

Implementasi Graf dalam Sistem Navigasi Jalur Kendaraan

Irgiansyah Mondo - 13521167
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13521167@itb.ac.id

Abstract— Transportasi sudah menjadi bagian dari kehidupan manusia sejak masa terdahulu. Perkembangan teknologi yang pesat membuat transportasi di zaman sekarang semakin modern. Banyaknya konsumen mempengaruhi kepadatan jalur transportasi sehingga para pengendara membutuhkan suatu pedoman untuk mencari jalur yang tepat untuk ditempuh. Maka dari itu, salah satu solusi yang bisa ditawarkan adalah dengan penggunaan sistem navigasi GPS (yang diimplementasikan dengan teori graf.

Keyword : Teori graf, sistem navigasi.

I. INTRODUCTION

GPS (Global Positioning System) adalah sistem yang bertujuan untuk mengetahui posisi dengan sistem navigasi. Sistem ini pertama kali ditemukan oleh Amerika Serikat yang didesain untuk efektifitas strategi dalam militer berupa informasi mengenai waktu, posisi dan kecepatan, kepada banyak orang secara kontinyu di seluruh dunia tanpa bergantung pada waktu dan cuaca.

Pada saat ini, GPS sudah banyak digunakan diberbagai platform digital, di Indonesia sendiri telah banyak digunakan para pengendara untuk menerima informasi tentang posisi dan lain sebagainya. Dibanding dengan sistem penentu posisi lainnya, GPS mempunyai banyak kelebihan dan keuntungan dalam operasional dan kualitas.

II. LANDASAN TEORI

A. Definisi Graf

Graf digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Graf $G = (V, E)$, yang dalam hal ini :

- V = himpunan tidak-kosong dari simpul simpul (vertices) = $\{v_1, v_2, \dots, v_n\}$
- E = himpunan sisi (edges) yang menghubungkan sepasang simpul = $\{e_1, e_2, \dots, e_n\}$

B. Jenis – jenis Graf

Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, maka graf digolongkan menjadi dua jenis yaitu graf sederhana dan graf tak-sederhana. Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi ganda. Sedangkan

graf tak-sederhana adalah graf yang mengandung sisi ganda atau gelang. Graf tak-sederhana dibedakan lagi menjadi dua yaitu graf ganda yang mengandung sisi ganda dan graf semu yang mengandung sisi gelang. Berdasarkan orientasi arah pada sisi, maka secara umum graf dibedakan atas dua jenis yaitu graf tak-berarah dan graf berarah. Graf tak-berarah adalah graf yang sisinya tidak mempunyai orientasi arah. Sedangkan graf berarah adalah graf yang setiap sisinya diberikan orientasi arah.

C. Terminologi Graf

1. Ketetangaan (adjacent) Dua buah simpul dikatakan bertetangga bila keduanya terhubung langsung.
2. Bersisian (incidency) Untuk sembarang sisi $e = (v_i, v_j)$ dikatakan bahwa e bersisian dengan simpul (vertex) v_i dan v_j .
3. Derajat (degree) Derajat suatu simpul adalah jumlah sisi yang bersisian dengan simpul tersebut. Notasinya adalah $d(v)$
4. Lintasan (path) Lintasan yang panjangnya n dari simpul awal v_0 ke simpul tujuan v_n di dalam sebuah graf adalah barisan berselang-seling simpul-simpul dan sisi-sisi yang berbentuk $v_0, e_1, v_1, e_2, v_2, \dots, e_n, v_n$.
5. Sirkuit (Circuit) Lintasan yang berawal dan berakhir di tempat (simpul) yang sama disebut sirkuit
6. Graf berbobot (weighted graph) Apabila sisi-sisi pada graph disertai juga dengan suatu (atau beberapa) harga yang menyatakan secara unik kondisi keterhubungan tersebut maka graph tersebut disebut graph berbobot.

D. Representasi Graf

1. Adjacency Matrix
Suatu matriks digunakan untuk menyatakan adjacency set setiap verteks dalam barisbarisnya. Nomor baris menyatakan nomor verteks adjacency berasal dan nomor kolom menunjukkan nomor verteks kemana arah adjacency. Elemen matriks $[x, y]$ berharga 1 bila terdapat sisi dari x ke y dan berharga 0 bila tidak ada.
2. Adjacency List
Mengingat bahwa rasio degree (atau outdegree pada digraph) rata-rata verteks terhadap jumlah verteks dalam sejumlah masalah adalah bisa sangat kecil maka representasi matriks tersebut akan berupa matriks sparse yaitu sebagian besarnya berisikan bilangan nol (atau bilangan ∞ pada weighted graph). Untuk

kepentingan efisiensi ruang maka tiap baris matriks tersebut digantikan list yang hanya berisikan verteks-verteks dalam adjacency set V_x dari setiap verteks x .

E. Lintasan dan Sirkuit Euler

Lintasan Euler adalah lintasan yang melalui masing-masing sisi di dalam graf secara tepat satu kali. Sirkuit Euler ialah sirkuit yang melewati masing –masing sisi tepat satu kali. Graf yang memiliki sirkuit Euler disebut dengan graf Euler. Graf yang memiliki lintasan euler dinamakan juga graf semi euler.

F. Graf Hamilton

Melalui tiap simpul di dalam graf tepat satu kali. Sirkuit Hamilton ialah sirkuit yang melalui tiap simpul di dalam graf tepat satu kali, kecuali simpul asal (sekaligus simpul akhir) yang dilalui dua kali. Graf yang memiliki sirkuit Hamilton dinamakan graf Hamilton, sedangkan graf yang hanya memiliki lintasan Hamilton disebut graf semi-Hamilton.

III. CARA KERJA GPS

A. Posisi dan Sistem Koordinat

Dalam menentukan posisi dan sistem koordinat dua dimensi atau tiga dimensi dengan menspesifikasi tiga parameter berikut, yaitu:

- Lokasi sistem koordinat,.
- Sumbu-sumbu koordinat.
- Besaran yang digunakan untuk mendefinisikan sistem koordinat.

Dalam Penentuan posisi titik, titik awal dari koordinat yang dipakai harusnya berlokasi di titik pusat berat objek (geosentrik), ataupun bisa juga di suatu titik di permukaan objek (toposentrik). Dalam menentukan objek, ada berbagai referensi yang bisa digunakan untuk menentukan objek koordinat. Jika yang digunakan bertujuan untuk suatu objek yang ada di bumi maka yang dipakai adalah orientasi yang terikat dengan bumi. Jika yang bertujuan untuk menganalisis posisi objek angkasa maka yang digunakan adalah orientasi sumbu koordinat yang terikat dengan langit. Untuk menentukan besaran jarak, biasanya dengan menggunakan sistem koordinat kartesian dan juga ada beberapa besaran yang bisa digunakan untuk menentukan besaran sudut dan jarak seperti sistem geodetic. Sistem penentuan posisi terbagi menjadi dua, yaitu CIS (Conventional Inertial System) dan CTS (Conventional Terrestrial System). CIS berguna untuk mendeskripsikan posisi dan satelit, sedangkan CTS berguna untuk menyatakan posisi titik dipermukaan bumi.

Sistem CIS bersifat geosentrik yang terikat dengan langit atau ESCF (Earth -Centered Earth-Fixed). Pada posisi GPS, suatu permukaan bumi diberikan koordinat kartesian tiga-dimensi (x,y,z) sebagai realisasi dari sistem CTS. Koordinat tersebut ditransformasikan menjadi koordinat geodesic. Hubungan matematis antara koordinat kartesian dan geodetic dapat

dituliskan sebagai berikut :

$$R = \frac{a}{\sqrt{1-e^2 \sin^2 x}}$$

$$a^2 = \frac{a^2 - b^2}{a^2}$$

Pada rumus diatas. R dan e adalah jari kelengkungan vertical dan aksentrisitas ellipsoid. Dimana a dan b adalah setengah sumbu Panjang dan setengan sumbu pendek dari ellipsoid referensi yang digunakan

B. Metode Penentuan Posisi

Penentuan posisi dilakukan secara terestris maupun ekstraterestris. Terdapat dua metode untuk menentukan posisi yaitu secara terestris dilakukan berdasarkan pengukuran dan pengamatan yang semuanya dilakukan di permukaan bumi. Sedangkan pada metode ekstraterestris, penentuan-penentuan posisi dilakukan dengan melakukan pengukuran dan pengamatan ke objek di langit. Ada beberapa metode atau sistem penentuan posisi secara ekstra-terestris yang telah dikenal selama ini. Yaitu : Astronomi geodesi , Fotografi satelit, SLR . Doppler dan GPS.

IV. NAVIGASI SISTEM

Sistem Navigasi GPS menyediakan berbagai fitur yang sangat berguna, misalnya dalam fitur real time yang setiap waktu dapat memudahkan posisi, kecepatan dan juga perilaku dari objek benda. Posisi benda pada GPS adalah yaitu penentuan suatu kemungkinan pada saat objek sedang bergerak dalam waktu sesingkat mungkin. Penerapan navigasi dengan sensor GPS sudah banyak digunakan di berbagai sector seperti darat,udara, laut dan angkasa. Dilaut sendiri digunakan untuk proses memandu pergerakan suatu wahana laut secara efektif dan efisien, sehingga proses yang dilaut dapat terlaksana sesuai dengan tujuannya digunakan untuk mengakomodir format data sehingga dapat diintegrasikan dengan peralatan elektronik. Didarat sendiri dikofigurasi dengan sistem electronic based board yang terintegrasi dengan alat elektronik berupa computer, PC, serta menerima koreksi diferensial.

Kinerja sistem navigasi sendiri di integrasikan untuk mengenali alur sistem yang telah terhubung dengan perangkat elektronik . GPS sendiri untuk membantu keakurasian yang diharapkan untuk mengurangi kesalahan-kesalahan dalam pergerakan kendaraan, dengan adanya perangkat elektronik mampu memudahkan pengaturan sistem navigasi untuk mencari arah tujuan.

V. ALGORITMA

A. Algoritma Dijkstra

Algoritme Dijkstra adalah sebuah algoritma yang dipakai dalam memecahkan permasalahan jarak terpendek untuk sebuah graf berarah (directed graph). Diskripsi matematis untuk grafik dapat diwakili $G = \{V, E\}$ berarti sebuah grafik didefinisikan oleh satu set simpul dan koleksi Edge (E). Algoritma Dijkstra

bekerja dengan membuat jalur ke satu simpul optimal pada setiap langkah. Jadi pada langkah ke n , setidaknya ada n node yang sudah kita tahu jalur terpendek. Langkah-langkah algoritma Dijkstra dapat dilakukan dengan langkah-langkah berikut:

- Tentukan titik node awal, lalu beri bobot jarak pada node pertama ke node terdekat satu per satu, Dijkstra akan melakukan pengembangan pencarian dari satu titik ke titik lain dan ke titik selanjutnya tahap demi tahap.
- Beri nilai untuk setiap titik ke titik lainnya, lalu set nilai awal pada node awal dan nilai tak hingga terhadap node lain.
- Atur semua simpul yang belum dilalui dan set node awal sebagai “Node keberangkatan”
- Dari node keberangkatan, pertimbangkan node tetangga yang belum dilalui dan hitung jaraknya dari titik keberangkatan. Jika jarak ini lebih kecil dari jarak sebelumnya (yang telah terekam sebelumnya) hapus data lama, simpan ulang data jarak dengan jarak yang baru
- Saat kita selesai mempertimbangkan setiap jarak terhadap node tetangga, tandai node yang telah dilalui sebagai “Node dilewati”. Node yang dilewati tidak akan pernah di cek kembali, jarak yang disimpan adalah jarak terakhir dan yang paling minimal bobotnya.
- Set “Node belum dilewati” dengan jarak terkecil (dari node keberangkatan) sebagai “Node Keberangkatan” selanjutnya dan ulangi langkah e.

B. Algoritma Floyd – Warshall

Algoritma Floyd-Warshall adalah salah satu varian dari pemrograman dinamis, yaitu suatu metode yang melakukan pemecahan dengan memandang solusi yang akan diperoleh sebagai suatu keputusan yang saling terkait. Artinya solusi-solusi tersebut dibentuk dari solusi yang berasal dari tahap sebelumnya dan ada kemungkinan solusi lebih dari satu. Algoritma yang ditemukan oleh Warshall untuk mencari rute terpendek merupakan algoritma yang sederhana dan mudah implementasinya. Algoritma Floyd-Warshall memiliki input graf berarah dan berbobot (V,E) , yang berupa daftar titik (node/titik V) dan daftar sisi (sisi E). Bobot garis dapat diberi simbol $w(e)$. Jumlah bobot sisi-sisi pada sebuah jalur adalah total bobot jalur tersebut. Sisi pada diperbolehkan memiliki bobot negatif, akan tetapi tidak diperbolehkan bagi graf W_{ij} untuk memiliki siklus dengan bobot negatif. Algoritma ini menghitung bobot terkecil dari semua jalur yang menghubungkan sebuah pasangan titik, dan melakukannya sekaligus untuk semua pasangan titik demi titik hingga mencapai titik tujuan dengan jumlah bobot yang paling minimum.

Dalam iterasinya untuk mencari lintasan terpendek, algoritma Floyd-Warshall membentuk n matriks, sesuai dengan iterasi- k . Ini akan menyebabkan prosesnya lambat, terutama untuk nilai n yang besar. Meskipun waktu prosesnya bukanlah yang tercepat, algoritma Floyd-Warshall sering dipergunakan untuk menghitung lintasan terpendek karena kesederhanaannya. Di samping itu,

implementasi algoritma Floyd-Warshall sangat mudah dibuat. (<https://mti.binus.ac.id/2017/11/28/algoritma-dijkstra/>)

VI. PSEUDOCODE

```
function Dijkstra(G, t, s)
  for each vertex v in V[G]
    d[v] := infinity
    previous[v] := undefined
  d[s] := 0
  S := empty set
  Q := V[G]
  while Q is not an empty set
    u := Extract_Min(Q)
    S := S union {u}
    for each edge (u,v) outgoing from u
      if d[u] + w(u,v) < d[v]
        d[v] := d[u] + w(u,v)
        previous[v] := u
```

Gambar 5.1. Algoritma Dijkstra

```
function fw(int[1..n,1..n] graph) {
  var int[1..n,1..n] dist := graph
  var int[1..n,1..n] pred
  for i from 1 to n
    for j from 1 to n
      pred[i,j] := nil
      if (dist[i,j] > 0) and (dist[i,j] < Infinity)
        pred[i,j] := i
  for k from 1 to n
    for i from 1 to n
      for j from 1 to n
        if dist[i,j] > dist[i,k] + dist[k,j]
          dist[i,j] = dist[i,k] + dist[k,j]
          pred[i,j] = pred[k,j]
  return ( dist, pred ) // Tuple of the distance and predecessor matrices
}
```

Gambar 5.2 Algoritma Floyd Warshall

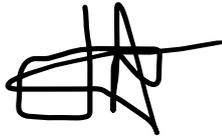
VIII. REFERENCES

- [1] Munir, Rinaldi. 2005. "Diktat Kuliah IF2153 Matematika Diskrit". Bandung : Sekolah Teknik Elektro dan Informatika ITB. ITB
- [2] https://simantu.pu.go.id/epel/edok/8733f_pengenalan_GPS.pdf. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.
- [3] <https://media.neliti.com/media/publications/169463-ID-sistem-navigasi-pada-mobile-robot-dengan.pdf>
- [4] <https://media.neliti.com/media/publications/140473-ID-penerapan-algoritma-floyd-warshall-dalam.pdf>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2020



Irgiansyah Mondo (13521167)