

# APLIKASI PROGRAM DINAMIS UNTUK PEMILIHAN JALUR TRANSPORTASI JARAK JAUH

Luqman Abdul Mushawwir – NIM:13507029

Institut Teknologi Bandung  
Jalan Ganesha 10 Bandung  
E-mail: [abdulmushawwir@gmail.com](mailto:abdulmushawwir@gmail.com)

## ABSTRAK

Kebutuhan masyarakat akan optimalnya waktu yang ditempuh atau biaya yang digunakan dalam suatu perjalanan jarak jauh sangat tinggi. Namun, terkadang masyarakat tidak mengetahui bagaimana cara mendapatkan alternatif transportasi jarak jauh yang optimal. Salah satu cara yang dapat digunakan untuk mengetahui jalur mana yang memberikan waktu atau biaya yang paling sedikit adalah dengan algoritma program dinamis. Dengan algoritma ini, pengguna sekarang dapat mengetahui angkutan-angkutan yang optimal dari segi biaya atau waktu tempuh.

**Kata Kunci:** Program Dinamis, Optimalisasi, Angkutan umum, Transportasi jarak jauh, Biaya perjalanan, Waktu Perjalanan

## 1. PENDAHULUAN

Zaman sekarang, untuk mencapai satu tempat dari tempat lain seseorang dapat menggunakan banyak sarana transportasi. Mulai dari mobil, sepeda motor, kereta, sampai pesawat terbang. Untuk memilih pun tidak sebatas alat transportasinya saja. Untuk satu jenis alat transportasi saja, ada banyak penyedia jasa transportasi dengan kelebihan dan kekurangannya masing-masing.

Hal-hal yang dipertimbangkan untuk memilih suatu sarana transportasi juga bermacam-macam, namun di sini penulis mengelompokkan menjadi tiga macam: waktu, biaya, dan kenyamanan.

Untuk suatu lingkup lokal (misal: kota Bandung dan sekitarnya), memilih sarana transportasi tersebut mungkin cukup mudah karena selain jalur yang tersedia sedikit, sarana transportasi yang tersedia pun hanya sedikit.

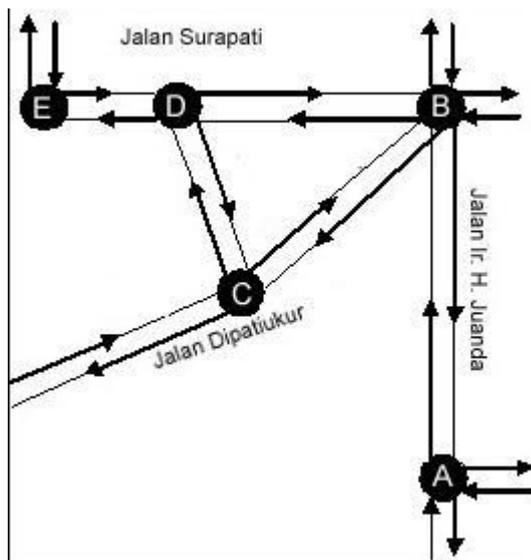
Sebagai contoh, untuk perjalanan pulang penulis dari kampus ITB saja, penulis dapat memilih beberapa angkutan umum seperti berikut:

Tabel 1 Pilihan angkutan umum dari Ganesha 10 menuju Cimahi

Jenis Angkutan	Tempat naik (+jarak)	Biaya	Waktu tempuh	Tempat turun (+jarak)
angkot*** (caringin-sd.serang)	jalan taman sari (300 m)	Rp. 2.000	20 menit	Bandara Husein (6 km)
KRD ekonomi**	Stasiun Bandung (2 km)	Rp. 1.500	35 menit	Stasiun Cimindi (1,5 km)
Bus Damri***	Alun-alun Bandung (2 km)	Rp. 1.500	40 menit	Jalan Pesantren (1 km)
Taksi****	Ganesha 10 (0 m)	Rp. 50.000	30 menit	Rumah (0 m)
Jalan kaki*	Ganesha 10 (0 m)	Rp. 0	150 menit	Rumah (0 m)

Tabel 1 menunjukkan beberapa alternatif angkutan umum yang dapat dinaiki penulis untuk mendekati tempat tujuan, meskipun yang sebenarnya tersedia lebih banyak lagi angkutan yang tersedia, namun tidak dapat ditulis pada tabel 1 (tanda \* menandakan tingkat kenyamanan). Perlu diingat tabel 1 tidak menggambarkan kondisi sebenarnya di lapangan, tetapi hanya perkiraan penulis.

Tempat-tempat pada suatu area dapat digambarkan dengan titik-titik dan garis, seperti pada gambar berikut:



**Gambar 1. Penggambaran jalan-jalan di Bandung menggunakan titik dan garis**

Dalam pemecahan masalah ini, diibaratkan seseorang akan menuju suatu titik dari titik lain melewati garis-garis yang tersedia.

Untuk makalah ini, masalah yang penulis bicarakan terbatas pada kondisi standar dan untuk daerah-daerah yang diketahui di bumi. Kondisi standar maksudnya kondisi biaya, waktu tempuh, dan tingkat kenyamanan pada kondisi sehari-hari, tidak ada kasus khusus (seperti kecelakaan atau bencana alam).

Contoh-contoh yang diambil dalam makalah ini juga mungkin bukan kondisi aktual, beberapa hanya perkiraan penulis, tetapi cukup menggambarkan kondisi yang sebenarnya (misal: jarak Jl. Ganesha-Jl. Tamansari tentu saja lebih dekat daripada jarak Jl. Ganesha-Jl. Dipati Ukur).

Dalam makalah ini penulis ingin mencari bagaimana cara seseorang menemukan jalan yang optimal dari suatu tempat ke tempat lain, baik dari segi waktu, biaya, tingkat kenyamanan, ataupun kombinasi dari ketiga aspek itu.

## 2. ALGORITMA PROGRAM DINAMIS

Program dinamis merupakan suatu cara penyelesaian masalah dengan menguraikan solusi menjadi sekumpulan langkah atau tahapan sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Pada penyelesaian metode ini kita menggunakan persyaratan optimasi dan kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan pada suatu tahap.

Program dinamis mempunyai karakteristik sebagai berikut:

1. Persoalan dapat dibagi menjadi beberapa tahap, yang pada setiap 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. Jumlah status bisa berhingga atau tidak berhingga.
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  $k+1$ .

Secara umum, ada empat langkah yang dilakukan dalam mengembangkan algoritma program dinamis:

1. Karakteristikkan struktur solusi optimal.
2. Definisikan secara rekursif nilai solusi optimal.
3. Hitung nilai solusi optimal.
4. Konstruksi solusi optimal.

Dalam menyelesaikan persoalan dengan program dinamis, orang dapat menggunakan dua pendekatan berbeda, yaitu maju (*forward* atau *up-down*) dan mundur (*backward* atau *bottom-up*). Misalkan  $x_1, x_2, x_3, \dots, x_n$  menyatakan peubah keputusan yang harus dibuat masing-masing untuk tahap 1, 2, 3, ..., n, maka,

- a. Program dinamis maju. Program dinamis bergerak mulai dari tahap 1, terus maju ke tahap 2, 3, dan seterusnya sampai tahap n. Runtunan peubah keputusan adalah  $x_1, x_2, x_3, \dots, x_n$ .
- b. Program dinamis mundur. Program dinamis bergerak mulai dari tahap n, terus mundur ke tahap n-1, n-2, sampai ke tahap 1. Runtunan peubah keputusan adalah  $x_n, x_{n-1}, x_{n-2}, \dots, x_1$ .

## 3. METODE PENYELESAIAN MASALAH

Seperti yang telah dipaparkan sebelumnya, masalah pencarian jalur transportasi jarak jauh terbaik ini akan diselesaikan menggunakan algoritma program dinamis yang sudah dijelaskan pada bab sebelumnya.

Untuk mempermudah pemecahan masalah, pendekatan yang dipakai adalah program dinamik mundur, dengan dua pemecahan masalah yang berbeda: optimalisasi waktu dan optimalisasi biaya.

Sebagai permulaan, kita menentukan titik awal dan titik akhir perjalanan yang akan ditempuh. Kita anggap awal

perjalanan ( $x_0$ ) adalah kampus ITB, yaitu Jalan Ganesha 10, sedangkan akhir perjalanan ( $x_n$ ) adalah Bandara Soekarno-Hatta Jakarta.

Dari titik awal dan titik akhir, kita membuat daftar angkutan yang berada di antara  $x_0$  dan  $x_n$ , dengan spesifikasi seperti yang tertera pada tabel 1.

**Tabel 2 angkutan yang berada di antara  $x_0$  dan  $x_n$**

Jenis Angkutan	Jarak tempat naik ( $d_1$ )	Biaya	Waktu tempuh	Jarak tempat turun ( $d_2$ )
angkot (caringin-sd.serang)	100 m	Rp. 600 * km jarak	10 menit * km jarak	125-120 km
angkot (cicaheum-ciroyom)	100 m	Rp. 600 * km jarak	10 menit * km jarak	124-120 km
amgkot (kalapadago)	60 m	Rp. 600 * km jarak	10 menit * km jarak	128-123 km
Kereta Bisnis	2 km	Rp. 20.000	240 menit	35 km
Kereta Eksekutif	2 km	Rp. 35.000	220 menit	35 km
Kereta Argo	2 km	Rp. 45.000	200 menit	35 km
Bus Damri	2 km	Rp. 1.500	40 menit	1 km
Travel 1	3 km	Rp. 50.000	240 menit	25 km
Travel 2	2.5 km	Rp. 35.000	220 menit	40 km

Pada tabel di atas,  $d_1$  dihitung dari  $x_0$  dan  $d_2$  dihitung dari  $x_n$ . Jumlah angkutan umum dibatasi, karena akan sangat banyak jika ditulis semua. Aspek kenyamanan diabaikan terlebih dahulu.

Setelah semua kemungkinan transport dibuat, asumsikan manusia hanya sanggup berjalan sampai 500 m untuk setiap pemberhentian, sehingga setiap transport yang mempunyai  $d_1$  lebih dari 500 m dieliminasi, saat menentukan solusi lokal.

Tahap yang digunakan dalam fungsi ini mungkin saja sangat banyak sampai tak berhingga, sehingga tahap untuk fungsi ini harus dibatasi, karena akan mempermudah kerja memori yang harus membangkitkan data-data transportasi setiap kali rekursif.

### 3.1 Optimalisasi Waktu Perjalanan

Menentukan solusi untuk optimalisasi waktu perjalanan merupakan sebuah operasi rekursif dengan notasi sebagai berikut:

$$f_n(s) = t_{xn} \quad (\text{basis})$$

$$f_k(s) = \min \{t_{xn} + f_{k+kx}(x_k)\} \quad (\text{rekurens})$$

Sedangkan untuk setiap tahap yang dilakukan dilakukan pembangkitan daftar angkutan yang tersedia (sama seperti tabel 2)

Jika operasi rekursif didefinisikan sebagai `optimizeTime(k)` dan operasi pembangkitan data didefinisikan sebagai `generate(k)` maka fungsi `optimizeTime(k)` adalah sebagai berikut:

```
Generate(k)
if (k <= 500) then {basis}
    return (t(x)+walk(k))
else {rekurens}
    return (min(t(x)+walk(k-k(x))
+optimizeTime(k-k(x))))
```

Pada notasi di atas,  $k$  adalah jarak dari  $x$  (tempat tersebut) sampai tempat akhir ( $x_n$ ). Fungsi tersebut akan mencapai basisnya saat  $k$  kurang dari atau sama dengan 500 m, yaitu anggapan manusia tidak butuh alat transportasi, namun waktu berjalannya juga ditambahkan pada waktu total.

### 3.2 Optimalisasi Biaya Perjalanan

Optimalisasi biaya perjalanan pada prinsipnya hampir sama dengan optimalisasi waktu perjalanan, namun semua operasi yang berhubungan dengan waktu diganti dengan biaya.

Menentukan solusi untuk optimalisasi biaya perjalanan merupakan sebuah operasi rekursif dengan notasi sebagai berikut:

$$f_n(s) = c_{xn} \quad (\text{basis})$$

$$f_k(s) = \min \{c_{xn} + f_{k+kx}(x_k)\} \quad (\text{rekurens})$$

Sedangkan untuk setiap tahap yang dilakukan dilakukan pembangkitan daftar angkutan yang tersedia (sama seperti tabel 2)

Jika operasi rekursif didefinisikan sebagai `optimizeCost(k)` dan operasi pembangkitan data didefinisikan sebagai `generate(k)` maka fungsi `optimizeCost(k)` adalah sebagai berikut:

```
Generate(k)
if (k <= 500) then {basis}
    return (c(x))
```

```
else {rekurens}
    return (min(c(x)+optimizeCost(k-
k(x))))
```

Pada notasi di atas,  $k$  adalah jarak dari  $x$  (tempat tersebut) sampai tempat akhir ( $x_n$ ). Fungsi tersebut akan mencapai basisnya saat  $k$  kurang dari atau sama dengan 500 m, yaitu anggapan manusia tidak butuh alat transportasi. Pada bagian ini, waktu untuk berjalan sama sekali tidak diperhitungkan, karena murni menghitung biaya perjalanan saja.

#### **4. KESIMPULAN**

Algoritma program dinamis sudah lumrah dipakai untuk suatu persoalan optimalisasi. Dalam kasus ini, algoritma program dinamis sudah cukup dapat menyelesaikan persoalan pemilihan jalur transportasi jarak jauh, namun yang perlu diperhatikan adalah banyaknya jumlah dan jenis angkutan yang sangat banyak dari suatu tempat ke tempat lain di dunia.

Untuk optimalisasi biaya atau waktu saja, hanya satu aspek yang dilihat (biaya atau waktu), namun jika melihat kedua aspek tersebut bersamaan, ditambah aspek kenyamanan, algoritma yang dibuat akan lebih kompleks lagi.

#### **REFERENSI**

- [1] <http://kereta-api.co.id/> waktu akses 6 Januari 2010, 00.13 WIB
- [2] Rinaldi Munir, "Diktat Kuliah IF3051 Strategi Algoritma", Informatika, 2009