

Analisa Algoritma *Floyd Warshal* Untuk Optimalisasi Jalur Destinasi Kota Makassar Berbasis GPS

Abdul Ibrahim¹, Rachmat², Faizal³, Rudy Donny Liklikwatil⁴

Information Systems, Dipa Makassar University

Information Management, LP3I Polytechnic Makassar

Street. Pioneer of Independence Km. 4 Makassar, Indonesia

Street. Minasaua No. 7 Makassar, Indonesia

e-mail: ¹rachmat27udinus@gmail.com, ²abdulibrahim@dipanegara.ac.id, ³ichalabinurullah@gmail.com,

⁴rudyliklikwatil@undipa.ac.id, Email Penulis Korespondensi: rachmat27udinus@gmail.com

Abstrak

Kota Makassar memiliki perkembangan yang sangat pesat dalam pertumbuhan ekonomi, sosial politik, budaya dan sebagainya. Salah satu penyebab pertumbuhan ini didorong oleh tingginya aktivitas perjalanan masyarakat. Dalam melakukan perjalanan antar kecamatan seperti di atas, masyarakat perlu memilih jalur tercepat untuk melintasi antar kecamatan di Kota Makassar sehingga diperoleh efisiensi waktu. Floyd Warshal memberikan alternatif optimal lokal dengan harapan setiap alternatif lokal menghasilkan alternatif global yang optimal secara keseluruhan. Algoritma pencarian Floyd Warshal dapat memecahkan algoritma terbaik dalam mencari jalur terpendek. Algoritma Floyd Warshal dengan knapsack problem membantu proses optimasi mencari waktu terpendek untuk mencapai tujuan dan pada setiap langkah merupakan pilihan, untuk membuat langkah optimum lokal dengan harapan langkah-langkah yang tersisa mengarah pada solusi optimum global. untuk menentukan jarak atau waktu terpendek.

Kata Kunci : Algoritma Floyd Warshal, Kota Makassar, mencari jalur terpendek, menentukan jarak atau waktu

Abstract

Makassar City has a very rapid development in economic growth, social politics, culture and so on. One of the reasons for this growth is driven by high community travel activities. In traveling between sub-districts as above, people need to choose the fastest route to cross between sub-districts in Makassar City so that time efficiency is obtained. Floyd Warshal provides local optimal alternatives in the hope that each local alternative produces an overall optimal global alternative. Floyd Warshal's algorithm for searching can solve the best algorithm in finding the shortest path. Floyd Warshal's algorithm with knapsack problem to help the optimization process of finding the shortest time to achieve the goal and at each step is an option, to make a local optimum step in the hope that the remaining steps lead to a global optimum solution. to determine the shortest distance or time.

KeyWord : Floyd Warshal Algorithm, Makassar City, looks for the shortest path, determines the distance or time

1. Pendahuluan

Dalam kehidupan sehari-hari sering dilakukan perjalanan dari suatu tempat ketempat lain dengan mempertimbangkan efisiensi waktu dan biaya, sehingga diperlukan ketepatan dalam menentukan jalur terpendek antar kecamatan.[1] Kota Makassar memiliki perkembangan yang sangat pesat dalam pertumbuhan ekonomi, sosial politik, budaya dan lain sebagainya. Salah satu penyebab pertumbuhan ini karena didorong oleh aktifitas perjalanan masyarakatnya yang tinggi. Dalam kegiatan perjalanan antar kecamatan seperti diatas masyarakat membutuhkan pemilihan rute tercepat untuk melintasi antar kecamatan dalam Kota Makassar sehingga didapatkan efisiensi waktu. Dengan perbandingan jumlah jalan dan kendaraan yang tidak seimbang maka diperlukan pengetahuan bagi pengendara untuk memilih jalur alternatif agar mendapatkan jalur tercepat guna melintasi antar kecamatan di Kota Makassar. Ada beberapa cara mencari optimasi lintasan tercepat untuk menghubungkan satu simpul dengan simpul yang lain diantaranya adalah algoritma Greedy, Dijkstra, algoritma Semut atau Ant Colony, algoritma Floyd Warshall, algoritma Bellman Ford, algoritma Distance Vector, algoritma Ford-Fulkerson dan algoritma A-Star. Algoritma-algoritma ini dibuat untuk mencari cara yang paling efisien dan efektif untuk menganalisa pemilihan jalur tercepat dengan berbagai penambahan variabel additional.[2]

Mobilitas penduduk adalah perpindahan penduduk dari suatu daerah ke daerah lain. mobilitas penduduk ada yang bersifat nonpermanen atau sementara ada pula mobilitas penduduk permanen atau menetap, perpindahan penduduk dari suatu tempat ke tempat lain dengan melewati batas dengan tujuan untuk menetap. persebaran penduduk yang tidak merata antara desa dan kota membuat peningkatan penduduk yang berasal dari kota menjadi sangat signifikan tanpa diimbangi dengan jumlah lapangan pekerjaan, fasilitas umum, perumahan, aparat penegak hukum, penyediaan pangan dan lain sebagainya.

pengaruh-pengaruh tersebut bisa dalam bentuk mendorong, memaksa atau faktor pendorong seseorang untuk melakukan urbanisasi dan transmigrasi. pencarian instansi – instansi penting di kota besar dan merupakan hal yang penting dilakukan oleh orang – orang dari luar daerah dalam melakukan perpindahan baik nonpermanen atau sementara atau permanen atau menetap di sebuah wilayah, dengan demikian bahwa faktor kesulitan dalam mengakses jarak serta waktu salah satu hal yang perlu di pertimbangkan agar mendapatkan informasi dimana letak posisi instansi – instansi yang tepat dan akurat, Algoritma Floyd Warshal yang memiliki pendekatan untuk membangun solusi secara bertahap melalui urutan yang terus berkembang sampai solusi dari masalah telah tercapai. Floyd Warshal memberikan alternatif optimal lokal dengan harapan setiap alternatif lokal menghasilkan alternatif global yang optimal secara keseluruhan. Algoritma Floyd Warshal untuk mencari dapat menyelesaikan algoritma yang terbaik dalam mencari jalur terpendek.[3]

Algoritma Floyd Warshal dengan knapsack problem untuk membantu proses optimasi terhadap pencarian waktu terpendek untuk mencapai tujuan dan pada setiap langkahnya merupakan pilihan, untuk membuat langka optimum lokal dengan harapan bahwa langkah sisahnya mengarah ke solusi optimum global dengan Algoritma ini dapat diketahui mana algoritma yang paling bagus digunakan untuk menentukan jarak atau waktu terpendek.[4]

GPS (Global Positioning System) adalah sistem satelit navigasi penentuan posisi dengan mengirimkan sinyal gelombang mikro ke bumi. Sinyal ini lalu diterima oleh alat penerima di permukaan dan digunakan untuk menentukan posisi dan kecepatan tiga dimensi serta informasi mengenai waktu secara kontinu di seluruh dunia tanpa tergantung pada waktu dan cuaca.[5] Ketelitian dari GPS dapat mencapai beberapa mm untuk ketelitian posisinya, beberapa cm/s untuk ketelitian kecepatannya dan beberapa nano detik untuk ketelitian waktu. Prinsip penentuan posisi dengan GPS adalah dengan menggunakan metode reseksi jarak, dimana pengukuran jarak dilakukan secara simultan ke beberapa satelit yang telah diketahui koordinatnya. Pada pengukuran GPS, sedikitnya ada empat parameter yang harus ditentukan yaitu 3 parameter yang harus ditentukan: yaitu 3 parameter yang harus ditentukan, yaitu: parameter X,Y,Z dan satu parameter kesalahan akibat ketidaksinkronan jam osilator antar satelit dengan jam receiver GPS.[6]

Algoritma Warshall menggunakan matriks *boolean* untuk mencari ada tidaknya jalan dari titik satu ke titik yang lain. Algoritma ini kemudian dikembangkan untuk mengurangi waktu komputasi sehingga hanya menggunakan satu perkalian matriks. Bentuk algoritma untuk mencari R adalah sebagai berikut :

Step 1: $T = R$

Step 2: For $k = 1$ to n do

For $i = 1$ to n do

$$T[i,j] = T[i,j] + (T[i,k] . T[k,j])$$

Step 3 : $R = T$

Algoritma Warshall dikembangkan oleh R.W. Floyd sehingga matriks merupakan graf berbobot dan bukan lagi matriks *boolean*. Algoritma ini digunakan untuk mencari jarak terpendek antara semua titik dalam graf. Ketiadaan garis yang menghubungkan sebuah pasangan dilambangkan dengan tak-hingga. Dalam pengertian lain algoritma Floyd-Warshall adalah algoritma yang akan memilih satu jalur terpendek dan teraman dari beberapa alternatif jalur yang dihasilkan dari proses kalkulasi.[7]

Algoritma Floyd-Warshall mencoba untuk memberikan solusi optimum yang memiliki pemikiran terhadap konsekuensi yang ditimbulkan dari pengambilan keputusan pada suatu tahap. Algoritma Floyd-Warshall mampu mengurangi pencarian keputusan yang tidak mengarah ke solusi. Prinsip yang dipegang oleh algoritma ini adalah prinsip optimalitas, yaitu jika solusi per-tahap optimal, maka bagian solusi sampai suatu tahap (misalnya tahap ke- i) juga optimal.[8] Bentuk algoritma untuk mencari $R+$ adalah sebagai berikut :

Step 1: Jika antara $i-j$ merupakan jalan (edge), $T[i,j]$ berisi panjang jalan. Jika tidak, $T[i,j]=$

Step 2: For $k = 1$ to n do

For $i = 1$ to n do

For $j = 1$ to n do

$$T[i,j] = \min \{T[i,j], T[i,k] . T[k,j]\}$$

Step 3: $R = T$

Algoritma Warshall dikembangkan oleh R.W. Floyd sehingga matriks merupakan graf berbobot dan bukan lagi matriks *boolean*. Algoritma ini digunakan untuk mencari jarak terpendek antara semua titik dalam graf. Ketiadaan garis yang menghubungkan sebuah pasangan dilambangkan dengan tak-hingga.

Dalam pengertian lain algoritma Floyd-Warshall adalah algoritma yang akan memilih satu jalur terpendek dan teraman dari beberapa alternatif jalur yang dihasilkan dari proses kalkulasi.

Algoritma Floyd-Warshall mencoba untuk memberikan solusi optimum yang memiliki pemikiran terhadap konsekuensi yang ditimbulkan dari pengambilan keputusan pada suatu tahap. Algoritma Floyd-Warshall mampu mengurangi pencarian keputusan yang tidak mengarah ke solusi. Prinsip yang dipegang oleh algoritma ini adalah prinsip optimalitas, yaitu jika solusi per-tahap optimal, maka bagian solusi sampai suatu tahap (misalnya tahap ke-*i*) juga optimal (Yusaputra, 2013). Bentuk algoritma untuk mencari $R+$ adalah sebagai berikut :

Step 1: Jika antara i - j merupakan jalan (edge), $T[i,j]$ berisi panjang jalan. Jika tidak $T[i,j]=$

Step 2 : For $k = 1$ to n do

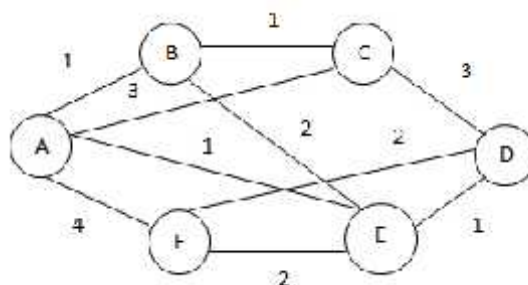
 For $i= 1$ to n do

 For $j=1$ to n do

$$T[i,j]=\min \{T[i,j],T[i,k].T[k,j]\}$$

Step 3 : $R = T$

Algoritma Floyd-Warshall sangat efisien dari sudut pandang penyimpanan data karena dapat diimplementasikan dengan hanya pengubahan sebuah matriks jarak. Berikut ini adalah contoh dari penggunaan algoritma Floyd-Warshall. Misalkan ada graf berbobot seperti pada Gambar1



Gambar 1. Graf berbobot untuk algoritma floyd-warshall

Graf berbobot ini dapat dijadikan sebagai matriks inisialisasi seperti pada Tabel1

Tabel 1. Inisialisasi Matriks T

Jalur	A	B	C	D	E	F
A	0	1	3		1	4
B	1	0	1		2	
C	3	1	0	3		
D			3	0	1	2
E	1	2		1	0	2
F	4			2	2	0

GPS Navigation System

GPS (Global Positioning System) adalah sistem yang digunakan untuk menentukan letak di permukaan bumi dengan bantuan penyalarsaan (*synchronization*) sinyal satelit. Sistem ini menggunakan 24 satelit untuk mengirimkan sinyal gelombang mikro ke bumi. Sinyal tersebut kemudian diterima oleh alat penerima di permukaan bumi dan digunakan untuk menentukan letak, kecepatan, arah, dan waktu. Sistem yang serupa dengan *GPS* antara lain: *GLONASS* dari Rusia, *Galileo* dari Uni Eropa, dan *IRNSS* dari India. *GPS* sendiri memiliki 2 jenis, yaitu *GPS* dan *A-GPS*, dimana sistem *GPS* ini sering kita jumpai pada ponsel yang ada dipasaran sekarang ini. Berikut penjelasan dari perbedaan *GPS* dengan *A-GPS*.

Jalur Terpendek

lintasan terpendek antara dua atau beberapa simpul lebih yang berhubungan. Persoalan mencari lintasan terpendek di dalam graf merupakan salah satu persoalan optimasi. Persoalan ini biasanya direpresentasikan dalam bentuk graf. Graf yang digunakan dalam pencarian lintasan terpendek atau shortest path adalah graf berbobot (weighted graph), yaitu graf yang setiap sisinya diberikan suatu nilai

atau bobot. Bobot pada sisi graf dapat menyatakan jarak antar kota, waktu pengiriman pesan, ongkos pembangunan, dan sebagainya [6].

Lintasaan terpendek dapat di cari dengan memodelkan jaingan jalan ke sebuah graf. Graf yang digunakan adalah graf berbobot, yaitu graf yang setiap sisinya memiliki nilai atau bobot [7].

Pencarian (Searching)

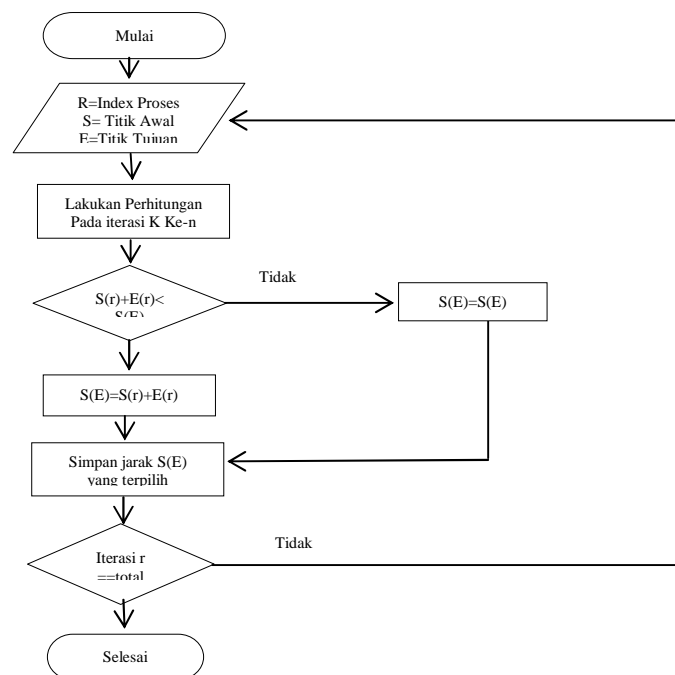
Pada dasarnya teknik searching (pencarian) dapat dibagi menjadi 2 (dua) kelompok yaitu: pencarian buta (blind search) dan pencarian terbimbing (heuristic search) [5]. Untuk mengukur performa metode pencarian, terdapat empat criteria yang dapat dilakukan adalah :

1. Completeness: apakah metode tersebut menjamin penemuan solusi jika solusinya memang ada?
2. Time complexity: berapa lama waktu yang diperlukan?
3. Space complexity: berapa banyak memori yang diperlukan?
4. Optimality: apakah metode tersebut menjamin menemukan solusi yang terbaik jika terdapat beberapa solusi yang berbeda?

2. Metodologi Penelitian

2.1. Flowchart

Dari proses gambar 2 inisialisasi membentuk nilai matriks yang ditunjukkan pada tabel. Nilai matriks ini digunakan untuk menganalisis apakah nilai yang ada pada matriks tersebut dihitung dalam mencari optimasi rute atau tidak. Pada awal perhitungan semua nilai akan diinisialisasikan ke nilai 1. Apabila nilai tersebut sudah dihitung maka program akan merubahnya menjadi 0 dan tidak akan dihitung lagi dalam perhitungan selanjutnya.



3.2 Analisis Manual

Dari data waktu tempuh yang didapatkan dari google maps maka berikut adalah tabel data yang didapatkan dan contoh perhitungan manual dengan algoritma greedy.

Tabel 1. Perhitungan Manual

X	A	B	C	D	E	F
A	X	3	4	6	23	30
B	3	X	1	7	25	32
C	4	1	X	8	26	33
D	6	6	7	X	19	28
E	18	20	21	16	X	17
F	29	31	32	27	17	X

- A. Kantor Walikota
- B. Polrestaes
- C. Kantor Gubernur
- D. SPN
- E. KODAM
- F. RSUD DR. Wahidin Sudirohusodo

Dari data tersebut di misalkan total alokasi yang dapat dimiliki adalah 6 Jam (360 Menit) dan perjalanan dimulai dari titik Kantor Walikota (A). Lokasi dan Alokasi waktu masing-masing tempat yang akan dikunjungi adalah sebagai berikut :

- a. Polrestaes(B) : 200 Menit = 3,33 Jam
- b. Kantor Gubernur(C) : 120 Menit = 2 Jam
- c. SPN(D) : 60 Menit = 1 Jam
- d. RSUD(F) : 120 Menit = 2 Jam

Model perhitungan pertama ini menjumlah waktu tempuh dan waktu kunjung terlebih dahulu, kemudian dicari nilai yang terkecil. Apabila instansi tersebut tidak dikunjungi maka tidak termasuk dalam perhitungan, titik terpilih sebagai titik awal perhitungan berikutnya

Maka perhitungannya adalah sebagai berikut :

- a. $A - B = 3 + 200 = 203$ Menit
- b. $A - C = 4 + 120 = 124$ Menit
- c. $A - D = 6 + 60 = 66$ Menit
- d. $A - F = 30 + 120 = 150$ Menit

Dari perhitungan tersebut maka $A - D$ adalah nilai terkecil dan D titik yang dipilih. Maka sisa waktu yang dimiliki menjadi $360 - 66 = 294$ Menit, kemudian perhitungan dilanjutkan dari titik D

- a. $D - B = 6 + 200 = 206$ Menit
- b. $D - C = 7 + 120 = 127$ Menit
- c. $D - F = 28 + 120 = 133$ Menit

Dari perhitungan tersebut maka $D - C$ adalah nilai terkecil dan C titik yang dipilih. Maka sisa waktu yang dimiliki menjadi $294 - 127 = 167$ Menit kemudian perhitungan dilanjutkan dari titik C

- a. $C - B = 1 + 200 = 201$ Menit
- b. $C - F = 33 + 120 = 153$ Menit

Dari perhitungan tersebut maka $C - F$ adalah nilai terkecil dan F titik yang dipilih, maka sisa waktu yang dimiliki menjadi $167 - 153 = 14$ Menit. Kemudian perhitungan dilanjutkan dari titik F

- a. $F - B = 31 + 200 = 231$ Menit

Hanya tersisa B yang belum dikunjungi tetapi waktu total, waktu tidak dapat mencukupi sehingga perhitungan selesai, jadi titik B tidak diperhitungkan maka rute perjalanan optimal yang memenuhi dengan perhitungan knapsack greedy adalah :

- a. Kantor Gubernur : 120 Menit
- b. SPN : 60 Menit
- c. RSUD : 120 Menit

Dengan total waktu kunjungan dan perjalanan adalah 355 Menit atau 5 Jam 55 Menit

3. Hasil Dan Pembahasan

Pada hasil penelitian diungkapkan data jalan yang mempunyai variabel panjang, lebar, kondisi jalan, kepadatan lalu lintas, dan tanjakan. Untuk pengujian optimasi jalur tercepat akan dilakukan dengan algoritma



Gambar 2 peta jalan kota Makassar

a. Peta Jalan.

Berikut pada gambar 2 diatas Kota Makassar, yang berasal dari beberapa kecamatan antara lain Kecamatan Panakukkang, Kecamatan Tamalanrea, Kecamatan Sombaopu, Kecamatan ujungPandang, dan Kecamatan UjungTanah.

b. Perhitungan Pemakaian Bahan Bakar.

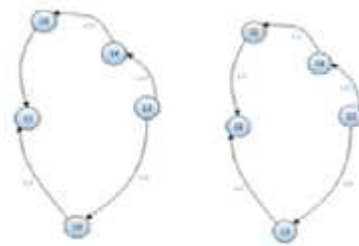
Cara menghitung bahan bakar yang diperlukan didapatkan dari rata-rata jarak tempuh yang dihasilkan dalam menghabiskan bakar per 1 liter. Perhitungan kebutuhan bahan bakar dengan perhitungan matematika dapat dilakukan dengan menentukan jarak tempuh dibagi dengan standar bahan bakat sepeda motor rata-rata sebesar 58,8 km/liter [5]. Rata rata pemakaian 1 liter bahan bakar dapat menempuh 58,5 km/liter (B-C). Jumlah kebutuhan bahan bakar disimbolkan dengan (A=Standar bahan bakar dalam liter) Sebagai contoh, jika jarak yang ditempuh adalah 7.2 km, maka perhitungan kebutuhan bahan bakar adalah sebagai berikut:

$$\frac{B - C}{A}$$

$$\frac{R \quad C}{A} = \frac{8,7 \text{ km}}{58,5 \text{ km/liter}} = 0,14 \text{ km/liter}$$

Gambar 3 Floyd Warshal iterasi ke-1, 2 dan 3

$t = s / v = 0,3 / 40 = 0,07 \text{ jam} = 1,7 \text{ menit}$. Diasumsikan dengan kecepatan rata-rata 40 km perjam dengan jarak 1,3 km: $t = s / v = 1,3 / 40 = 0,03 \text{ jam} = 1,9 \text{ menit}$. Node 12 ke node 22 = $1,8 - 2 + 1 + 0 + 0 = 1,3 \text{ km}$. Setelah dimodifikasi, dari jalur node 12 ke 3 - 4



Gambar 4 Floyd Warshal iterasi ke-3 dan 4

c. Perhitungan dengan Floyd Warshal Standar.

Dengan menggunakan Algoritma Floyd Warshal Standar, perhitungan untuk mencari jalur terpendek mulai node 12 ke node 22 menggunakan persamaan 5.1. $M_{i,v} = \min(M_{i-1,v}, M_{i-1,n} + C_{vn}) \dots\dots\dots(3)$ Di mana i adalah iterasi, v adalah vertex (node), n adalah neighbor node, dan C adalah cost. Langkah-langkah yang diperlukan adalah sebagai berikut: 1. Dari node 12 ke node 22 adalah: jalur 12-10 dengan jarak 2 km, jalur 10-22 dengan jarak 4,3 km. Kemudian dengan fungsi matematik sebagai berikut: $M[12,22] = \min(M[12,10], M[10,22]) = \min(2, 4,3) = 2$. Diasumsikan dengan kecepatan rata-rata 40 km per jam dengan jarak 2 km maka didapatkan sebagai berikut: $t = s / v = 2 / 40 = 0,05$ jam = 3 menit. Diasumsikan dengan kecepatan rata-rata 40 km per jam dengan jarak tempuh 4,3 km, maka didapatkan hasil: $t = s / v = 4,3 / 40 = 0,107$ jam = 6,45 menit.

Perhitungan dengan Floyd Warshal yang modifikasi untuk menghitung Floyd Warshal yang dimodifikasi yaitu dengan menambahkan bobot kondisi jalan, lebar jalan, tanjakan dan turunan serta kepadatan lalu lintas. Berikut ini adalah tabel-tabel pembobotan masing-masing parameter.

Dengan menggunakan Algoritma Floyd Warshal termodifikasi, perhitungan untuk mencari jalur tercepat mulai node 12 ke node 22 menggunakan rumus sebagai berikut:

$$M_{i,v} = \min(M_{i-1,v}, M_{i-1,n} + C_{vn} + \text{Kondisi} + \text{Lebar} + \text{Tanjakan} + \text{Kepadatan}) \dots\dots\dots 4$$

d. Hasil Pengujian Floyd Warshal Standart dan Floyd Warshal Modifi.

Dari hasil analisis tabel di atas menjelaskan bahwa terdapat 2 proses, yaitu Floyd Warshal Standart dan Floyd Warshal Modifikasi. Dari hasil pengujian algoritma Floyd Warshal modifikasi lebih baik karena nilai yang dihasilkan lebih kecil dari nilai Floyd Warshal Standart sehingga mencapai optimasi lebih baik dari contoh 5 jalur yang ada di Kota Makassar dari jarak, waktu dan bahan bakar yang diperlukan.

e. Pengujian Data Riil.

Optimasi jalur tercepat menggunakan Algoritma Floyd Warshal Modifikasi akan dibandingkan dengan hasil jalur tercepat menggunakan Algoritma Floyd Warshal Standart pada validasi program berikut ini. Data validasi diambil menggunakan dua macam problem. Problem pertama adalah dari simpul 25 menuju simpul 22, sedangkan problem kedua adalah simpul 7 menuju simpul 26 dengan kecepatan rata-rata 40 km/jam.

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan beberapa hal sebagai berikut: Dengan memodifikasi Algoritma Floyd Warshal dimungkinkan untuk memberikan pembobotan untuk beberapa kriteria, Algoritma Floyd Warshal yang hanya memungkinkan untuk mendefinisikan jarak tanpa kriteria tambahan. Dengan memanfaatkan compiler Borland Delphi 7 dapat dihasilkan software optimasi menerapkan Algoritma Floyd Warshal termodifikasi.

Daftar Pustaka

[1] 赵岩, "No Title ," *Pontif. Univ. Catol. del Peru*, vol. 8, no. 33, p. 44, 2014.
 [2] 宗成庆, "No Title统计自然语言处理 (第二版)," pp. 17–21.
 [3] M. Anis, "Penggunaan Algoritma Dynamic Programming pada Aplikasi GPS Car Navigation System." 2011.
 [4] D. W. Nugraha, A. Amriana, and A. Arif, "Implementasi Algoritma Floyd Warshall Pada Pada Pencarian Lokasi Agen Bus, Tour And Travel, Dan Rental Mobil Di Kota Palu Berbasis Android," *Sains, Apl. Komputasi dan Teknol. Inf.*, vol. 2, no. 2, p. 41, 2020, doi:

- 10.30872/jsakti.v2i2.2400.
- [5] R. Yusaputra, "Aplikasi Mobile Pencarian Rute Terpendek Lokasi Fasilitas Umum Berbasis Android Menggunakan Algoritma Floyd-Warshall Menggunakan Algoritma Floyd-Warshall," 2013.
- [6] Y. Darnita and R. Toyib, "Penerapan Algoritma Greedy Dalam Pencarian Jalur Terpendek Pada Instansi-Instansi Penting Di Kota Argamakmur Kabupaten Bengkulu Utara," *J. Media Infotama*, vol. 15, no. 2, 2019, doi: 10.37676/jmi.v15i2.867.
- [7] R. Ghaniy and R. Darmawan, "Analisa dan Penerapan Algoritma Floyd Warshal Untuk Optimalisasi Jalur Berbasis GPS," *Teknois J. Ilm. Teknol. Inf. dan Sains*, vol. 8, no. 2, pp. 67–78, 2019, doi: 10.36350/jbs.v8i2.16.
- [8] A. P. Agustian, H. P. Sholeh, and A. M. M, "Optimasi Jalur Tercepat dengan Menggunakan Modifikasi Algoritma Bellman Ford," *Eccis*, vol. 9, no. 2, pp. 168–172, 2015.