

Penelusuran Heuristik untuk Menyelesaikan Suatu Masalah Rute Terpendek Wilayah Oblast Moskwa di Negara Rusia Dengan Metode A* (A Star)

Mata Kuliah Kecerdasan Buatan
Oleh Dosen Finsa Nurpandi M. T.



Disusun oleh :
Mikayla Arzetha Susanto (5520123091)

**Fakultas Teknik
Jurusan Teknik Informatika
Universitas Suryakancana
2024/2025**

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya, Saya dapat menyelesaikan makalah ini dengan judul Penelusuran Heuristik untuk Menyelesaikan Suatu Masalah Rute Terpendek Wilayah Oblast Moskwa di Negara Rusia Dengan Metode A* (A Star). Makalah ini disusun sebagai bagian dari tugas mata kuliah Kecerdasan Buatan untuk memahami konsep dasar algoritma pencarian jalur serta penerapannya dalam kehidupan nyata.

Saya menyadari bahwa makalah ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun dari pembaca sangat diharapkan untuk perbaikan di masa mendatang. Semoga makalah ini dapat memberikan manfaat dan wawasan baru bagi pembaca, khususnya dalam memahami algoritma dan penerapannya.

Akhir kata, saya mengucapkan terima kasih kepada semua pihak yang telah membantu dalam penyusunan makalah ini, terutama kepada dosen pengampu mata kuliah serta rekan-rekan yang telah memberikan dukungan dan masukan berharga.

DAFTAR ISI

BAB I PENDAHULUAN	4
A. LATAR BELAKANG.....	4
B. TUJUAN	4
C. RUMUSAN MASALAH.....	4
BAB II PEMBAHASAN	5
A. PENGERTIAN PENCARIAN DAN FUNGSI HEURISTIK	5
B. PERHITUNGAN HEURISTIK EQUIRECTANGULAR APPROXIMATION.....	5
C. PERHITUNGAN METODE ALGORITMA A* (A STAR)	8
BAB III PENUTUP	11
A. KESIMPULAN.....	11
DAFTAR GAMBAR	12
DAFTAR PUSTAKA	13
ATRIBUSI KONTEN.....	14

BAB I

PENDAHULUAN

A. LATAR BELAKANG

Dunia yang semakin terhubung, penentuan jalur terpendek menjadi salah satu masalah fundamental yang sering dijumpai dalam berbagai bidang, seperti transportasi, logistik, jaringan komputer, dan sistem navigasi. Kemampuan untuk menemukan rute optimal tidak hanya menghemat waktu dan biaya, tetapi juga meningkatkan efisiensi operasional secara keseluruhan. Untuk menyelesaikan masalah ini algoritma pencarian jalurnya adalah algoritma A* (A Star).

Algoritma ini dikenal karena kemampuannya dalam menggabungkan keunggulan dari pencarian Heuristik dan pencarian Terbaik Utama (Best First Search), sehingga menghasilkan solusi yang optimal dan efisien. Makalah ini juga akan membahas bagaimana pemilihan fungsi heuristik yang tepat dapat memengaruhi performa algoritma, serta memberikan rekomendasi untuk pengembangan sistem navigasi yang lebih canggih.

B. TUJUAN

1. Dapat mengetahui pengertian pencarian Heuristik.
2. Dapat mengetahui dan membedakan jenis-jenis algoritma dalam fungsi Heuristik.
3. Dapat mengetahui algoritma Best First Search.
4. Dapat mengetahui cara penghitungan Heuristik untuk mengetahui rute terdekat.
5. Dapat menghitung dengan algoritma A* (A Star) untuk mencari rute terdekat.

C. RUMUSAN MASLAH

1. Bagaimana algoritma A* (A Star) dapat digunakan untuk menentukan jalur terpendek dalam berbagai bidang seperti transportasi, logistik, jaringan komputer, dan sistem navigasi?
2. Bagaimana pemilihan fungsi heuristik yang tepat memengaruhi performa dan efisiensi algoritma A dalam menemukan jalur optimal?
3. Apa kelebihan dan keterbatasan algoritma A dalam menyelesaikan masalah pencarian jalur terpendek dibandingkan dengan algoritma lain?

4. Bagaimana implementasi algoritma A dapat dioptimalkan untuk menghemat waktu, biaya, dan meningkatkan efisiensi operasional dalam berbagai aplikasi?

BAB II

PEMBAHASAN

A. PENGERTIAN PENCARIAN DAN FUNGSI HEURISTIK

Pencarian Heuristik (Heuristic searching) adalah sebuah metode dalam ilmu komputer dan kecerdasan buatan (AI) yang digunakan untuk mencari solusi dari masalah dengan cara yang lebih efisien berdasarkan pengetahuan yang ada meskipun memungkinkan solusinya tidak selalu memberikan solusi yang optimal. Secara umum, dalam pencarian Heuristik, kita menggunakan fungsi heuristik untuk memberikan penilaian atau perkiraan terhadap sejauh mana suatu langkah atau keputusan akan membawa kita lebih dekat ke tujuan. Pencarian Heuristik terdapat beberapa metode, sebagai berikut;

1. Hill Climbing

Metode Pencarian Heuristik Hill Climbing adalah salah satu teknik optimasi lokal yang digunakan untuk menemukan solusi terbaik (maksimum atau minimum) dari suatu masalah.

2. Best First Search

Best-First Search adalah algoritma pencarian heuristik yang efektif untuk menemukan solusi optimal dengan memanfaatkan fungsi heuristik.

3. Greedy Best First Search

Greedy Best-First Search adalah algoritma pencarian yang efektif untuk masalah di mana kecepatan lebih penting daripada optimalitas solusi. Namun, harus mempertimbangkan sifat masalah dan kualitas fungsi heuristik yang digunakan.

Fungsi Heuristik adalah suatu metode yang digunakan untuk memperkirakan seberapa dekat suatu titik atau keadaan dengan tujuan akhir, berdasarkan informasi yang ada, meskipun tidak selalu memberikan solusi yang optimal. Fungsi heuristik ini bertujuan untuk mempercepat pencarian dengan memilih jalur yang dianggap lebih menjanjikan berdasarkan perkiraan tersebut.

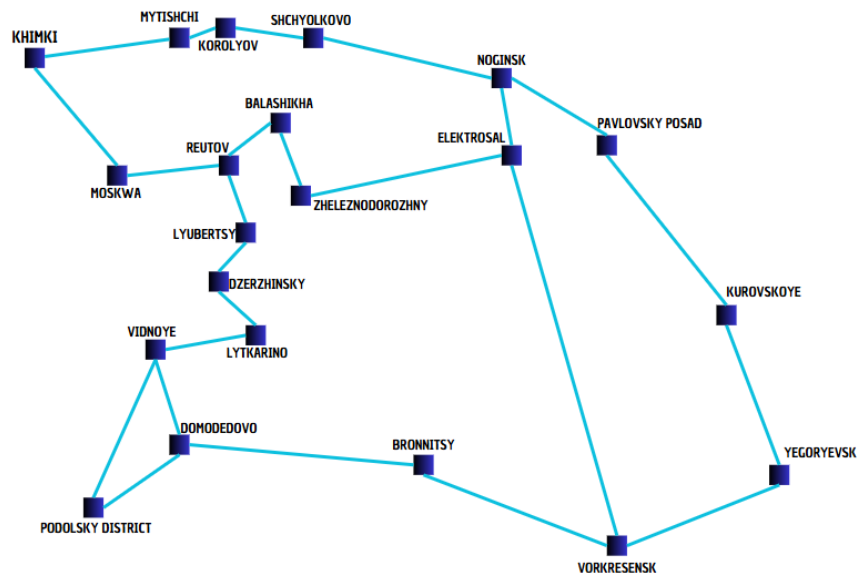
Ada beberapa cara menghitung dalam fungsi Heuristik, berikut beberapa contohnya;

1. Manhattan Distance
2. Euclidean Distance
3. Haversine Formula
4. Equirectangular Approximation

B. PERHITUNGAN HEURISTIK EQUIRECTANGULAR APPROXIMATION

Equirectangular Approximation adalah salah satu metode untuk menghitung fungsi heuristik yang digunakan dalam algoritma pencarian. Metode ini sering digunakan untuk mengestimasi jarak antara dua titik di permukaan bumi atau permukaan bola lainnya dengan cara yang sederhana dan cepat. Meskipun tidak seakurat metode lain seperti Haversine, Equirectangular Approximation cukup efektif untuk banyak aplikasi, terutama ketika performa komputasi lebih penting daripada presisi absolut.

Dalam studi kasus yang saya ambil untuk menyelesaikan masalah pencarian rute terpendek di wilayah Oblast Moskwa, Rusia, saya akan melakukan perhitungan menggunakan fungsi heuristik Equirectangular Approximation dengan melibatkan 21 kota beserta latitude dan longitude masing-masing kota. Saya memilih kota Vorkresensk sebagai node tujuan atau goals.



Gambar 2. 1 Node dari Wilayah Oblast Moskwa di Negara Rusia

a. Tabel Nama-Nama Kota beserta Latitude dan Longitude

No	Nama Kota	Latitude (N°)	Longitude (E°)
1	Moskwa (Moscow)	55.7558	37.6173
2	Khimki	55.8970	37.4297

3	Mytishchi	55.9167	37.7333
4	Korolyov	55.9167	37.8167
5	Shchyolkovo	55.9167	38.0000
6	Balashikha	55.8000	37.9500
7	Lytkarino	55.6833	37.8833
8	Podolsky District	55.4317	37.5450
9	Domodedovo	55.4333	37.7500
10	Vidnoye	55.5500	37.7000
11	Dzerzhinsky	55.6333	37.8500
12	Bronnitsy	55.4225	38.2619
13	Reutov	55.7667	37.8667
14	Zheleznodorozhny	55.7500	38.0167
15	Noginsk	55.8500	38.4333
16	Elektrostal	55.8000	38.4500
17	Pavlovsky Posad	55.7833	38.6500
18	Lyubertsy	55.6778	37.8933
19	Vorkresensk	55.3175	38.6528
20	Kurovskoye	55.5833	38.9167
21	Yegoryevsk	55.3833	39.0333

Equirectangular Approximation memiliki rumus sendiri dengan melibatkan latitude dan longitude, berikut rumusnya;

$$X = \Delta lng \cdot \cos \frac{(lat_1 + lat_2)}{2}$$

$$Y = \Delta lat$$

$$d = R \cdot \sqrt{X^2 + Y^2}$$

Gambar 2. 2 Rumus Menghitung Equirectangular Approximation

Dimana, Δlng bisa dicari dengan lng_2 (Longitude 2) dikurangi lng_1 (Longitude 1) lng_2 adalah longitude node tujuan. Disini saya mengambil kota Vorkresensk sebagai node tujuan. Dan lng_1 adalah longitude dari node asal atau bisa juga disebut sebagai kota asal. Begitu juga dengan Δlat , bisa dicari dengan lat_2 dikurangi lat_1 .

Untuk mencari lat_1 atau 2 dan lng_1 atau 2, bisa dengan rumus;

$$lat = \text{Latitude} \times 1 \text{ derajat} = \text{Latitude} \times 0.0174532925$$

$$lng = \text{Longitude} \times 1 \text{ derajat} = \text{Longitude} \times 0.0174532925$$

$$\text{Dan, } d = h(n) \text{ dimana } R = \text{Radius} = 6.371 \text{ km}$$

Dengan begitu bisa didapatkan nominal $d = h(n)$ setiap node kota-kota lain menuju node tujuan kota Vorkresensk.

b. Tabel Nama-Nama Kota beserta nilai hasil $d = h(n)$

No	Nama Kota	$d = h(n)$
1	Moskwa (Moscow)	81.62
2	Khimki	100.31
3	Mytishchi	86.59
4	Korolyov	84.79
5	Shchyolkovo	98.73
6	Balashikha	69.48
7	Lytkarino	63.53
8	Podolsky District	71.38
9	Domodedovo	101.45
10	Vidnoye	109.54
11	Dzerzhinsky	95.81
12	Bronnitsy	45.59
13	Reutov	101.29
14	Zheleznodorozhny	85.69
15	Noginsk	84.19
16	Elektrostal	58.22
17	Pavlovsky Posad	52.27
18	Lyubertsy	84.72
19	Vorkresensk	0
20	Kurovskoye	30.95
21	Yegoryevsk	42.04

Karena telah diketahui nilai $h(n)$ = Biaya perkiraan (heuristik) dari titik saat ini (n) ke titik tujuan. Maka, perhitungan jarak terdekat dilanjutkan dengan metode A* (A star).

C. PERHITUNGAN METODE ALGORITMA A* (A STAR)

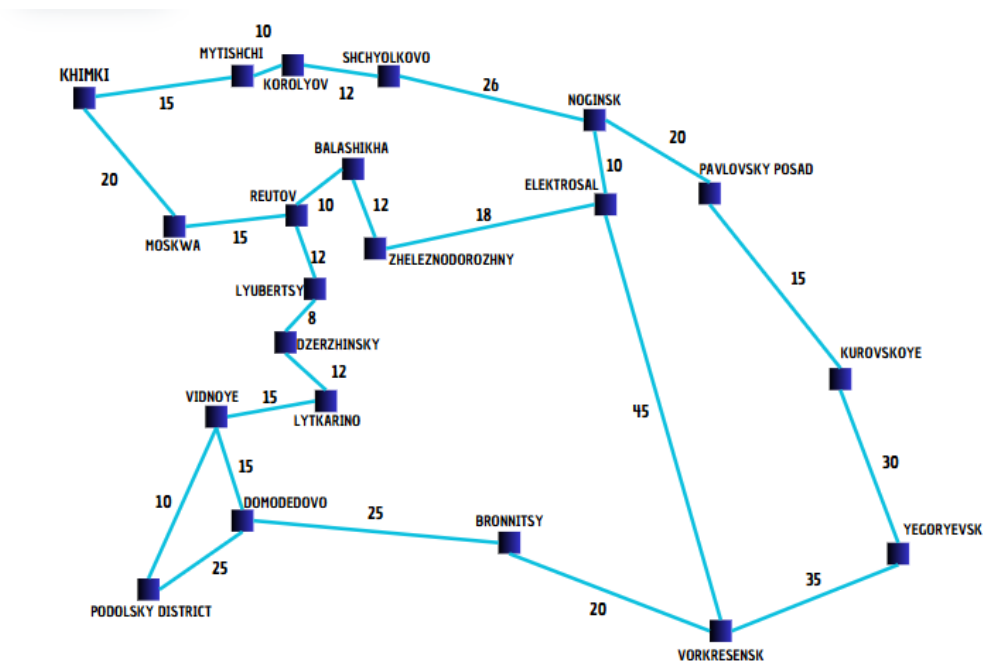
Algoritma A* (A Star) adalah algoritma pencarian jalur yang banyak digunakan dalam bidang ilmu komputer untuk menemukan jalur terpendek antara dua titik (node) dalam sebuah graf. Algoritma A* merupakan algoritma Best First Search yang menggabungkan Uniform Cost Search dan Greedy Best-First Search. Biaya yang diperhitungkan didapat dari biaya sebenarnya ditambahkan dengan biaya perkiraan, yang ditulis dalam notasi matematika sebagai berikut;

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

Keterangan;

- $g(n)$: Biaya aktual dari titik awal ke titik saat ini (node n).
- $h(n)$: Biaya perkiraan (heuristik) dari titik saat ini (n) ke titik tujuan.
- $f(n)$: Total biaya perkiraan, nilai ini digunakan untuk menentukan urutan ekspansi node.

Dalam studi kasus ini saya menempatkan kota **Khimki** sebagai **node asal** atau kota asal dan kota **Vorkresensk** sebagai **node tujuan** atau kota asal.



Gambar 2. 3 Gambar Node dalam Wilayah Oblast Moskwa dengan $g(n)$

- Langkah 1
Bangkitkan suksesor kota Khimki, yaitu; Mytishchi dan Moskwa. Semuanya berada di OPEN maupun CLOSED. Pindahkan Khimki ke CLOSED dan OPEN Mytishchi dan Moskwa.
- Langkah 2
Menghitung biaya terkecil Mytishchi dan Moskwa. Dengan rumus $f(n)$;
 - Khimki ke Mytishchi
$$f(n) = 15 + 86,59 = 101,59$$
 - Khimki ke Moskwa
$$f(n) = 20 + 81,62 = 101,62$$Mytishchi memiliki biaya terkecil dengan hasil 101,59 terpilih sebagai best-node dan dipindahkan ke CLOSED. Dibangkitkan Korolyov dan dimasukan ke OPEN. Dengan begitu OPEN = [Moskwa, Korolyov] dan CLOSED = [Khimki, Mytishchi].
- Langkah 3

Lanjut menghitung biaya terkecil Mytishchi ke Korolyov dengan rumus $f(n)$.

- a. Mytishchi ke Korolyov

$$f(n) = 25 + 84,79 = 109,79$$

Dibangkitkan Shchyolkovo dan dimasukkan ke OPEN = [Moskwa, Shchyolkovo] dan CLOSED = [Khimki, Mytishchi, Korolyov].

4. Langkah 4

Menghitung biaya terkecil Korolyov ke Shchyolkovo dengan rumus $f(n)$.

- a. Korolyov ke Shchyolkovo

$$f(n) = 37 + 98,73 = 135,73$$

Dibangkitkan Noginsk dan dimasukkan ke OPEN = [Moskwa, Noginsk] dan CLOSED = [Khimki, Mytishchi, Korolyov, Shchyolkovo].

5. Langkah 5

Menghitung biaya terkecil lagi Shchyolkovo ke Noginsk dengan rumus $f(n)$.

- a. Shchyolkovo ke Noginsk

$$f(n) = 63 + 84,19 = 147,19$$

Dibangkitkan Elektrosal dan Pavlovsky Posad dimasukkan ke OPEN = [Moskwa, Elektrosal, Pavlovsky Posad] dan CLOSED = [Khimki, Mytishchi, Korolyov, Shchyolkovo, Noginsk].

6. Langkah 6

Menghitung biaya terkecil Elektrosal dan Pavlovsky Posad dengan rumus $f(n)$.

- a. Noginsk ke Elektrosal

$$f(n) = 73 + 58,22 = 131,22$$

- b. Noginsk ke Pavlovsky Posad

$$f(n) = 83 + 52,27 = 135,27$$

Elektrosal memiliki biaya terkecil dengan hasil 131,22 akan dimasukkan ke CLOSED. OPEN = [Moskwa, Pavlovsky Posad] dan CLOSED = [Khimki, Mytishchi, Korolyov, Shchyolkovo, Noginsk, Elektrosal].

7. Langkah 7

Karena node Elektrosal membangkitkan langsung ke node tujuan Vorkresensk dan node lain Zheleznodorozhny.

- a. Elektrosal ke Vorkresensk

$$f(n) = 118 + 0 = 118$$

- b. Elektrosal ke Zheleznodorozhny

$$f(n) = 91 + 85,69 = 176,69$$

Langsung menuju node tujuan Vorkresensk memiliki biaya yang lebih kecil, maka OPEN = [Moskwa, Pavlovsky Posad, Zheleznodorozhny] dan CLOSED = [Khimki, Mytishchi, Korolyov, Shchyolkovo, Noginsk, Elektrosal, Vorkresensk].

BAB III

PENUTUP

A. KESIMPULAN

Algoritma A* (A Star) telah terbukti sebagai solusi efektif dalam menyelesaikan masalah pencarian jalur terpendek di berbagai bidang, seperti transportasi, logistik, jaringan komputer, dan sistem navigasi. Dengan menggabungkan keunggulan pencarian heuristik dan pencarian Terbaik Utama (Best First Search), algoritma ini mampu menghasilkan solusi yang optimal dan efisien.

Dalam studi kasus yang saya ambil, dapat disimpulkan bahwa jalur dengan jarak terdekat dari **kota Khimki ke kota Vorkresensk** adalah **jalur = [Khimki – Mytishchi – Korolyov – Shchyolkovo – Noginsk – Elektrosal – Vorkresensk]**. Dengan jumlah jalur terdekat, **$f(n) = 118 + 0 = 118$** .

DAFTAR GAMBAR

Gambar 2. 1 Node dari Wilayah Oblast Moskwa di Negara Rusia.....	6
Gambar 2. 2 Rumus Menghitung Equirectangular Approximation	7
Gambar 2. 3 Gambar Node dalam Wilayah Oblast Moskwa dengan $g(n)$	9

DAFTAR PUSTAKA

- D. Tomlin, Geographic Information Systems and Cartographic Modeling. Englewood Cliffs, NJ, USA: Prentice Hall, 1990.
- B. P. McClain, Python for Geospatial Data Analysis. Sebastopol, CA, USA: O'Reilly Media, 2022.
- S. Shekhar and S. Chawla, Spatial Computing: A Comprehensive Approach. San Francisco, CA, USA: Morgan Kaufmann, 2003.
- Wikipedia. (2023). Haversine formula. Diakses pada 13 Maret 2025, dari https://en.wikipedia.org/wiki/Haversine_formula
- Movable Type Scripts. (2023). Calculate distance and bearing between two latitude/longitude points. Diakses pada 13 Maret 2025, dari <https://www.movable-type.co.uk/scripts/latlong.html>
- Google Maps. (2023). Directions from Moscow to Reutov. Diakses pada 14 Maret 2025, dari <https://www.google.com/maps>

OpenStreetMap. (2023). Map data for Shchyolkovo and Noginsk. Diakses pada 15 Maret 2025, dari <https://www.openstreetmap.org>

Shekhar, S., & Chawla, S. (2003). Spatial Computing. Diakses pada 15 Maret 2025, dari <https://www.spatial-computing.com>

ATRIBUSI KONTEN

Saya mengakui telah menggunakan DeepSeek-V3 dari perusahaan DeepSeek dalam proses riset dan penulisan makalah ini. Aplikasi DeepSeek berfungsi sebagai alat bantu untuk meningkatkan kualitas tulisan dan menganalisa data longitude dan latitude kota-kota dari wilayah studi kasus makalah ini. Penting untuk diketahui bahwa konten yang dihasilkan oleh DeepSeek tidak disalin secara verbatim, melainkan telah direview secara keseluruhan, diedit, dan dikurasi oleh saya, sebagai penulis untuk memastikan akurasi, otentisitas, dan integritas dari informasi di dalamnya. Penilaian dan keputusan individu berkontribusi besar dalam menginterpretasi dan memvalidasi keluaran AI ini. Oleh sebab itu, hasil akhir yang disampaikan ini merupakan hasil kolaborasi antara manusia dan AI.