

Optimasi Rute Bus Pariwisata dengan *Dynamic Programming*

Eric Jonathan / 13516117
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
135@itb.ac.id

Abstract— This Paper explains about how to optimize Bus route using Dynamic Programming algorithm, specifically Traveling Salesperson Problem. Bus route problem is one of examples in real life for Travelling Salesperson Problem, which is to find the minimum weight of Hamilton Circuit. There'll be explanation about the benefits of this algorithm for satisfying the problem and how it fits the solution. The purpose of this paper is to increase the reader's knowledge about Dynamic Programming version of Travelling Salesperson Problem and the optimization of .

Keywords—Dynamic Programming, Traveling Salesperson Problem, Optimization, Route

I. PENDAHULUAN

Pariwisata selalu menjadi daya tarik dan sumber pendapatan dari suatu daerah. Oleh karena itulah tak jarang pariwisata menjadi salah satu sektor ekonomi utama. Indonesia yang penuh kekayaan alam dan peninggalan budaya tentunya sangat kaya akan tempat wisata. Untuk menunjang akses ke berbagai tempat pariwisata ini tentunya pemerintah daerah setempat membangun berbagai fasilitas yang mempermudah pengunjung, salah satunya pada bidang transportasi. Bus wisata merupakan fasilitas yang sudah umum kita jumpai di berbagai kota besar di Indonesia, contohnya saja Bandung. Bus wisata tentunya menjadi sarana transportasi yang membantu pendatang dalam mengelilingi tempat-tempat wisata di suatu daerah dengan harga yang terjangkau dan akses yang relatif mudah. Bus wisata akan mengelilingi objek-objek wisata dengan jalur tertentu yang sudah ditetapkan sebelumnya.

Tentunya salah satu daya tarik penggunaan bus wisata ini adalah harganya yang terjangkau bagi masyarakat. Untuk menunjang hal tersebut tentunya bus wisata perlu mengoptimalkan biaya operasionalnya sesangkil mungkin. Rute tempuh bus dalam mengelilingi tiap objek wisata merupakan hal yang paling mungkin untuk dioptimalkan dan merupakan faktor terbesar dalam biaya operasional bus wisata. Penghematan waktu dan biaya tentunya menjadi sasaran utama dalam optimalisasi ini. Di sinilah kita dapat menerapkan berbagai algoritma yang ada untuk melakukan optimalisasi dan *Dynamic Programming* dapat menjadi pilihan yang paling tepat dan mangkus. Pengoptimalan rute bus wisata dapat membuat operasional bus wisata berjalan dengan lebih baik dan tetap mendapatkan daya tarik di masyarakat sebagai sarana

transportasi yang merakyat.

II. TRAVELING SALESPERSON PROBLEM

Permasalahan TSP (Traveling Salesman Problem) adalah permasalahan optimalisasi klasik dimana terdapat seorang *salesman* dimana ia harus mengunjungi semua kota dimana tiap kota hanya boleh dikunjungi sekali, dan perjalanan ini harus dimulai dari dan kembali ke kota asal. Tujuannya adalah menentukan rute perjalanan terpendek untuk melewati setiap kota tersebut dan kembali ke kota asal.

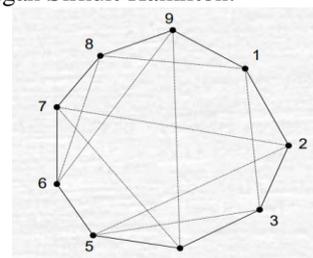
Masalah optimalisasi TSP ini sering ditemui sarana transportasi, begitu juga pada pengiriman surat atau barang. Optimalisasi dari transportasi dan pengiriman sangat ditentukan oleh rute atau lintasan yang diambil. Oleh karena itu solusi optimal dari permasalahan TSP ini, akan sangat membantu perusahaan pengiriman surat atau barang begitu juga bus pariwisata untuk meningkatkan efisiensi proses pengiriman barang, baik dari segi waktu maupun dana. Hingga kini kompleksitas algoritma permasalahan TSP masih tidak dapat diketahui pasti, bahkan setelah 50 tahun lebih pencarian. Hal tersebut menjadikan TSP menjadi salah satu permasalahan yang hingga kini belum terselesaikan dalam banyak permasalahan optimasi matematis. TSP juga dapat dilihat sebagai masalah untuk menemukan bobot minimum pada Sirkuit Hamilton.

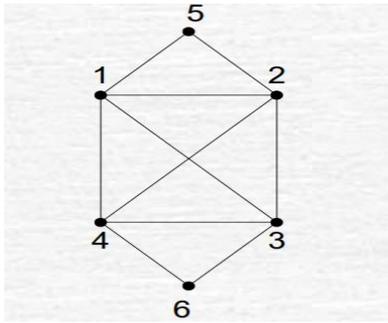
Sirkuit Hamilton

Sirkuit Hamilton ialah sirkuit yang melalui tiap simpul di dalam graf tepat satu kali, kecuali simpul asal (sekaligus simpul akhir) yang dilalui dua kali.

Syarat cukup supaya graf sederhana G dengan n (≥ 3) buah simpul adalah graf Hamilton ialah bila derajat tiap simpul paling sedikit $n/2$ (yaitu, $d(v) \geq n/2$ untuk setiap simpul v di G). Setiap graf lengkap adalah graf Hamilton.

Contoh graf dengan Sirkuit Hamilton:





Sejarah Permasalahan TSP

TSP pertama kali diperkenalkan oleh Rand pada tahun 1948. TSP melibatkan seorang travelling salesman yang harus melakukan kunjungan ke sejumlah kota dalam menjajakan produknya. Rangkaian kota-kota yang dikunjungi harus membentuk suatu jalur sedemikian sehingga kota-kota tersebut hanya boleh dilewati tepat satu kali dan kemudian kembali lagi ke kota awal (untuk setoran ke bosnya, hehe). Maka dari itu, penyelesaian terhadap permasalahan TSP ini membutuhkan optimasi untuk dapat memperoleh jalur terpendek.

Sejarah TSP bisa ditelusur dari Euler yang mempelajari Knight Tour's Problem (1759), Kirkman yang mempelajari grafik polihedron (1856) maupun Hamilton yang membuat game Icosian (1856) yang bertujuan mencari jalur sirkuit berbasis grafik polihedron yang memenuhi kondisi tertentu. Istilah TSP sendiri diperkirakan berasal dari buku yang diterbitkan oleh seorang veteran salesman sekitar tahun 1930an di Jerman, meski dalam buku ini masalah TSP lebih dibahas dari aspek bisnis dan belum diformulasikan secara matematis.

Karakteristik permasalahan TSP

Secara ringkas, berikut adalah karakteristik dari permasalahan TSP:

1. Perjalanan berawal dan berakhir dari dan ke kota awal
2. Ada sejumlah kota yang semuanya harus dikunjungi tepat satu kali
3. Perjalanan tidak boleh kembali ke kota awal sebelum semua kota tujuan dikunjungi
4. Tujuan dari permasalahan ini adalah meminimumkan total jarak yang ditempuh salesman dengan mengatur urutan-urutan kota yang harus dikunjungi

Perkembangan Penyelesaian TSP

- a. 49 Kota (*DANTZIG49*)

DANTZIG49 adalah permasalahan yang diteliti oleh Dantzig, Fulkerson, dan Johnson yang dapat kita lihat dalam makalah mereka tahun 1954 tentang solusi dari permasalahan TSP yang terdiri dari satu kota tiap 48 negara bagian di Amerika Serikat ditambah kota Washington, D. C. Para penulis bekerja dengan 49 kota tersebut. Jalur optimal dari 49 kota ini, menggunakan jalan pintas yang terdapat pada 7 kota selain 49 kota tersebut. Jalur yang dibuat berdasarkan jarak pada setiap jalur dalam kota diantara 49 kota tersebut. Para penulis mengatakan bahwa mereka mendapat tabel jarak dari Bernice Brown karyawan Perusahaan Rand.

- b. 120 Kota (*GR120*)

GR120 telah menjadi pengujian standar bagi permasalahan

TSP sejak tahun 1977. GR120 ini mempunyai 120 titik yang terdiri dari jarak tempuh antara 120 kota yang terdapat di sekitar Jerman. Daftar kota-kota ini terdapat pada Atlas Umum Negara Jerman tahun 1967/68.

- c. 318 Kota (*LIN318*)

LIN318 muncul pada tahun 1973 dalam makalah yang ditulis oleh S.Lin dan B.W. Kernighan. Data dari LIN318 terdiri dari 318 yang muncul dari hasil pengeboran, dimana bornya adalah sebuah sinar laser. Lin dan Kernighan menulis bahwa mereka mendapatkan datanya dari R.Haberman. Permasalahan ini pertama diselesaikan oleh H.Crowder dan M. W. Padberg pada tahun 1980.

- d. 532 Kota (*ATT532*)

ATT532 muncul pada tahun 1987 didalam makalah yang ditulis oleh M. Padberg dan G. Rinaldi. Dalam ATT532 terdapat data mengenai 532 kota yang berlokasi di Benua Amerika Serikat, dan Padberg serta Rinaldi menulis bahwa mereka mendapatkan permasalahannya dari Shen Lin karyawan dari laboratorium AT&T Bell.

- e. 666 Kota (*GR666*)

GR666 pertama kali diselesaikan oleh O. Holland dan M. Groetschel, yang muncul dalam Tesis PhD Olaf Holland's pada tahun 1987. Data terdiri dari 666 kota menarik yang tersebar di seluruh dunia.

- f. 2392 Kota (*PR2392*)

PR2392 memiliki data sebanyak 2,392 titik, hasil kontribusi dari M. Padberg dan G. Rinaldi. Layout dari titik-titik tersebut dibuat oleh Perusahaan Tektronics.

- g. 7397 Kota (*PLA7397*)

PLA7397 adalah sebuah programmed logic array application, yang terdiri dari 7,397 kota.

- h. 15112 Kota (*DI5112*)

DI5112 ini mempunyai data mengenai 15,112 kota-kota yang terdapat di Negara Jerman.

- i. 24978 Kota (*Sweden24978*)

24,978 kota di Swedia, didapat datanya dari National Imagery and Mapping Agency database nama-nama fitur geografi.

Permasalahan TSP ini dapat diselesaikan dengan metode optimasi salah satunya adalah dengan Program Dinamis.

III. PROGRAM DINAMIS

A. Definisi

Program Dinamis (*Dynamic Programming*) adalah metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan tahapan (*stage*) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Istilah program dinamis muncul karena perhitungan solusi menggunakan tabel-tabel.

Terdapat perbedaan antara program dinamis dengan algoritma *Greedy*. Pada program dinamis terdapat lebih dari satu rangkaian keputusan yang dipertimbangkan, sedangkan pada algoritma *Greedy* hanya menghasilkan satu rangkaian keputusan. Juga seperti yang kita tahu program dinamis merupakan penyelesaian masalah dengan prinsip optimalitas yang tidak lain adalah untuk mendapatkan hasil yang paling optimal.

B. Prinsip Optimalitas

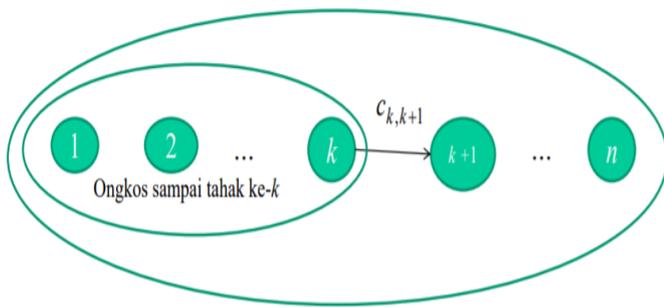
Pada program dinamis, rangkaian keputusan yang optimal dibuat dengan menggunakan prinsip optimalitas.

Prinsip Optimalitas:

jika solusi total optimal, maka bagian solusi sampai tahap ke-k juga optimal.

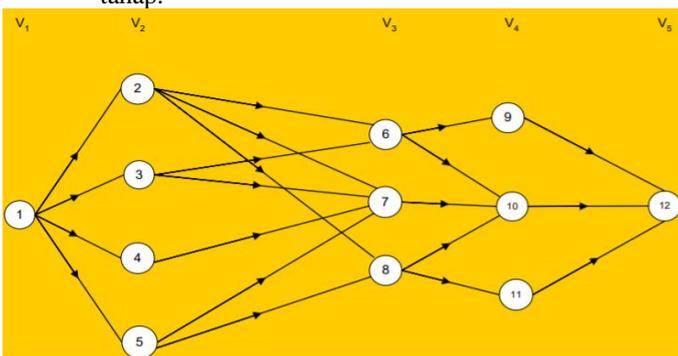
Prinsip optimalitas berarti bahwa jika kita bekerja dari tahap k ke tahap k+1, kita dapat menggunakan hasil optimal dari tahap k tanpa harus kembali ke tahap awal.

Ongkos pada tahap k+1 = (ongkos yang dihasilkan pada tahap k) + (ongkos dari tahap k ke tahap k+1)



C. Karakteristik Persoalan Program Dinamis

1. Persoalan dapat dibagi menjadi beberapa tahap (*stage*), yang pada tiap tahap hanya diambil satu keputusan.
2. Masing-masing tahap terdiri dari sejumlah status (*state*) yang berhubungan dengan tahap tersebut. Secara umum, status merupakan bermacam kemungkinan masukan yang ada pada tahap tersebut. Graf multistage artinya tiap simpul dalam graf tersebut menyatakan status, sedangkan V_1, V_2, \dots menyatakan tahap.



3. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan ke status berikutnya pada tahap berikutnya.
4. Ongkos (*cost*) pada suatu tahap meningkat secara teratur (*steadily*) dengan bertambahnya jumlah tahapan.
5. Ongkos pada suatu tahap bergantung pada ongkos tahap-tahap yang sudah berjalan dan ongkos pada tahap tersebut.

6. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya.
7. Adanya hubungan rekursif yang mengidentifikasi keputusan terbaik untuk setiap status pada tahap k memberikan keputusan terbaik untuk setiap status pada tahap k+1.
8. Prinsip optimalitas berlaku pada persoalan tersebut.

D. Jenis Program Dinamis

1. Forward dynamic programming

Program ini bergerak mencari solusi dari tahap 1, ke 2, 3 dan seterusnya sampai tahap n. Runtutan peubah keputusan adalah $x_1, x_2, x_3, \dots, x_n$.

Prinsip Optimalitas pada Program Dinamis maju:
 Ongkos pada tahap k+1 = (ongkos yang dihasilkan pada tahap k) + (ongkos dari tahap k ke tahap k+1)

$$k = 1, 2, \dots, n-1$$

2. Backward dynamic programming

Program dinamis mundur bergerak dari tahap n, terus mundur ke tahap n-1, n-2 dan seterusnya sampai tahap 1. Runtutan peubah keputusan adalah x_n, x_{n-1}, \dots, x_1 .

Prinsip Optimalitas pada Program Dinamis mundur:
 Ongkos pada tahap k = (ongkos yang dihasilkan pada tahap k+1) + (ongkos dari tahap k+1 ke tahap k)

$$k = n, n-1, \dots, 1.$$

E. Langkah-langkah Pengembangan Algoritma Program Dinamis

1. Karakteristikan struktur solusi optimal.
2. Definisikan secara rekursif nilai solusi optimal.
3. Hitung nilai solusi optimal secara maju atau mundur.
4. Konstruksi solusi optimal.

F. TSP dengan Program Dinamis

Misalkan $G = (V, E)$ adalah graf lengkap berarah dengan sisi-sisi yang diberi harga $c_{ij} > 0$ untuk setiap i dan j adalah simpul-simpul di dalam V. Misalkan $V = n$ dan $n > 1$. Setiap simpul diberi nomor 1, 2, ..., n.

Asumsikan perjalanan (tur) dimulai dan berakhir pada simpul 1.

Setiap tur pasti terdiri dari sisi (1, k) untuk beberapa $k \in V - \{1\}$ dan sebuah lintasan dari simpul k ke simpul 1.

Lintasan dari simpul k ke simpul 1 tersebut melalui setiap

simpul di dalam $V - \{1, k\}$ tepat hanya sekali.

Prinsip Optimalitas: jika tur tersebut optimal maka lintasan dari simpul k ke simpul 1 juga menjadi lintasan k ke 1 terpendek yang melalui simpul-simpul di dalam $V - \{1, k\}$.

Misalkan $f(i, S)$ adalah bobot lintasan terpendek yang berawal pada simpul i , yang melalui semua simpul di dalam S dan berakhir pada simpul 1.

Nilai $f(1, V - \{1\})$ adalah bobot tur terpendek. Berdasarkan prinsip optimalitas tersebut, diperoleh hubungan rekursif sebagai berikut:

$$f(1, V - \{1\}) = \min_{2 \leq k \leq n} \{c_{1k} + f(k, V - \{1, k\})\} \quad (1)$$

Dengan merampatkan persamaan (1), diperoleh

$$f(i, \emptyset) = c_{i,1}, \quad 2 \leq i \leq n$$

(basis)

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

(rekurens) (2)

Persamaan (1) dapat dipecahkan untuk memperoleh $f(1)$ jika kita mengetahui $f(k, V - \{1, k\})$ untuk semua pilihan nilai k . Nilai f tersebut dapat diperoleh dengan menggunakan persamaan (2).

Kita menggunakan persamaan(2) untuk memperoleh $f(i, S)$ untuk $S = 1$, kemudian kita dapat memperoleh $f(i, S)$ untuk $S = 2$, dan seterusnya. Bila $S = n - 1$, nilai i dan S ini diperlukan sedemikian sehingga $i \neq 1, 1 \notin S$ dan $i \notin S$.

Tinjau persoalan TSP untuk $n = 4$:

0	10	15	20
5	0	9	10
6	13	0	12
8	8	9	0

Ini adalah matriks ketetanggan dari suatu graf yang akan kita cari lintasan terpendeknya.

Tahap 1:

$$f(i, \emptyset) = c_{i,1}, \quad 2 \leq i \leq n$$

Diperoleh:

$$f(2, \emptyset) = c_{21} = 5;$$

$$f(3, \emptyset) = c_{31} = 6;$$

$$f(4, \emptyset) = c_{41} = 8;$$

terdapat tiga kemungkinan yang dapat dijadikan lintasan pertama.

Tahap 2:

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

untuk $S = 1$

Diperoleh:

$$f(2, \{3\}) = \min\{c_{23} + f(3, \emptyset)\} = \min\{9 + 6\} = \min\{15\} = 15$$

$$f(3, \{2\}) = \min\{c_{32} + f(2, \emptyset)\} = \min\{13 + 5\} = \min\{18\} = 18$$

$$f(4, \{2\}) = \min\{c_{42} + f(2, \emptyset)\} = \min\{8 + 5\} = \min\{13\} = 13$$

$$f(2, \{4\}) = \min\{c_{24} + f(4, \emptyset)\} = \min\{10 + 8\} = \min\{18\} = 18$$

$$f(3, \{4\}) = \min\{c_{34} + f(4, \emptyset)\} = \min\{12 + 8\} = \min\{20\} = 20$$

$$f(4, \{3\}) = \min\{c_{43} + f(3, \emptyset)\} = \min\{9 + 6\} = \min\{15\} = 15$$

Tahap 3:

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

untuk $S = 2$ dan $i \neq 1, 1 \notin S$ dan $i \notin S$.

Diperoleh:

$$\begin{aligned} f(2, \{3, 4\}) &= \min\{c_{23} + f(3, \{4\}), c_{24} + f(4, \{3\})\} \\ &= \min\{9 + 20, 10 + 15\} \\ &= \min\{29, 25\} = 25 \end{aligned}$$

$$\begin{aligned} f(3, \{2, 4\}) &= \min\{c_{32} + f(2, \{4\}), c_{34} + f(4, \{2\})\} \\ &= \min\{13 + 18, 12 + 13\} \\ &= \min\{31, 25\} = 25 \end{aligned}$$

$$\begin{aligned} f(4, \{2, 3\}) &= \min\{c_{42} + f(2, \{3\}), c_{43} + f(3, \{2\})\} \\ &= \min\{8 + 15, 9 + 18\} \\ &= \min\{23, 27\} = 23 \end{aligned}$$

Dengan menggunakan persamaan (1) diperoleh:

$$\begin{aligned} f(1, \{2, 3, 4\}) &= \min\{c_{12} + f(2, \{3, 4\}), c_{13} + f(3, \{2, 4\}), c_{14} + f(4, \{2, 3\})\} \\ &= \min\{10 + 25, 15 + 25, 20 + 23\} \\ &= \min\{35, 40, 43\} = 35 \end{aligned}$$

Jadi, bobot tur yang berawal dan berakhir di simpul 1 adalah 35.

Lintasan yang dilalui di dalam tur tersebut dapat direkonstruksi jika kita menyimpan pada setiap $f(i, S)$ nilai j yang meminimumkan ruas kanan persamaan (2).

Misalkan $J(i, S)$ adalah nilai yang dimaksudkan tersebut. Maka, $J(1, \{2, 3, 4\}) = 2$. Jadi, tur mulai dari simpul 1 selanjutnya ke simpul 2.

Simpul berikutnya dapat diperoleh dari $f(2, \{3, 4\})$, yang mana $J(2, \{3, 4\}) = 4$. Jadi, simpul berikutnya adalah simpul 4.

Simpul terakhir dapat diperoleh dari $f(4, \{3\})$, yang mana $J(4, \{3\}) = 3$. Jadi, tur yang optimal adalah 1, 2, 4, 3, 1 dengan bobot (panjang) = 35.

IV. OPTIMASI JALUR BUS DENGAN PROGRAM DINAMIS

Jalur bus pariwisata dapat kita tinjau sebagai salah satu persoalan Traveling Salesperson Problem, yaitu optimasi rute dengan jarak minimum dari lokasi awal mengelilingi semua tujuan dan kembali lagi ke rute asal.

Akan diambil salah satu contoh dari bus pariwisata yang ada di Bandung yaitu bus bandros.

Bandros yang merupakan kepanjangan dari Bandung Tour on The Bus adalah bus tingkat di Kota Bandung yang disediakan oleh pemerintah Kota Bandung bagi wisatawan yang hendak berkeliling kota Bandung. Bus ini diresmikan oleh Walikota Bandung Ridwan Kamil bertepatan dengan malam tahun baru 2014. Bus wisata ini akan melayani para wisatawan di kota Bandung secara gratis. Bandros juga merupakan bentuk Corporate Social Responsibility dari Telkomsel yang bekerjasama dengan pemerintah kota Bandung. Bus berkapasitas kurang lebih 30 orang ini dimaksimalkan di lantai dua, sementara di lantai satu hanya ada 6 kursi bulat, kursi panjang, lalu ada balkon di luar yang bisa dipakai turis untuk berdiri. Bandros ini juga memiliki banyak warna yang berbeda rutenya yaitu merah, biru, kuning, hijau, ungu, dan hitam.

Bus Bandros Ungu memiliki titik tujuan antara lain Alun-alun Ujungberung, Pusdai, Museum Geologi, Gasibu dan Taman Cikapayang. Titik tujuan ini dapat kita anggap sebagai suatu simpul V1, V2, ..., V5 dengan sisi-sisinya dengan bobot yang berbeda.

- V1 : Gasibu
- V2 : Pusdai
- V3 : Alun-alun Ujungberung
- V4 : Museum Geologi
- V5 : Taman Cikapayang

Dengan bobot sisi-sisinya yang melambangkan jaran dapat ditinjau dalam *Adjacency Matrix*.

$$\begin{bmatrix} 0 & 9 & 6 & 7 & 2 \\ 9 & 0 & 7 & 3 & 9 \\ 6 & 7 & 0 & 4 & 2 \\ 7 & 3 & 4 & 0 & 5 \\ 2 & 8 & 2 & 5 & 0 \end{bmatrix}$$

Dengan basis dan rekurens sesuai algoritma Program Dinamis:

$$f(i, \emptyset) = c_{i,1}, \quad 2 \leq i \leq n$$

(basis)

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

(rekurens) (2)

Tahap 1:

$$f(2, \emptyset) = c_{21} = 9;$$

$$f(3, \emptyset) = c_{31} = 6;$$

$$f(4, \emptyset) = c_{41} = 7;$$

$$f(5, \emptyset) = c_{51} = 2;$$

Tahap 2:

$$f(2, \{3\}) = \min\{c_{23} + f(3, \emptyset)\} = \min\{7 + 6\} = \min\{13\} = 13$$

$$f(3, \{2\}) = \min\{c_{32} + f(2, \emptyset)\} = \min\{7 + 9\} = \min\{16\} = 16$$

$$f(4, \{2\}) = \min\{c_{42} + f(2, \emptyset)\} = \min\{3 + 9\} = \min\{12\} = 12$$

$$f(2, \{4\}) = \min\{c_{24} + f(4, \emptyset)\} = \min\{3 + 7\} = \min\{10\} = 10$$

$$f(3, \{4\}) = \min\{c_{34} + f(4, \emptyset)\} = \min\{4 + 7\} = \min\{11\} = 11$$

$$f(4, \{3\}) = \min\{c_{43} + f(3, \emptyset)\} = \min\{4 + 6\} = \min\{10\} = 10$$

$$f(2, \{5\}) = \min\{c_{25} + f(5, \emptyset)\} = \min\{9 + 2\} = \min\{11\} = 11$$

$$f(3, \{5\}) = \min\{c_{35} + f(5, \emptyset)\} = \min\{2 + 2\} = \min\{4\} = 4$$

$$f(4, \{5\}) = \min\{c_{45} + f(5, \emptyset)\} = \min\{5 + 2\} = \min\{7\} = 7$$

$$f(5, \{2\}) = \min\{c_{52} + f(2, \emptyset)\} = \min\{8 + 9\} = \min\{17\} = 17$$

$$f(5, \{3\}) = \min\{c_{53} + f(3, \emptyset)\} = \min\{2 + 6\} = \min\{8\} = 8$$

$$f(5, \{4\}) = \min\{c_{54} + f(4, \emptyset)\} = \min\{5 + 7\} = \min\{12\} = 12$$

Tahap 3:

$$f(i, S) = \min_{j \in S} \{c_{ij} + f(j, S - \{j\})\}$$

untuk $S = 2$ dan $i \neq 1, 1 \notin S$ dan $i \in S$.

Diperoleh:

$$f(2, \{3, 4\}) = \min\{c_{23} + f(3, \{4\}), c_{24} + f(4, \{3\})\}$$

$$= \min\{7 + 11, 3 + 10\}$$

$$= \min\{18, 13\} = 13$$

$$f(2, \{3, 5\}) = \min\{c_{23} + f(3, \{5\}), c_{25} + f(5, \{3\})\}$$

$$= \min\{7 + 4, 12 + 8\}$$

$$= \min\{11, 20\} = 11$$

$$f(2, \{4, 5\}) = \min\{c_{24} + f(4, \{5\}), c_{25} + f(5, \{4\})\}$$

$$= \min\{10 + 7, 12 + 12\}$$

$$= \min\{17, 24\} = 17$$

$$f(3, \{2, 4\}) = \min\{c_{32} + f(2, \{4\}), c_{34} + f(4, \{2\})\}$$

$$= \min\{7 + 10, 4 + 12\}$$

$$= \min\{17, 16\} = 16$$

$$f(3, \{2, 5\}) = \min\{c_{32} + f(2, \{5\}), c_{35} + f(5, \{2\})\}$$

$$= \min\{7 + 11, 2 + 17\}$$

$$= \min\{19, 19\} = 19$$

$$f(3, \{4, 5\}) = \min\{c_{34} + f(4, \{5\}), c_{35} + f(5, \{4\})\}$$

$$= \min\{4 + 7, 5 + 12\}$$

$$= \min\{11, 17\} = 11$$

$$f(4, \{2, 3\}) = \min\{c_{42} + f(2, \{3\}), c_{43} + f(3, \{2\})\}$$

$$= \min\{3 + 13, 4 + 16\}$$

$$= \min\{16, 20\} = 16$$

$$f(4, \{2, 5\}) = \min\{c_{42} + f(2, \{5\}), c_{45} + f(5, \{2\})\}$$

$$= \min\{3 + 11, 5 + 17\}$$

$$= \min\{14, 22\} = 14$$

$$f(4, \{3, 5\}) = \min\{c_{43} + f(3, \{5\}), c_{45} + f(5, \{3\})\}$$

$$= \min\{4 + 4, 5 + 8\}$$

$$= \min\{8, 13\} = 8$$

$$f(5, \{2, 3\}) = \min\{c_{52} + f(2, \{3\}), c_{53} + f(3, \{2\})\}$$

$$= \min\{8 + 13, 2 + 16\}$$

$$= \min\{21, 18\} = 18$$

$$f(5, \{2, 4\}) = \min\{c_{52} + f(2, \{4\}), c_{54} + f(4, \{2\})\}$$

$$= \min\{8 + 10, 5 + 12\}$$

$$= \min\{18, 17\} = 17$$

$$f(5, \{3, 4\}) = \min\{c_{53} + f(3, \{4\}), c_{54} + f(4, \{3\})\}$$

$$= \min\{2 + 11, 5 + 10\}$$

$$= \min\{13, 15\} = 13$$

Tahap 4:

$$f(2, \{3, 4, 5\}) = \min\{c_{23} + f(3, \{4, 5\}), c_{24} + f(4, \{3, 5\}), c_{25} + f(5, \{2, 3\})\}$$

$$= \min\{7 + 11, 3 + 8, 9 + 18\}$$

$$= \min\{18, 11, 27\} = 11$$

$$f(3, \{2, 4, 5\}) = \min\{c_{32} + f(2, \{4, 5\}), c_{34} + f(4, \{2, 5\}), c_{35} + f(5, \{2, 4\})\}$$

$$= \min\{7 + 17, 4 + 14, 2 + 17\}$$

$$= \min\{24, 18, 19\} = 18$$

$$\begin{aligned}
f(4, \{2, 3, 5\}) &= \min\{c_{42} + f(2, \{3, 5\}), c_{43} + f(3, \{2, 5\}), c_{45} + f(5, \\
&\quad \{2, 3\})\} \\
&= \min\{3 + 11, 4 + 19, 5 + 18\} \\
&= \min\{14, 23, 23\} = 14 \\
f(5, \{2, 3, 4\}) &= \min\{c_{52} + f(2, \{3, 4\}), c_{53} + f(3, \{2, 4\}), c_{54} + f(4, \\
&\quad \{2, 3\})\} \\
&= \min\{8 + 13, 2 + 16, 5 + 16\} \\
&= \min\{21, 18, 21\} = 18
\end{aligned}$$

Akhirnya diperoleh :

$$\begin{aligned}
f(1, \{2, 3, 4, 5\}) &= \min\{c_{12} + f(2, \{3, 4, 5\}), c_{13} + f(3, \{2, 4, 5\}), c_{14} + \\
&\quad f(4, \{2, 3, 5\}), c_{15} + f(5, \{2, 3, \\
&\quad 4\})\} \\
&= \min\{9 + 11, 6 + 18, 7 + 14, 2 + 18\} \\
&= \min\{20, 24, 21, 20\} = 20 \\
&=
\end{aligned}$$

Dari tahapan-tahapan program dinamis di atas dapat ditelusuri bahwa jarak minimum untuk rute Bus Bandros Ungu adalah 20 km dengan 2 alternatif rute yaitu:

- 1 → 5 → 3 → 4 → 2 → 1:
Gasibu → Taman Cikapayang → Alun-alun Ujungberung, Museum Geologi → Pusdai → Gasibu
- 1 → 2 → 4 → 3 → 5 → 1:
Gasibu → Pusdai → Museum Geologi → Alun-alun Ujungberung → Taman Cikapayang → Gasibu

Algoritma dari program dinamis di atas dapat diimplementasikan salah satunya dalam bahasa Ruby:

```

def tspSolve(start, rnode, weight)
  if rnode.length == 0
    solution = [weight[start][0], 999]
    return solution
  else
    distance = []
    for i in (0..rnode.length-1)
      newrestnode = []
      copyArray(rnode, newrestnode)
      newstart = rnode[i]
      newrestnode.delete_at(i)
      distance.push(weight[start][newstart] +
tspSolve(newstart, newrestnode, weight)[0])
    end
    min = distance.index(distance.min)
    solution = [distance.min, rnode[min]]
    return solution
  end
end

path = [starting]
finish = false
while !finish
  sol = tspSolve(starting, restnode, weight)
  if starting == first-1
    weightSol = sol[0]
  end

  if sol[1] == 999
    finish = true
  else
    path.push(sol[1])
    starting = sol[1]
    restnode.delete(sol[1])
  end
end
end

```

Hal ini dapat pula diterapkan pada jalur Bus Bandros lainnya untuk mendapatkan rute yang paling optimal sehingga Bus dapat beroperasi dengan efisien dan efektif.

V. KESIMPULAN

Program Dinamis dapat menyelesaikan persoalan optimasi dengan lebih baik dibandingkan dengan algoritma lain seperti Greedy. Dengan program dinamis kita dapat memperoleh banyak rangkaian keputusan yang paling optimal dalam satu waktu.

Pada persoalan rute bus pariwisata misalnya, kita dapat menentukan rute paling optimal dari bus dengan mempertimbangkan faktor jarak antar lokasi. Dengan hal ini, bus pariwisata dapat beroperasi dengan optimal dalam hal biaya operasional dan waktu. Permasalahan ini umum kita jumpai pada rute transportasi lain selain bus pariwisata, sehingga algoritma ini dapat diterapkan dengan sedikit modifikasi untuk menyelesaikan banyak permasalahan optimasi lainnya.

REFERENCES

- [1] Munir, Rinaldi. *Strategi Algoritmik bab Dynamic Programming*. Bandung. Informatika ITB.
- [2] Munir, Rinaldi. *Matematika Diskrit bab Graf*. Bandung. Informatika ITB.
- [3] <https://www.geeksforgeeks.org/travelling-salesman-problem-set-1/>
- [4] <http://industrialengineeringdepartment.blogspot.co.id/2015/04/travelling-salesman-problem-tsp.html>
- [5] <http://www.pikiran-rakyat.com/bandung-raya/2018/01/19/ini-tarif-dan-rute-12-bandros-baru-kota-bandung-ada-bandros-khusus-vip>
- [6] <http://www.galamedianews.com/pariwisata/176483/ini-dia-rute-anyar-wisata-kota-bandung-bus-bandros.html>

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, 14 Mei 2018



Eric Jonathan - 13516117