perhitungan A-Star (Metode *Haversine* dan *Vincenty*)

Pasangan RS	Heuristik	Rute	Jarak (km)
$A_1 \rightarrow T_1$	<i>Haversine</i>	$A_1 \rightarrow J_1 \rightarrow J_2 \rightarrow J_4 \rightarrow J_{10} \rightarrow J_{11} \rightarrow J_8 \rightarrow J_9 \rightarrow J_{12} \rightarrow J_{13} \rightarrow T_1$	3,83
$A_1 \rightarrow T_1$	<i>Vincenty</i>	$A_1 \rightarrow J_1 \rightarrow J_2 \rightarrow J_4 \rightarrow J_{10} \rightarrow J_{11} \rightarrow J_8 \rightarrow J_9 \rightarrow J_{12} \rightarrow J_{13} \rightarrow T_1$	3,83
$A_2 \rightarrow T_2$	<i>Haversine</i>	$A_2 \rightarrow J_{14} \rightarrow J_{15} \rightarrow J_{16} \rightarrow J_{19} \rightarrow J_{20} \rightarrow J_{21} \rightarrow J_{22} \rightarrow J_{23} \rightarrow J_{24} \rightarrow J_{27} \rightarrow J_{28} \rightarrow J_{29} \rightarrow J_{30} \rightarrow J_{31} \rightarrow J_{32} \rightarrow J_{33} \rightarrow J_{34} \rightarrow J_{36} \rightarrow J_{37} \rightarrow J_{38} \rightarrow T_2$	13,72
$A_2 \rightarrow T_2$	<i>Vincenty</i>	$A_2 \rightarrow J_{14} \rightarrow J_{15} \rightarrow J_{16} \rightarrow J_{19} \rightarrow J_{20} \rightarrow J_{21} \rightarrow J_{22} \rightarrow J_{23} \rightarrow J_{24} \rightarrow J_{27} \rightarrow J_{28} \rightarrow J_{29} \rightarrow J_{30} \rightarrow J_{31} \rightarrow J_{32} \rightarrow J_{33} \rightarrow J_{34} \rightarrow J_{36} \rightarrow J_{37} \rightarrow J_{38} \rightarrow T_2$	13,72
$A_3 \rightarrow T_3$	<i>Haversine</i>	$A_3 \rightarrow J_{39} \rightarrow J_{42} \rightarrow J_{43} \rightarrow J_{44} \rightarrow J_{45} \rightarrow J_{46} \rightarrow J_{47} \rightarrow J_{48} \rightarrow J_{49} \rightarrow J_{50} \rightarrow J_{51} \rightarrow J_{52} \rightarrow J_{53} \rightarrow J_{63} \rightarrow J_{64} \rightarrow J_{65} \rightarrow T_3$	9,80
$A_3 \rightarrow T_3$	<i>Vincenty</i>	$A_3 \rightarrow J_{39} \rightarrow J_{42} \rightarrow J_{43} \rightarrow J_{44} \rightarrow J_{45} \rightarrow J_{46} \rightarrow J_{47} \rightarrow J_{48} \rightarrow J_{49} \rightarrow J_{50} \rightarrow J_{51} \rightarrow J_{52} \rightarrow J_{53} \rightarrow J_{63} \rightarrow J_{64} \rightarrow J_{65} \rightarrow T_3$	9,80

Hasil dari Tabel 4.33, menunjukkan bahwa baik metode *Haversine* maupun *Vincenty* menghasilkan rute dan jarak tempuh yang identik untuk seluruh pasangan rumah sakit yang di uji. Hal ini menunjukkan bahwa kedua metode

heuristik memberikan hasil yang sama dalam penerapan algoritma *A-Star* pada graf yang digunakan dalam penelitian ini.

2. Pembahasan dan Analisis

Berdasarkan hasil implementasi algoritma *A-Star* dengan kedua metode heuristik, dapat disimpulkan bahwa:

- a. Heuristik *Haversine* dan *Vincenty* menghasilkan rute serta jarak yang sama untuk semua pasangan rumah sakit yang diteliti, sehingga dalam konteks graf dan data geografis yang digunakan, keduanya memiliki tingkat keakuratan estimasi jarak yang setara.
- b. Dari segi perhitungan, metode *Haversine* memiliki keunggulan dalam hal kesederhanaan dan kecepatan komputasi karena hanya mengasumsikan bentuk bumi sebagai bola sempurna. Sebaliknya, metode *Vincenty* mempertimbangkan bentuk elipsoid bumi sehingga lebih kompleks dalam perhitungan, namun tidak memberikan perbedaan hasil dalam studi ini.

Dengan demikian, implementasi algoritma *A-Star* berhasil menemukan rute terpendek antar rumah sakit yang sama baik dengan menggunakan metode heuristik *Haversine* maupun *Vincenty*, karena keduanya menghasilkan rute dan jarak yang identik pada seluruh pasangan rumah sakit yang diuji.

4.5 Integrasi Nilai-nilai Islam dalam Penerapan Algoritma *A-Star* untuk Penyelamatan Jiwa

Penelitian ini tidak hanya berfokus pada aspek teknis penentuan rute terpendek antar rumah sakit, tetapi juga berupaya mengintegrasikan nilai-nilai Islam yang luhur dalam setiap langkahnya. Sebagaimana telah dijelaskan pada bab

sebelumnya, Islam sangat menekankan pentingnya perlindungan jiwa (*hifz an-nafs*) sebagai salah satu tujuan utama *maqashid syariah*. Prinsip ini menuntut setiap individu dan masyarakat untuk berikhtiar secara maksimal dalam menjaga dan menyelamatkan nyawa manusia.

Dalam hasil penelitian ini, penggunaan algoritma *A-Star* untuk mempercepat akses layanan kesehatan merupakan bentuk nyata implementasi nilai tersebut. Dengan memanfaatkan teknologi untuk menentukan rute terpendek, peluang untuk menyelamatkan jiwa menjadi lebih besar. Hal ini sejalan dengan firman Allah SWT dalam QS. Al-Isra': 33 yang menegaskan larangan membunuh jiwa tanpa alasan yang benar dan pentingnya menjaga keselamatan manusia. Selain itu, QS. Al-Ma'idah: 32 juga menegaskan bahwa menyelamatkan satu jiwa sama dengan menyelamatkan seluruh umat manusia, yang menjadi landasan moral bagi setiap inovasi yang bertujuan untuk kemaslahatan.

Lebih lanjut, integrasi nilai Islam dalam penelitian ini dapat diperkuat dengan sabda Rasulullah SAW dalam hadits yang diriwayatkan oleh Imam Muslim:

عَنْ أَبِي هُرَيْرَةَ رَضِيَ اللَّهُ عَنْهُ قَالَ، قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ: مَنْ نَفَسَ عَنْ مُؤْمِنٍ كُرْبَةً مِنْ كُرْبِ الدُّنْيَا نَفَسَ اللَّهُ عَنْهُ كُرْبَةً مِنْ كُرْبِ يَوْمِ الْقِيَامَةِ، وَمَنْ يَسَّرَ عَلَى مُعْسِرٍ يَسَّرَ اللَّهُ عَلَيْهِ فِي الدُّنْيَا وَالْآخِرَةِ، وَمَنْ سَتَرَ مُسْلِمًا سَتَرَهُ اللَّهُ فِي الدُّنْيَا وَالْآخِرَةِ، وَاللَّهُ فِي عَوْنِ الْعَبْدِ مَا كَانَ الْعَبْدُ فِي عَوْنِ أَخِيهِ،

(Hasan, 1421 H)

Artinya: Dari Abu Hurairah radhiyallahu ‘anhu dia berkata: Rasulullah shallallahu ‘alaihi wa sallam bersabda: “Barangsiapa yang menghilangkan satu kesulitan seorang mukmin yang lain dari kesusulitannya di dunia, niscaya Allah akan menghilangkan darinya satu kesulitan pada hari kiamat. Barangsiapa yang meringankan orang yang kesusahan, niscaya Allah akan meringankan baginya (urusannya) di dunia dan akhirat.” (HR. Muslim, No. 2699)

Hadits ini mengandung pesan mendalam tentang pentingnya membantu sesama, khususnya dalam kondisi darurat dan penuh kesulitan. Penerapan algoritma *A-Star* untuk mempercepat akses layanan kesehatan merupakan salah satu bentuk

nyata dari upaya memudahkan urusan orang lain, yang sangat dianjurkan dalam Islam. Dengan mempercepat penanganan pasien darurat, kita tidak hanya memenuhi tuntutan profesionalisme dan kemanusiaan, tetapi juga menjalankan perintah agama untuk saling menolong dalam kebaikan dan takwa.

Hasil penelitian yang menunjukkan efektivitas algoritma *A-Star* dalam menentukan rute terpendek juga menjadi bukti bahwa teknologi dapat menjadi sarana untuk mengamalkan nilai-nilai Islam secara praktis. Setiap inovasi yang bertujuan untuk menyelamatkan jiwa dan meningkatkan kemaslahatan umat merupakan bagian dari ibadah dan amal saleh yang bernilai tinggi di sisi Allah SWT. Dalam Islam, segala bentuk ikhtiar yang dilakukan untuk mencegah bahaya, mempercepat pertolongan, dan menjaga keselamatan hidup manusia sangat dianjurkan dan mendapatkan ganjaran pahala yang besar.

Dengan demikian, penelitian ini tidak hanya memberikan kontribusi dalam bidang ilmu pengetahuan dan teknologi, tetapi juga menjadi bagian dari upaya mewujudkan *maqashid syariah*, khususnya perlindungan jiwa. Implementasi algoritma *A-Star* dalam sistem layanan kesehatan merupakan wujud nyata dari integrasi antara sains dan nilai-nilai Islam, di mana setiap langkah inovasi diarahkan untuk kemaslahatan dan perlindungan kehidupan manusia. Hal ini menjadi bukti bahwa Islam sangat mendorong pemanfaatan teknologi untuk tujuan mulia, serta menegaskan pentingnya kolaborasi antara ilmu pengetahuan, teknologi, dan nilai agama dalam membangun peradaban yang lebih baik dan beradab.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil evaluasi terhadap perbandingan heuristik *Haversine* dan *Vincenty* dalam algoritma A-*Star*, diperoleh bahwa keduanya menghasilkan rute dan jarak tempuh yang identik untuk setiap pasangan rumah sakit yang diteliti. Hal ini menunjukkan bahwa baik metode *Haversine* maupun *Vincenty* memiliki tingkat keakuratan estimasi jarak yang sama. Dengan hasil tersebut, dapat disimpulkan bahwa tidak terdapat perbedaan akurasi antara kedua metode heuristik dalam proses pencarian rute terpendek menggunakan algoritma A-*Star*.

Implementasi algoritma A-*Star* dalam penentuan rute terpendek antar rumah sakit di Bali dilakukan dengan membangun graf berarah berbobot yang merepresentasikan jaringan jalan antar simpul, terdiri dari titik-titik persimpangan dan lokasi rumah sakit. Bobot antar simpul dihitung berdasarkan koordinat geografis menggunakan metode heuristik *Haversine* dan *Vincenty*. Hasil implementasi pada tiga pasangan rumah sakit menunjukkan bahwa algoritma A-*Star* berhasil menemukan rute terpendek yang konsisten dengan hasil rute dan jarak yang sama untuk kedua metode heuristik. Hal ini menunjukkan bahwa algoritma A-*Star* dapat diimplementasikan secara tepat untuk kasus penentuan rute terpendek berdasarkan representasi graf dan data geografis yang tersedia.

5.2 Saran

Untuk pengembangan penelitian selanjutnya, disarankan agar sistem pencarian rute terpendek mengintegrasikan data kondisi lalu lintas secara *real-time* dan

memperluas cakupan jaringan jalan dengan menambahkan lebih banyak titik dan jalur alternatif agar hasil pencarian menjadi lebih akurat dan representatif. Selain itu, eksplorasi sebagai fungsi heuristik lain serta evaluasi kinerja algoritma dalam hal efisiensi waktu dan penggunaan sumber daya sangat penting untuk meningkatkan performa sistem. Dengan demikian, sistem pencarian rute dapat menjadi lebih optimal, responsif, dan bermanfaat untuk berbagai aplikasi navigasi di masa depan.

DAFTAR PUSTAKA

- Abdurahman, D., & Maulana, D. I. (2018). *Implementasi Algoritma A-Star Untuk Pemetaan Lokasi Sarana Kesehatan Kabupaten Majalengka Berbasis Geographic Information System (GIS)*. 57–65.
- Ardiansyah, M., Gasba, M., & Harlinda, L. (2022). *Implementasi Algoritma A * (A Star) dalam Menentukan Jarak Terpendek Menuju Rumah Sakit Rujukan Covid-19*. 3(3), 203–212.
- Arsyad, M. A., Supriyadi, D., Anggie, V., Hidayah, L. N., & Putri, D. (2019). *Penerapan Algoritma A Star Untuk Pencarian Rute Terpendek Puskesmas Rawat Inap Di*.
- Cahyono, A. D. (2022). *Penerapan Algoritma a-Star Untuk Mencari Rute Terpendek Destinasi Wisata Budaya Kota Yogyakarta*.
- Dalem, I. B. Gede Wahyu Antara. (2018). Penerapan Algoritma A* (Star) Menggunakan Graph Untuk Menghitung Jarak Terpendek. *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, 1(1), 41–47. <https://doi.org/10.31598/jurnalresistor.v1i1.253>
- Diana, E. L., Suryaningtyas, W., & Suprpti, E. (2016). Pengaturan Lampu Lalu Lintas di Persimpangan Jalan Ahmad Yani Giant dengan Aplikasi Pewarnaan Teori Graf. *MUST: Journal of Mathematics Education, Science and Technology*, 1(1), 69. <https://doi.org/10.30651/must.v1i1.99>
- Harahap, R. H. (2021). *Prinsip maqashid asy-syariah dalam Undang-Undang no 35 tahun 2014 tentang perlindungan anak*. 35. <http://etd.uinsyahada.ac.id/7880/%0Ahttp://etd.uinsyahada.ac.id/7880/1/1710700014.pdf>
- Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. In *IEEE Transactions on Systems Science and Cybernetics* (Vol. 4, Issue 2, pp. 100–107). <https://doi.org/10.1109/TSSC.1968.300136>
- Hasan, Abu. (1421 H). Hadits Arba'in Nawawi. Diakses tanggal 22 Mei 2025, dari <https://haditsarbain.com/>
- Hermawan, H., & Setiyani, H. (2019). Implementasi Algoritma a-Star Pada Permainan Komputer Roguelike Berbasis Unity. *Jurnal Algoritma, Logika Dan Komputasi*, 2(1), 111–120. <https://doi.org/10.30813/j-alu.v2i1.1571>
- Kementrian Agama. (2022). Qur'an Kemenag. Diakses pada 3 Desember 2024, dari <https://quran.kemenag.go.id/>
- Marcelina, D., & Yulianti, E. (2020). *Aplikasi Pencarian Rute Terpendek Lokasi Kuliner Khas Palembang Menggunakan Algoritma Euclidean Distance dan A *(Star)*. 09, 195–202.
- Marsudi. 2016. Teori Graf. *Universitas Brawijaya Press*.

- Mauliddiyah, N. L. (2021). *Pemilihan Jalur Terpendek Pada Game Candi Borobudur Menggunakan Algoritma A**.
- Munir, R. (2023). *Pendahuluan. Revisi Buku Matematika Diskrit 2012, Update*.
- Nursaily, S., & Agustian, H. (2023). Penerapan Metode Algoritma A* Untuk Penentuan Jalur Terpendek Dalam Pengiriman Barang Berbasis Mobile. *TeknoIS: Jurnal Ilmiah Teknologi Informasi Dan Sains*, 13(1), 101–109. <https://doi.org/10.36350/jbs.v13i1.179>
- Panjaitan, D. J., & Aprilia, R. (2022). *Teori Graf*. Medan: LPPM UMNAW.
- Prasetyo, A. C., Arnandi, M. P., Hudnanto, H. S., & Setiaji, B. (2019). Perbandingan Algoritma Astar dan Dijkstra Dalam Menentukan Rute Terdekat. *Sisfotenika*, 9(1), 36. <https://doi.org/10.30700/jst.v9i1.456>
- Putra, A. B. W., Rachman, A. A., Santoso, A., & Mulyanto, M. (2020). Perbandingan Hasil Rute Terdekat Antar Rumah Sakit di Samarinda Menggunakan Algoritma A*(star) dan Floyd-Warshall. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 9(1), 59–68. <https://doi.org/10.32736/sisfokom.v9i1.685>
- Ratnawati, D and Liliana, D and Regasari, R and Muflikhah, L. (2012). Modul Bahan Ajar Kecerdasan Buatan. *Program Teknologi Innformasi Dan Ilmmu Komputer Universitas Brawijaya*, 12.
- Shihab, M. Q. (2002). *Tafsir Al-Mishbah: Pesan, Kesan dan Keserasian Al-Qur'an* (Vol. 7). Jakarta: Lentera Hati.
- Slamin. (2019). *Teori Graf dan Aplikasinya (Edisi Kedua)*. In *Living Spiritual Quotient*.
- Sriyanto, N. I., Maulana, F. A., Aprilian, R., Christian, E., Hart, P., Nilsson, N., & Raphael, B. (2024). *Pemanfaatan Algoritma A-Star dalam Menentukan Rute Wisata di Kota Palangka Raya Utilisation of A-Star Algorithm in Determining Tourism Routes in Palangka Raya City*. 12(1).
- Studi, P., & Informatika, T. (2024). *Navigasi Pergerakan Robot Otonom*.
- Syihabuddin, R. F., Jauhari, M. N., Khudzaifah, M., & Fahmi, H. (2022). Implementasi Algoritma A-Star dalam Menentukan Rute Terpendek Destinasi Wisata Kota Malang. *Jurnal Riset Mahasiswa Matematika*, 1(5), 236–245. <https://doi.org/10.18860/jrmm.v1i5.14497>
- Tiwa, S., A. (2018). Aplikasi Tempat Kulier Berdasarkan Jenis Manan Wilayah Kota Tangerang Menggunakan Algoritma A-Star Berbasis Android. *Skripsi. Teknik Informatika Fakultas Sains dan Teknologi Universitas Buddhi Dharma Tangerang*.
- Tryastie, M., Nelda, L., Priskila, R., & Pranatawijaya, V. H. (2024). *Implementasi Algoritma A * (Star) untuk Menentukan Rute Jarak Terpendek*. 2, 449–458.
- Wati, N., & Permatasari, S. H. (2019). Perbandingan Algoritma a* Dengan Algoritma Dijkstra Untuk Pencarian Jarak Dan Rute Terpendek Berbasis Web.

Jurnal Teknologi Informasi Indonesia (JTII), 4(1), 1–6.
<https://doi.org/10.30869/jtii.v4i1.371>

- Windarni, V. A., & Setiawan, A. (2022). Comparative Analysis of Vincenty and Geodesic Method Approaches in Measuring the Distance Between Subdistrict Offices in Salatiga City. *Barekeng: Jurnal Ilmu Matematika Dan Terapan*, 16 (4). <https://doi.org/10.30598/barekengvol16iss4pp1207-1220>
- Wiratama Putra, I. P. A., & Wibawa I Gede Arta. (2023). Implementasi Algoritma A* (Star) dengan Graf untuk Menentukan Rute Terpendek Distributor Kopi. *Jnatia*, 1, 1053–1062.
<https://ojs.unud.ac.id/index.php/jnatia/article/view/102421>
- Yamin, M., & Talai, M. B. (2015). Aplikasi Pencarian Jalur Terpendek Pada Rumah Sakit Umum Bahteramas Menggunakan Algoritma a* (a-Star). *Jurnal Informatika*, 9(2). <https://doi.org/10.26555/jifo.v9i2.a2961>
- Yuliana, Ananda, & Surya, I. (2012). Implementasi Algoritma A Star pada Pemecahan Puzzle 8. *Jurnal Teknik Informatika*, 1(September), 1–9.

LAMPIRAN

Lampiran 1. *Script Code*

Graf Rute dari RSU Balimed ke RSUD Wangaya

```
import networkx as nx
import matplotlib.pyplot as plt

# Data koordinat RS dan titik persimpangan (latitude, longitude)
coordinates = {
    'A1': (-8.656791, 115.190929),
    'T1': (-8.648779, 115.213433),
    'J1': (-8.656874, 115.191602),
    'J2': (-8.651537, 115.195444),
    'J3': (-8.643664, 115.196089),
    'J4': (-8.653478, 115.204416),
    'J5': (-8.644745, 115.199734),
    'J6': (-8.650553, 115.204403),
    'J7': (-8.650553, 115.208147),
    'J8': (-8.650567, 115.208876),
    'J9': (-8.650636, 115.210993),
    'J10': (-8.654456, 115.208534),
    'J11': (-8.651968, 115.208723),
    'J12': (-8.652119, 115.210811),
    'J13': (-8.652438, 115.213279),
}

# Data edges dengan bobot jarak aktual (km)
edges = [
    ('A1', 'J1', 0.07),
    ('J1', 'J2', 0.80),
    ('J2', 'J3', 0.90),
    ('J2', 'J4', 1.00),
    ('J3', 'J5', 0.40),
    ('J4', 'J6', 0.35),
    ('J4', 'J10', 0.45),
    ('J5', 'J6', 0.85),
    ('J6', 'J7', 0.45),
    ('J7', 'J8', 0.08),
    ('J8', 'J9', 0.24),
    ('J9', 'J12', 0.17),
    ('J10', 'J11', 0.28),
    ('J11', 'J8', 0.15),
    ('J11', 'J12', 0.23),
    ('J12', 'J13', 0.27),
    ('J13', 'T1', 0.40),
]
```

```

# Buat graf berarah berbobot
G = nx.DiGraph()

# Tambahkan node dengan atribut posisi (longitude, latitude) untuk visualisasi
for node, (lat, lon) in coordinates.items():
    G.add_node(node, pos=(lon, lat))

# Tambahkan edge dengan bobot jarak aktual
for u, v, w in edges:
    G.add_edge(u, v, weight=w)

# Ambil posisi node untuk visualisasi
pos = nx.get_node_attributes(G, 'pos')

# Ukuran dan warna node: merah (awal), hijau (tujuan), biru (persimpangan)
node_size = [600 if n == 'A1' or n == 'T1' else 300 for n in G.nodes()]
node_color = ['red' if n == 'A1' else 'green' if n == 'T1' else 'lightblue' for n in G.nodes()]

# Gambar graf
plt.figure(figsize=(14, 10))
nx.draw_networkx_nodes(G, pos, node_size=node_size, node_color=node_color)
nx.draw_networkx_labels(G, pos, font_size=10, font_weight='bold')
nx.draw_networkx_edges(G, pos, arrowstyle='->', arrowsize=20,
edge_color='gray', width=2)

# Tampilkan bobot jarak pada edge
edge_labels = {(u, v): f'{d["weight"]:.2f} km' for u, v, d in G.edges(data=True)}
nx.draw_networkx_edge_labels(G, pos, edge_labels=edge_labels, font_size=9)

plt.title('Graf Rute dari RSUD Balimed ke RSUD Wangaya dengan Jarak Aktual')
plt.axis('off')
plt.tight_layout()
plt.show()

```

Rumus Heuristik *Haversine*

```
import math
```

```
def haversine_detailed(lat1, lon1, lat2, lon2):
```

```
    """
```

Menghitung jarak Haversine antara dua titik koordinat (latitude dan longitude) dengan menampilkan semua komponen perhitungan.

Args:

lat1, lon1: Latitude dan longitude titik pertama (dalam derajat)
lat2, lon2: Latitude dan longitude titik kedua (dalam derajat)

Returns:

dict: Semua nilai perhitungan dan jarak akhir dalam km

```

"""
# Konversi derajat ke radian
phi1 = math.radians(lat1)
lambda1 = math.radians(lon1)
phi2 = math.radians(lat2)
lambda2 = math.radians(lon2)

# Selisih koordinat dalam radian
delta_phi = phi2 - phi1
delta_lambda = lambda2 - lambda1

# Komponen rumus Haversine
a = math.sin(delta_phi / 2)**2 + math.cos(phi1) * math.cos(phi2) *
math.sin(delta_lambda / 2)**2
c = 2 * math.asin(math.sqrt(a))

# Jari-jari bumi dalam kilometer
radius_earth = 6371

# Jarak akhir
distance = radius_earth * c

# Kembalikan semua nilai sebagai dictionary
return {
    "Latitude 1 (rad)": phi1,
    "Longitude 1 (rad)": lambda1,
    "Latitude 2 (rad)": phi2,
    "Longitude 2 (rad)": lambda2,
    "Delta Phi (rad)": delta_phi,
    "Delta Lambda (rad)": delta_lambda,
    "a": a,
    "c": c,
    "Jarak (km)": distance
}

# Input koordinat
lat1 = float(input("Masukkan latitude titik pertama (derajat): "))
lon1 = float(input("Masukkan longitude titik pertama (derajat): "))
lat2 = float(input("Masukkan latitude titik kedua (derajat): "))
lon2 = float(input("Masukkan longitude titik kedua (derajat): "))

# Hitung dengan detail
hasil = haversine_detailed(lat1, lon1, lat2, lon2)

# Tampilkan hasil perhitungan
print("\n--- Detail Perhitungan Haversine ---")
print(f'Latitude 1 (rad): {hasil["Latitude 1 (rad)"]:.8f}')
print(f'Longitude 1 (rad): {hasil["Longitude 1 (rad)"]:.8f}')

```

```

print(f'Latitude 2 (rad): {hasil['Latitude 2 (rad)']:.8f}")
print(f'Longitude 2 (rad): {hasil['Longitude 2 (rad)']:.8f}")
print(f'Delta Phi (rad): {hasil['Delta Phi (rad)']:.8f}")
print(f'Delta Lambda (rad): {hasil['Delta Lambda (rad)']:.8f}")
print(f'a: {hasil['a']:.10f}")
print(f'c: {hasil['c']:.10f}")
print(f'Jarak Haversine: {hasil['Jarak (km)']:.4f} km")

```

Rumus Heuristik *Vincenty*

```
import math
```

```

def vincenty_distance(lat1, lon1, lat2, lon2, max_iter=200, tol=1e-12):
    a = 6378137.0 # semi-major axis (meter)
    f = 1 / 298.257223563 # flattening
    b = (1 - f) * a # semi-minor axis

    φ1, λ1 = math.radians(lat1), math.radians(lon1)
    φ2, λ2 = math.radians(lat2), math.radians(lon2)

    U1 = math.atan((1 - f) * math.tan(φ1))
    U2 = math.atan((1 - f) * math.tan(φ2))
    sinU1, cosU1 = math.sin(U1), math.cos(U1)
    sinU2, cosU2 = math.sin(U2), math.cos(U2)

    L = λ2 - λ1
    λ = L

    for iteration in range(max_iter):
        sinλ = math.sin(λ)
        cosλ = math.cos(λ)
        sinσ = math.sqrt((cosU2 * sinλ) ** 2 + (cosU1 * sinU2 - sinU1 * cosU2 *
cosλ) ** 2)
        if sinσ == 0:
            return 0 # Titik sama
        cosσ = sinU1 * sinU2 + cosU1 * cosU2 * cosλ
        σ = math.atan2(sinσ, cosσ)
        sinα = cosU1 * cosU2 * sinλ / sinσ
        cos2α = 1 - sinα ** 2
        if cos2α == 0:
            cos2σm = 0 # Equatorial line
        else:
            cos2σm = cosσ - 2 * sinU1 * sinU2 / cos2α
            C = f / 16 * cos2α * (4 + f * (4 - 3 * cos2α))
            λ_prev = λ
            λ = L + (1 - C) * f * sinα * (σ + C * sinσ * (cos2σm + C * cosσ * (-1 + 2 *
cos2σm ** 2)))
        if abs(λ - λ_prev) < tol:
            break

```

```

else:
    raise ValueError("Vincenty formula failed to converge")

    u2 = cos2α * (a ** 2 - b ** 2) / (b ** 2)
    A = 1 + u2 / 16384 * (4096 + u2 * (-768 + u2 * (320 - 175 * u2)))
    B = u2 / 1024 * (256 + u2 * (-128 + u2 * (74 - 47 * u2)))
    Δσ = B * sinσ * (cos2σm + B / 4 * (cosσ * (-1 + 2 * cos2σm ** 2) - B / 6 *
cos2σm * (-3 + 4 * sinσ ** 2) * (-3 + 4 * cos2σm ** 2)))

    s = b * A * (σ - Δσ) # meter
    return s / 1000 # kilometer

if __name__ == "__main__":
    print("Masukkan koordinat titik awal:")
    lat_awal = float(input("Latitude awal (-8.xxx): "))
    lon_awal = float(input("Longitude awal (115.xxx): "))

    print("\nMasukkan koordinat titik tujuan:")
    lat_tujuan = float(input("Latitude tujuan (-8.xxx): "))
    lon_tujuan = float(input("Longitude tujuan (115.xxx): "))

    jarak = vincenty_distance(lat_awal, lon_awal, lat_tujuan, lon_tujuan)
    print(f"\nJarak Vincenty antara titik awal dan tujuan adalah: {jarak:.4f} km")

```

Lampiran 2. Script Perhitungan Jarak Terpendek $g^*(n)$ ke T_3

```

import networkx as nx

# Data edges dengan bobot jarak (km) dari Tabel 4.26
edges = [
    ('A_3', 'J_39', 0.10),
    ('J_39', 'J_40', 0.07),
    ('J_39', 'J_42', 0.08),
    ('J_40', 'J_41', 0.27),
    ('J_41', 'J_44', 0.21),
    ('J_42', 'J_43', 0.05),
    ('J_43', 'J_44', 0.28),
    ('J_44', 'J_45', 0.50),
    ('J_45', 'J_46', 0.10),
    ('J_46', 'J_47', 0.45),
    ('J_46', 'J_54', 0.85),
    ('J_47', 'J_48', 0.55),
    ('J_48', 'J_49', 1.60),
    ('J_49', 'J_50', 0.17),
    ('J_50', 'J_51', 0.35),

```

```

('J_51', 'J_52', 0.13),
('J_52', 'J_53', 0.45),
('J_53', 'J_63', 0.45),
('J_54', 'J_55', 0.95),
('J_54', 'J_57', 0.26),
('J_55', 'J_56', 0.10),
('J_56', 'J_53', 1.40),
('J_57', 'J_58', 0.12),
('J_58', 'J_59', 3.80),
('J_59', 'J_60', 0.02),
('J_60', 'J_61', 0.30),
('J_61', 'J_62', 0.25),
('J_62', 'J_65', 2.60),
('J_63', 'J_64', 3.40),
('J_64', 'J_65', 0.90),
('J_65', 'T_3', 0.25),
]

# Buat graph berarah
G = nx.DiGraph()
G.add_weighted_edges_from(edges)

target = 'T_3'
shortest_distances = {}

# Hitung jarak terpendek dari setiap node ke T_3
for node in G.nodes:
    if node != target:
        try:
            dist = nx.dijkstra_path_length(G, node, target, weight='weight')
            shortest_distances[node] = dist
        except nx.NetworkXNoPath:
            shortest_distances[node] = float('inf')

# Tampilkan hasil jarak terpendek
print("Jarak aktual terpendek g*(n) ke T_3:")
for node, dist in sorted(shortest_distances.items()):
    print(f'{node}: {dist:.2f} km
```

RIWAYAT HIDUP



Penulis yang lebih dikenal dengan nama Mahmuda, lahir di Pasuruan, 20 Agustus 2003. Penulis merupakan anak pertama dari dua bersaudara, dari pasangan Bapak Adi dan Ibu Sati. Saat ini, penulis bertempat tinggal di Desa Watulumbung, Kecamatan Lumbang, Kabupaten Pasuruan. Penulis menempuh pendidikan di RA (2006–2008), melanjutkan ke SDN Watulumbung II (2008–2014), kemudian melanjutkan pendidikan di SMP Negeri 2 Lumbang (2014–2017), dan meneruskan ke jenjang SMA di SMA Negeri 1 Lumbang (2017–2020). Pada tahun 2021, penulis diterima melalui jalur SNMPTN di Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Selama menempuh pendidikan di perguruan tinggi, penulis aktif mengikuti berbagai organisasi dan kegiatan yang menunjang pengembangan soft skill serta kemampuan kepemimpinan. Pada tahun 2023, penulis menjadi anggota Dewan Eksekutif Mahasiswa (DEMA) Fakultas Sains dan Teknologi. Selain itu, penulis turut berpartisipasi dalam beberapa kepanitiaan internal kampus, baik di tingkat jurusan maupun fakultas, sebagai bentuk kontribusi terhadap kegiatan kemahasiswaan. Pengalaman ini memberikan penulis kesempatan untuk belajar bekerja dalam tim, mengelola waktu, dan meningkatkan kemampuan komunikasi. Penulis juga mengikuti beberapa kegiatan akademik dan non-akademik yang bertujuan untuk memperluas wawasan serta mengasah keterampilan, baik di bidang keorganisasian, kepanitiaan, maupun dalam kegiatan kemasyarakatan yang bersifat sukarela.



KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

BUKTI KONSULTASI SKRIPSI

Nama : Mahmuda
NIM : 210601110008
Fakultas / Jurusan : Sains dan Teknologi / Matematika
Judul Skripsi : Implementasi Algoritma A-Star dalam Menentukan Rute Terpendek Antar Rumah Sakit di Bali
Pembimbing I : Prof. Dr. H. Turmudi, M.Si., Ph.D
Pembimbing II : Dr. Fachrur Rozi, M.Si

No	Tanggal	Hal	Tanda Tangan
1.	7 Oktober 2024	Konsultasi Topik dan Data	1.
2.	10 Oktober 2024	Konsultasi Bab I, II, dan III	2.
3.	24 Oktober 2024	Konsultasi Revisi Bab I, II, dan III	3.
4.	19 November 2024	ACC Bab I, II, dan III	4.
5.	16 Desember 2024	Konsultasi Kajian Agama Bab I dan II	5.
6.	10 Februari 2025	ACC Kajian Agama Bab I dan II	6.
7.	12 Februari 2025	ACC Seminar Proposal	7.
8.	8 April 2025	Konsultasi Revisi Seminar Proposal	8.
9.	11 April 2025	Konsultasi Bab IV dan V	9.
10.	14 April 2025	Konsultasi Revisi Bab IV dan V	10.
11.	5 Mei 2025	ACC Bab IV dan V	11.
12.	14 Mei 2025	Konsultasi Kajian Agama Bab IV	12.
13.	23 Mei 2025	ACC Kajian Agama Bab IV	13.
14.	27 Mei 2025	ACC Seminar Hasil	14.
15.	2 Juni 2025	Konsultasi Revisi Seminar Hasil	15.



**KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

16.	12 Juni 2025	ACC Sidang Skripsi	16.
17.	20 Juni 2025	ACC Keseluruhan	17.

Malang, 20 Juni 2025

Mengetahui,

Ketua Program Studi Matematika



Dr. Elly Susanti, M.Sc.

NIP. 19741129 200012 2 005