

ANALISIS PERBANDINGAN GREEDY SHORTEST PATH MASTERDAN SHORTEST PATH

Laporan

Ditujukan Sebagai Salah Satu Syarat
Untuk Memenuhi Tugas Mata Kuliah
Strategi Algoritma

Oleh

1214053 Raul Mahya Komaran

1214078 Maulana Imanulhaq Nurdiana



ULBI

Universitas Logistik & Bisnis Internasional

**PROGRAM DIPLOMA IV TEKNIK INFORMATIKA
UNIVERSITAS LOGISTIK DAN BISNIS INTERNASIONAL
BANDUNG
2023**

SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIARISME

Saya yang bertandatangan di bawah ini:

Nama : RAUL MAHYA KOMARAN

NPM 1214053

Program Studi : D-IV TEKNIK INFORMATIKA

Judul : ANALISIS PERBANDINGAN GREEDY SHORTEST PATH MASTER
DAN SHORTEST PATH

Dengan ini menyatakan bahwa:

1. Tugas harian ini adalah murni hasil gagasan, rumusan, dan penelitian saya sendiri tanpa bantuan pihak lain, kecuali arahan pembimbing.
2. Dalam tugas harian ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
3. Apabila dikemudian hari terbukti melakukan plagiat dan penyimpangan-penyimpangan lain dalam pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku.

Bandung,

Yang membuat pernyataan

RAUL MAHYA KOMARAN

NPM 1.21.4.053

SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIARISME

Saya yang bertandatangan di bawah ini:

Nama : MAULANA IMANULHAQ NURDIANA

NPM : 1214078

Program Studi : D-IV TEKNIK INFORMATIKA

Judul : ANALISIS PERBANDINGAN GREEDY SHORTEST PATH MASTER
DAN SHORTEST PATH

Dengan ini menyatakan bahwa:

1. Tugas harian ini adalah murni hasil gagasan, rumusan, dan penelitian saya sendiri tanpa bantuan pihak lain, kecuali arahan pembimbing.
2. Dalam tugas harian ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
3. Apabila dikemudian hari terbukti melakukan plagiat dan penyimpang-penyimpangan lain dalam pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku.

Bandung,
Yang membuat pernyataan

MAULANA IMANULHAQ NURDIANA
NPM 1.21.4.078

ABSTRAK

Optimasi merupakan metode pemecah masalah meminimalkan atau memaksimalkan untuk meningkatkan produktivitas kerja, Pencarian lintasan terpendek menjadi hal yang diperlukan pencarian jalur terpendek digunakan misalkannya seorang pengendara mobil, perutean network computer algoritma Greedy, program dinamis. Algoritma exhaustive search mencari lintasan terpendek dengan mengenumerasi semua rangkaian keputusan yang mungkin dari lintasan yang ada dan memilih rangkaian keputusan yang terbaik. Algoritma Dijkstra dengan menggunakan prinsip Greedy mencari lintasan terpendek dari satu simpul ke semua simpul lain yang terhubung. Dari banyak metode penyelesaian masalah lintasan terpendek pada graf, tentu saja banyak perbedaan meskipun tujuan yang diharapkan sama. Yaitu menemukan lintasan terpendek / minimum yang optimum. Meskipun tujuannya utamanya adalah mencari lintasan terpendek tetapi ada juga tujuan lainnya yaitu menemukan lintasan terpendek yang optimum dengan waktu pencarian yang minimum juga. Tetapi sangatlah sulit untuk mendapatkan kedua hal tersebut. [1]

Pencarian lintasan terpendek yang optimum tentu saja memakan waktu yang maksimum pula. Karena itulah banyak cara yang ditawarkan untuk pencarian lintasan terpendek pada graf dengan kelebihan dan kekurangan masing – masing.

Kata Kunci : *Greedy, Algoritma, Lintasan*

KATA PENGANTAR

Segala puji dan syukur kami panjatkan kehadiran Allah SWT karena atas berkat dan rahmatnya kami diberikan kesehatan jasmani dan rohani sehingga penulis dapat menyelesaikan Laporan Tugas 1 yang berjudul “ANALISIS PERBANDINGAN GREEDY SHORT PATH MASTER DAN SHORTEST PATH”.

Sebelumnya kami juga menyadari banyak kekurangan dalam penulisan laporan ini. Maka dari itu kami mengharapkan kritik dan saran yang baik dari para pembaca demi kesempurnaan laporan proyek ini. Untuk itu kami mengucapkan terima kasih kepada:

1. Dr. Ir. Agus Purnomo, M.T. selaku Direktur Universitas Logistik Bisnis Internasional .
2. Kedua orang tua dan keluarga kami yang telah mendorong dan memberi semangat kepada kami .
3. Yang terhormat Roni Andarsyah, ST., M.Kom., SFPC selaku Ketua Program Studi DIV Teknik Informatika
4. Roni Habibi, S.Kom., M.T., SFPC selaku dosen mata kuliah Strategi Algoritma.

Kami berharap semoga apa yang telah penulis kembangkan pada Laporan Tugas 1 ini dapat bermanfaat bagi pihak – pihak yang berkepentingan.

Bandung, February 2023

Penulis

DAFTAR ISI

SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIARISME	2
SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIARISME	3
BAB I.....	7
HASIL ANALISIS.....	7
APA ITU ALGORITMA GREEDY	7
APA ITU SHORTEST PATH.....	7
BAB II.....	9
PENGUJIAN.....	9
PENGETESAN APLIKASI.....	9
1. Algoritma greedy shortest path	9
2. Shortest-Path-AI.....	10
BAB III	12
PERBANDINGAN	12
BAB IV	14
DAFTAR PUSTAKA	14

BAB I HASIL ANALISIS

APA ITU ALGORITMA GREEDY

Algoritma Greedy adalah algoritma yang memecahkan masalah dengan mengambil keputusan yang optimal pada setiap langkah, berdasarkan informasi yang tersedia pada saat itu saja. Pendekatan ini sering kali digunakan untuk memecahkan masalah optimasi, di mana kita ingin mencari solusi yang memaksimalkan atau meminimalkan suatu fungsi tujuan.

Berikut adalah langkah-langkah umum untuk algoritma Greedy:

1. Tentukan fungsi tujuan yang ingin dioptimalkan.
2. Tentukan kriteria pengambilan keputusan yang optimal pada setiap langkah.
3. Buat daftar pilihan yang tersedia pada setiap langkah.
4. Pilih opsi terbaik yang memenuhi kriteria pengambilan keputusan pada langkah saat ini.
5. Perbarui daftar pilihan dan evaluasi kembali keputusan yang diambil.
6. Ulangi langkah 4 dan 5 sampai solusi optimal ditemukan.

Algoritma Greedy sering kali digunakan dalam kombinasi dengan algoritma lain, seperti Dynamic Programming atau Backtracking, untuk menemukan solusi optimal.[1]

APA ITU SHORTEST PATH

Shortest Path adalah istilah yang digunakan dalam teori graf dan matematika untuk merujuk pada jalur terpendek antara dua titik di dalam sebuah graf. Dalam graf, setiap titik atau simpul biasanya dihubungkan oleh satu atau lebih sisi atau tepi, yang menunjukkan hubungan antara dua titik tersebut.

Untuk mencari jalur terpendek antara dua titik di dalam graf, algoritma Shortest Path digunakan. Algoritma ini mencari jalur terpendek dari titik awal ke titik akhir dengan mempertimbangkan bobot pada setiap sisi atau tepi. Bobot ini dapat merepresentasikan biaya, jarak, waktu, atau faktor lain yang ingin dioptimalkan.

Ada beberapa algoritma yang digunakan untuk menemukan jalur terpendek antara dua titik dalam graf, seperti Dijkstra's algorithm, Bellman-Ford algorithm, Floyd-Warshall algorithm, dan A* algorithm. Setiap algoritma memiliki kelebihan dan kekurangan tergantung pada sifat dan ukuran graf.

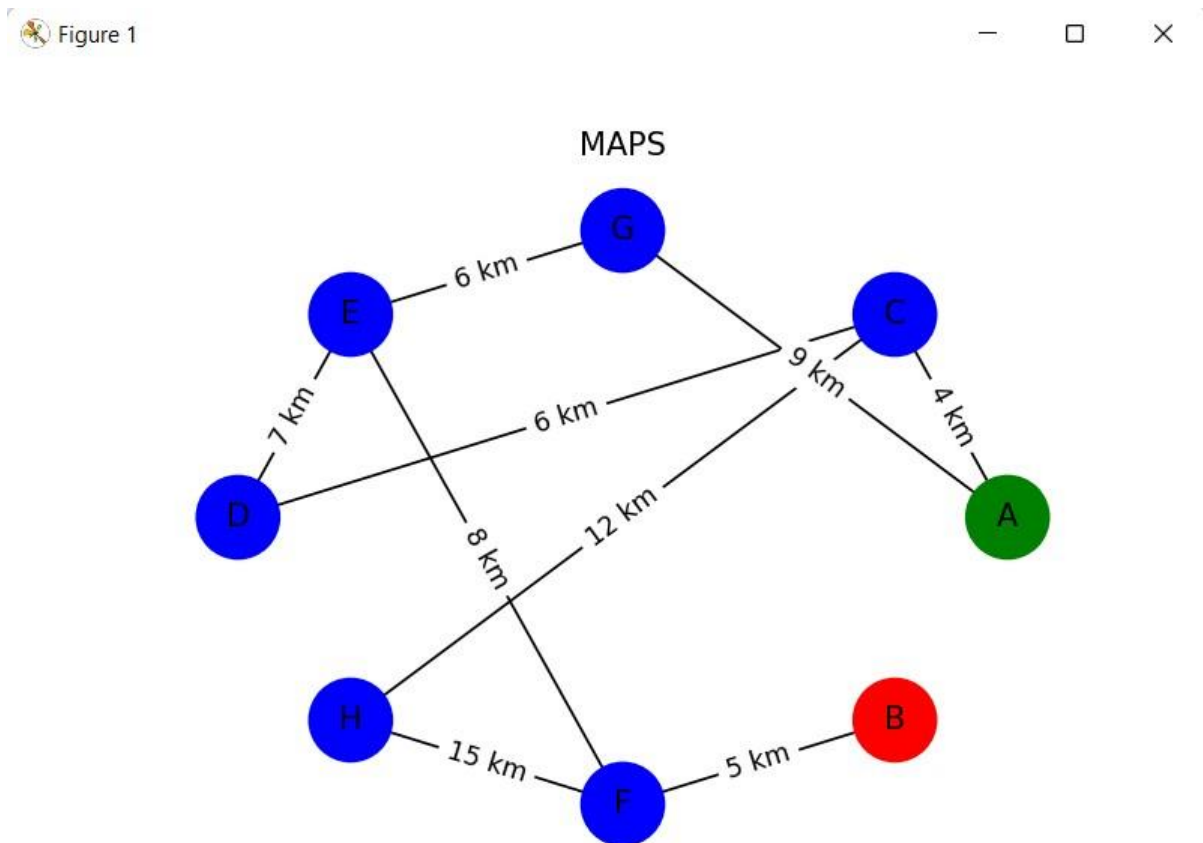
Aplikasi dari Shortest Path sangat luas, seperti dalam sistem navigasi, pengiriman paket, perencanaan jaringan, analisis sosial, dan banyak lagi. Mencari jalur terpendek antara dua titik dapat membantu kita meminimalkan biaya, waktu, atau upaya yang diperlukan untuk mencapai tujuan.

BAB II PENGUJIAN

PENGETESAN APLIKASI

1. Algoritma greedy shortest path

Mencari lintasan terpendek dengan exhaustive search yaitu dengan mengenumerasi setiap lintasan yang mungkin dengan cara yang sistematis. Dari setiap kemungkinan tersebut dievaluasi satu persatu, selanjutnya bandingkan setiap lintasan yang telah dievaluasi, lintasan yang memberikan nilai terkecil merupakan lintasan terpendek yang kita cari. [2]



Misalkan akan dicari lintasan terpendek dari simpul A ke simpul B. Banyaknya kemungkinan lintasan dari simpul A ke simpul B ada 4 kemungkinan lintasan. Kemudian enumerasi dan evaluasi setiap kemungkinan tersebut.

NO	LINTASAN	JARAK	KETERANGAN
1	a-c-d-e-f-b	30	-
2	a-c-h-f-b	36	-
3	a-g-e-f-b	28	solusi
4	a-g-e-d-c-h-f-b	60	-

Algoritma ini akan memberikan solusi yang paling optimal untuk semua kasus graf dengan bobot yang tidak negatif. Akan tetapi, untuk jumlah simpul yang banyak algoritma ini tidak mangkus karena membutuhkan waktu komputasi yang lama.

2. Shortest-Path-AI

Repositori ini berisi file main.py yang membuat grid acak atau mengambil input grid dan menemukan jalur terpendek menggunakan algoritma A* dan Theta*.

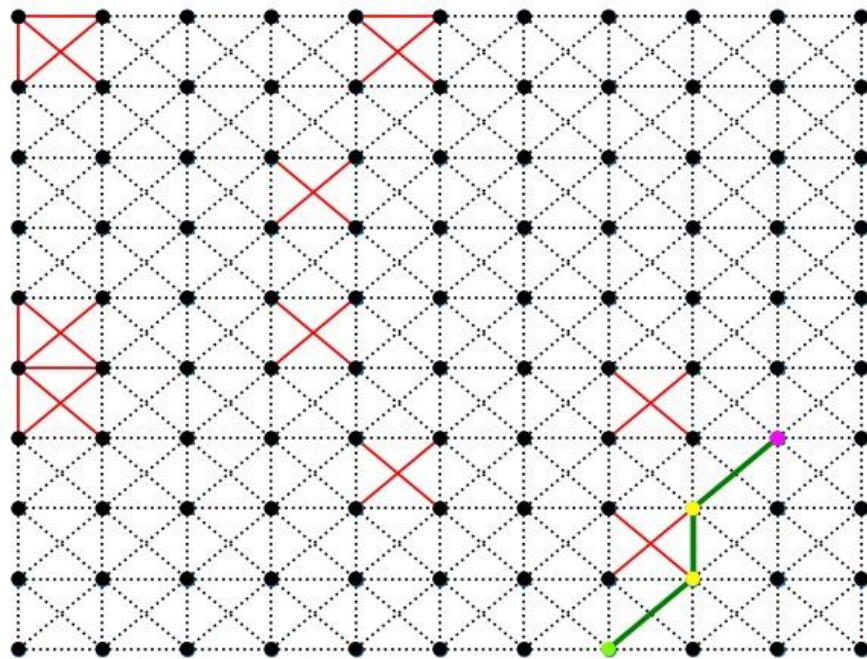
Keuntungan

Keuntungan dari kedua algoritma tersebut dibandingkan dengan menggunakan dykstras atau pencarian pertama yang luas adalah bahwa algoritma tersebut menggunakan memori yang jauh lebih sedikit dengan kemungkinan kesalahan yang kecil. Alih-alih melintasi seluruh ruang grid, algoritme secara efisien menggunakan heuristiknya dan hanya menjelajahi tetangga terdekat dari node yang dipilih.

A*

A* huristic menjelajahi titik-titik di sekitar titik saat ini dan memilih titik terdekat berdasarkan titik ke simpul awal, dan jarak ke simpul akhir.

Figure 1



x=13.06 y=-10.20

No	LINTASAN X	LINTASAN Y	HASIL
1	[8, 10]	[10, 7]	[[8, 10], [9, 9], [9, 8], [10, 7]]

Algoritma ini akan memberikan solusi yang paling optimal untuk semua kasus graf dengan bobot yang tidak negatif. Akan tetapi, untuk jumlah simpul yang banyak algoritma ini tidak mangkus karena membutuhkan waktu komputasi yang lama.

BAB III

PERBANDINGAN

Raul Mahya :

Greedy Shortest Path dan Shortest Path keduanya adalah algoritma untuk menemukan jalur terpendek antara dua titik di dalam graf. Namun, ada perbedaan penting antara keduanya.

Shortest Path adalah algoritma yang secara tepat mencari jalur terpendek antara dua titik di dalam graf, dengan menggunakan teknik seperti Dijkstra's algorithm atau Bellman-Ford algorithm. Algoritma Shortest Path memberikan jaminan bahwa solusi yang diberikan adalah jalur terpendek yang mungkin. Namun, algoritma ini memerlukan kompleksitas waktu yang relatif besar.

Di sisi lain, Greedy Shortest Path adalah algoritma yang bekerja dengan mengambil keputusan lokal terbaik pada setiap langkah dalam perjalanan menuju tujuan. Algoritma ini mencoba memilih jalur terpendek yang tersedia pada setiap saat, tanpa mempertimbangkan bagaimana pilihan tersebut akan mempengaruhi solusi akhir. Karena pendekatan ini, Greedy Shortest Path sering kali lebih cepat daripada Shortest Path, tetapi tidak memberikan jaminan bahwa solusi yang diberikan adalah jalur terpendek yang mungkin.

Dengan kata lain, perbedaan antara Greedy Shortest Path dan Shortest Path adalah bahwa Greedy Shortest Path mengambil keputusan yang hanya berdasarkan informasi lokal dan dapat memberikan solusi yang tidak optimal, sementara Shortest Path mencari jalur terpendek yang mungkin dengan kompleksitas waktu yang lebih besar. Pilihan antara kedua algoritma tergantung pada kebutuhan aplikasi dan apakah solusi yang tepat diperlukan atau solusi yang cepat diperlukan.

Maulana Imanulhaq :

Greedy Shortest Path dan Shortest Path adalah dua algoritma pencarian jalan terpendek yang digunakan dalam graf. Berikut adalah perbedaan antara keduanya:

Pendekatan: Greedy Shortest Path menggunakan pendekatan egois untuk mencari jalan terpendek dari suatu titik ke titik lainnya. Algoritma ini hanya mempertimbangkan jalur terpendek saat ini, tanpa memperhatikan jalur alternatif yang mungkin lebih pendek. Sedangkan, Shortest Path menggunakan pendekatan yang lebih sistematis untuk mencari jalan terpendek dengan mempertimbangkan semua jalur yang mungkin dari titik asal ke titik tujuan.

Kompleksitas waktu: Algoritma Greedy Shortest Path relatif sederhana dan memiliki kompleksitas waktu yang lebih rendah dibandingkan dengan algoritma Shortest Path. Namun, kualitas solusi yang dihasilkan oleh algoritma ini tidak selalu optimal, sedangkan algoritma Shortest Path menghasilkan solusi optimal dalam hal jarak terpendek.

Ketersediaan informasi: Greedy Shortest Path hanya mempertimbangkan informasi tentang node yang terhubung langsung ke node saat ini. Sementara itu, Shortest Path mempertimbangkan semua informasi tentang graf, seperti bobot setiap edge, untuk menemukan jarak terpendek.

Tujuan: Tujuan Greedy Shortest Path adalah untuk mencari jalan terpendek yang meminimalkan jarak dari titik awal ke titik tujuan. Sedangkan, Shortest Path mencari jalan terpendek yang meminimalkan jumlah bobot (atau jarak) dari titik awal ke semua titik lain di graf.

Kecepatan konvergensi: Greedy Shortest Path menghasilkan solusi dengan cepat, tetapi tidak selalu optimal. Sedangkan Shortest Path membutuhkan waktu lebih lama untuk menghasilkan solusi, tetapi solusi yang dihasilkan lebih akurat.

Jadi, sementara Greedy Shortest Path dan Shortest Path keduanya digunakan untuk mencari jalan terpendek, pendekatan dan kualitas solusi yang dihasilkan oleh keduanya berbeda.

BAB IV

DAFTAR PUSTAKA

- [1] N. W. Y. Agus Ambarwari, "PENERAPAN ALGORITMA GREEDY PADA PERMASALAHAN KNAPSACK UNTUK OPTIMASI PENGANGKUTAN PETI KEMAS," *See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/290797775>*, January 2016.
- [2] E. Romelta, "Metode Pencarian Lintasan Terpendek Dalam Graf," *JURNAL ILMU KOMPUTER DAN TEKNOLOGI INFORMASI*,, vol. III, 2003.
- [3] G. S. D. T. Indra Fajar, "ALGORITMA MENCARI LINTASAN TERPENDEK".