

# Pencarian Lintasan Terpendek Jalur Pendakian Gunung dengan Program Dinamis

Mochamad Lutfi Fadlan<sup>1</sup>

13512087

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

<sup>1</sup>13512087@std.stei.itb.ac.id

**Abstract**—Mendaki gunung adalah salah satu hobi yang cukup digemari oleh segelintir orang. Mendaki gunung sekarang ini tidak terlalu sulit karena memang sudah terdapat dan diketahui jalur yang bisa dilalui untuk mencapai puncak gunung yang ingin dicapai. Biasanya, jalur yang dilalui adalah jalur darat dengan berbagai alternatif. Terdapat beberapa pos di sekitar gunung yang bisa digunakan untuk beristirahat, mempersiapkan keberangkatan, dan tempat berkumpul para pendaki. Dari pos-pos itulah kita bisa menentukan jalur mana yang ingin kita lalui berdasarkan jarak dari tempat asal ke tempat tujuan yang diketahui. Dari jarak tersebut kita bisa mendapatkan lintasan terpendek untuk jalur pendakian dengan menggunakan program dinamis.

**Index Terms**—program dinamis, lintasan terpendek, jarak, graf

## I. PENDAHULUAN

Pendakian gunung adalah kegiatan berjalan kaki di daerah pegunungan yang biasanya bertujuan untuk mencapai puncak gunung tersebut. Kegiatan ini biasanya dilakukan oleh anak muda, dewasa, dan bahkan orang tua. Biasanya kebanyakan orang lebih ingin mencapai puncak dengan jalur yang lebih pendek karena jika jalur yang diambil lebih banyak akan menguras tenaga dan waktu. Dengan diketahui jarak dari pos-pos tertentu sampai tempat yang ingin dituju bisa ditentukan terlebih dahulu mana jalur yang lebih baik diambil.

Jika sudah diketahui medan lapangannya (jarak antar tempat) maka dapat dibuat jalurnya dengan menggunakan program dinamis. Dengan memanfaatkan program dinamis maka akan didapatkan lintasan terpendek dari tempat asal menuju puncak gunung sehingga kita bisa mendaki dengan menghemat waktu dan energi.

## II. LANDASAN TEORI

### Graf

Graf adalah sebuah kumpulan atau himpunan tidak kosong yang terdiri dari simpul (*vertex*) yang terhubung dengan sisi (*edge*). Sederhananya, graf itu adalah

kumpulan titik yang mana suatu titik dihubungkan dengan titik yang lainnya dengan menggunakan garis atau busur.

Formulasi untuk graf adalah  $G = (V, E)$  dengan  $G$  adalah graf,  $V$  adalah *vertex*, yaitu himpunan yang tidak kosong dari simpul, dan  $E$  adalah *Edge*, yaitu himpunan sisi yang menghubungkan satu simpul dengan satu simpul lainnya.

Graf bisa digunakan untuk menganalisis berbagai permasalahan umum yang ada. Graf sering dipakai untuk penyelesaian masalah dalam bidang matematika, komputer, salah satunya yaitu jaringan komputer, dan masih banyak bidang lainnya. Aplikasinya lainnya yang dibahas dalam kuliah matematika diskrit adalah Perancangan IC (*integrated Circuit*), Lintasan terpendek (*shortest path*), Persoalan pedagang keliling (*travelling sales person problem*), persoalan tukang pos Cina (*chinese postman problem*), dan pewarnaan graf (*graph colouring*).

Berdasarkan orientasi arah pada sisinya secara umum, graf dibedakan menjadi dua jenis, yaitu graf tidak berarah (*undirected graph*) dan graf berarah (*directed graf*). Pada pembahasan kali ini akan digunakan graf berarah yang memiliki bobot, yang artinya pada masing-masing sisinya diberikan sebuah bobot / nilai.

### Program Dinamis

Program dinamis adalah sebuah cara pemecahan suatu masalah dengan mendekomposisi solusi menjadi sekumpulan tahapan sehingga solusi yang didapatkan merupakan seluruh rangkaian keputusan yang saling memiliki keterhubungan di mana penentuan pengambilan keputusannya memaksimalkan seluruh keefektifannya. Istilah program dinamis bisa muncul disebabkan adanya penggunaan tabel dalam memperhitungkan solusi. Tujuan dari program dinamis adalah memudahkan penyelesaian persoalan optimasi yang memiliki karakteristik khusus dan mencari solusi dengan penentuan keputusan yang optimal.

Persoalan masalah yang dapat diselesaikan dengan menggunakan program dinamis adalah persoalan yang memenuhi tiga sifat dibawah ini:

1. Ada sejumlah pilihan berhingga yang dapat dipilih.

2. Solusi pada tiap tahap saling berkaitan.
3. Adanya syarat optimasi dalam penentuan pilihan yang ada.

Ciri-ciri khusus dari program dinamis adalah terdapat lebih dari satu rangkaian keputusan yang digunakan dalam pembuatan solusi. Selain itu terdapat prinsip optimalitas yang digunakan dalam pembuatan rangkaian keputusan yang optimal. Maksud prinsip optimalitas di sini adalah jika seluruh solusi optimal maka terdapat salah satu solusi pada tahap tertentu yang optimal juga. Artinya, jika ingin mendapatkan solusi pada salah satu tahap maka bisa dilihat solusi pada tahap sebelumnya saja tanpa harus kembali melihat solusi pada tahap awal.

Karakteristik yang lebih rinci dari program dinamis:

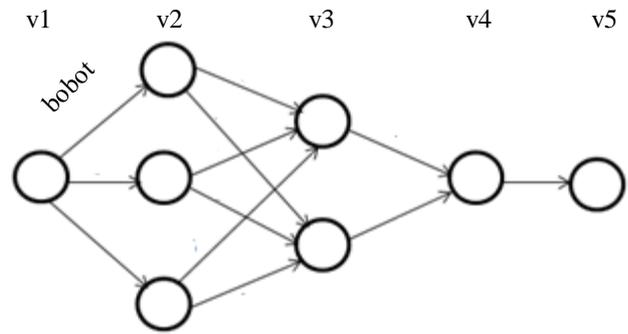
1. Persoalan dipecah / dibagi menjadi sekumpulan tahap yang mana pada masing-masing tahap tersebut diambil satu keputusan di mana keputusan tersebut tidak bergantung pada tahap sebelumnya dan terdapat hubungan rekursif pada keputusan yang diambil tiap status pada suatu tahap yang memberikan keputusan yang diambil tiap status pada tahap yang selanjutnya.
2. Pada setiap tahap terdapat status yang berisi kemungkinan masukan pada tahap itu.
3. Hasil keputusan pada setiap tahap ditransformasi.
4. *Cost* pada satu tahap mengingkat dengan teratur ketika jumlah tahapan bertambah dan bergantung pada *cost* yang telah berjalan dan *cost* pada tahap itu.
5. Diberlakukannya prinsip optimalitas ada persoalan.

Terdapat dua pendekatan program dinamis, yaitu maju (dimulai dari tahap ke-1 sampai ke-n) dan mundur (dimulai dari tahap ke-n sampai ke-1). Adapun langkah-langkah yang digunakan dalam pembuatan program dinamis, yaitu

1. Struktur solusi yang optimal dikarakteristikan.
2. Nilai solusi yang optimal didefinisikan.
3. Nilai solusi yang optimal dihitung sesuai dengan pendekatan yang digunakan.
4. Solusi optimal dikonstruksikan.

Pos-pos yang ada dijadikan simpul dan jika terdapat lebih dari satu pos yang berdekatan pada satu daerah maka kumpulan pos tersebut dijadikan satu tahap. Dan jika tidak ada lagi pos lain yang berdekatan dengan suatu pos maka pos tersebut dijadikan simpul sekaligus satu tahap.

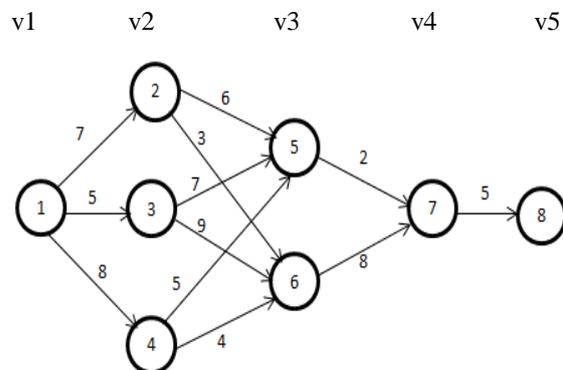
Awal pencarian dimulai dari kaki gunung dilanjutkan dengan pos-pos terus sampai diakhiri dengan puncak gunung. Masing-masing tahap merupakan pencapaian pos-pos yang berkaitan pada daerah tertentu dengan terdapatnya keputusan optimal yang telah diambil pada tahap tersebut. Kasus ini dapat digambarkan seperti di bawah ini.



Pada gambar di atas menunjukkan jalur pendakian yang berasal dari simpul ke-1 (kaki gunung) menuju ke simpul ke-7 (puncak gunung). Di sini terdapat v1, v2, v3, v4, dan v5 yang merupakan representasi dari tahap ke-1, tahap ke-2, dan seterusnya sampai dengan tahap ke-5. Lalu, terdapat simpul-simpul yang merepresentasikan pos-pos yang terdapat pada setiap tahap(daerah). Arah dari satu simpul menuju simpul lain merepresentasikan perjalanan yang dilalui oleh pendaki. Bobot dari arah merepresentasikan jarak dari satu simpul ke simpul yang lain. Pada persoalan ini, tahap adalah proses pemilihan simpul pada tujuan yang berikutnya, yang berarti ada tujuan yang berhubungan dengan masing-masing tahap adalah simpul-simpul yang ada di dalam graf. Dengan tahap mundur dapat dicari lintasan terpendek dari graf berarah yang memiliki bobot tersebut dengan penyelesaian secara program dinamis.

### III. ANALISIS

Pada pembahasan ini akan diambil salah satu sampel yang dapat merepresentasikan pencarian jalur lintasan terpendek pendakian gunung dengan menggunakan program dinamis. Sampel yang diambil adalah gambar yang terdapat di bawah ini.



#### Penyelesaian persoalan dengan pendekatan mundur

Algoritma yang digunakan kurang lebih seperti di bawah ini.

1. Bagi persoalan menjadi beberapa tahap
2. Tentukan status, yaitu kemungkinan keputusan

optimal.

3. Buat simpul pada setiap tahap berdasarkan statusnya.
4. Buat sisi pada simpul sehingga simpul menjadi graf berarah
5. Berikan bobot pada setiap sisi sehingga graf menjadi graf berarah yang berbobot.
6. Hitung nilai cost dari simpul pertama berdasarkan bobot yang telah ada.
7. Hitung nilai cost pada simpul selanjutnya dengan sambil dipilih solusi optimal pada tiap tahap dengan mempertimbangkan nilai cost yang minimum dan setiap penghitungan cost selalu dipengaruhi oleh keputusan pada tahap sebelumnya.
8. Semua keputusan optimal telah didapatkan sehingga solusi pada tiap tahap didapatkan dan didapatkan solusi totalnya yaitu berupa jalur lintasan terpendek.

Dimisalkan di sini  $x_1, x_2, x_3$ , dan  $x_4$  adalah simpul-simpul yang dikunjungi pada tahap  $k$ , di mana  $k=1, k=2, k=3$ , dan  $k=4$ . Dengan begitu, jalur yang dilalui adalah simpul 1,  $x_2, x_3, x_4$ , dan sampai pada simpul 8 secara berurutan. Di sini tahap( $k$ ) merupakan proses pemilihan simpul pada tujuan yang berikutnya (terdapat empat tahap). Kemudian, status( $s$ ) yang berhubungan dengan tiap tahap merupakan simpul-simpul yang berada di dalam graf.

Dalam persoalan pencarian jalur pendakian ini,  $x_1, x_2, x_3$ , dan  $x_4$  mewakili daerah yang di mana terdapat pos-pos yang dapat disinggahi oleh para pendaki. Simpul-simpul yang terdapat pada setiap tahap mewakili pos-pos yang dapat dilalui oleh para pendaki.

Relasi rekurens di bawah ini menunjukkan lintasan terpendek dari status  $s$  menuju simpul  $x_4$  pada tahap  $k$ .

$$f_1(s) = c_{x_1s} \text{ (basis)}$$

$$f_k(s) = \min_{x_k} \{c_{x_k s} + f_{k-1}(x_k)\} \text{ (rekurens)}$$

$$k = 2, 3, 4$$

Keterangan

- a.  $x_k$  : peubah keputusan pada tahap  $k$  ( $k = 2, 3, 4$ ).
- b.  $C_{sx_k}$  : bobot (*cost*) sisi dari  $s$  ke  $x_k$
- c.  $f_k(s)$  : nilai minimum dari  $f_k(x_k, s)$
- d.  $f_k(x_k, s)$  : total bobot lintasan dari ke  $x_k$  ke  $s$

Tujuan dari program dinamis mundur adalah mendapatkan  $f_4(8)$  dengan cara mencari  $f_1(s), f_2(s), f_3(s)$  terlebih dahulu.

Tahap 1

$$f_1(s) = c_{x_1s}$$

s	Solusi Optimum	
	$f_1(s)$	$x_1^*$
2	7	1
3	5	1
4	8	1

$x_k^*$  adalah nilai  $x_k$  yang meminimumkan  $f_s$ .

Tahap 2:

$$f_2(s) = \min_{x_2} \{c_{x_2s} + f_1(x_2)\}$$

$x_2 \backslash s$	$f_2(x_2, s) = c_{x_2s} + f_1(x_2)$			Solusi Optimum	
	2	3	4	$f_2(s)$	$x_2^*$
5	13	12	13	12	3
6	10	14	12	10	2

Tahap 3

$$f_3(s) = \min_{x_3} \{c_{x_3s} + f_2(x_3)\}$$

$x_3 \backslash s$	$f_3(x_3, s) = c_{x_3s} + f_2(x_3)$		Solusi Optimum	
	5	6	$f_3(s)$	$x_3^*$
7	14	18	14	5

Tahap 4

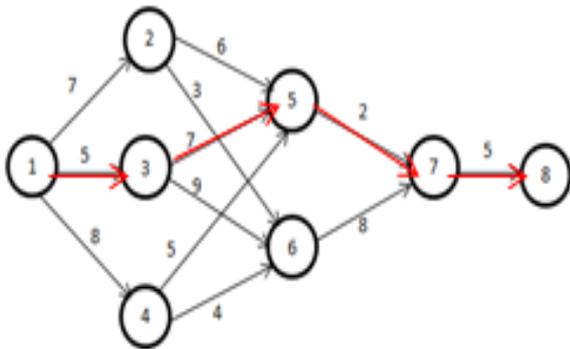
$$f_4(s) = \min_{x_4} \{c_{x_4s} + f_3(x_4)\}$$

$x_4 \backslash s$	$f_4(x_4, s) = c_{x_4s} + f_3(x_4)$		Solusi Optimum	
	7	8	$f_4(s)$	$x_4^*$
8	19	19	19	7

Sehingga solusi optimumnya :

	$x_4$	$x_3$	$x_2$	$x_1$	Panjang Lintasan Terpendek
	8 ←	7 ←	5 ←	3 ← 1	19

Jadi didapatkan satu lintasan terpendek dari jalur pendakian, yaitu



$1 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8$

Dengan panjang lintasannya 19.

### Penyelesaian persoalan dengan pendekatan maju

Dengan pendekatan ini cara-cara yang digunakan di atas dan hasil perhitungan solusi dan keputusannya pun sama. Tujuan dari program dinamis mundur adalah mendapatkan  $f_1(8)$  dengan cara mencari  $f_2(s)$ ,  $f_3(s)$ , dan  $f_4(s)$  terlebih dahulu. Hasil yang didapatkan jika digunakan pencarian lintasan terpendek dengan menggunakan program dinamis pendekatan maju dengan solusi optimumnya adalah

	$x_4$	$x_3$	$x_2$	$x_1$	Panjang Lintasan Terpendek
	1 →	3 →	5 →	7 → 8	19

### Kelebihan dan kekurangan

Cara pencarian lintasan terpendek ini bisa dilakukan jika diketahui dengan pasti jarak antara satu pos di suatu tahap dengan pos pada tahap yang lain. Jika tidak diketahui

jarak, mungkin waktu yang ditempuh untuk mencapai pos lain pada tahap selanjutnya dari pos pada suatu tahap bisa dijadikan pertimbangan untuk pencarian lintasan terpendek, tetapi hal ini tidak disarankan disebabkan waktu yang ditempuh setiap pendaki berbeda (bergantung pada kecepatan berjalannya).

Dengan menggunakan program dinamis, ketika dibutuhkan solusi pada suatu tahap, tidak perlu lagi melihat solusi pada tahap awal, melainkan cukup hanya dengan melihat pada solusi pada tahap sebelumnya karena solusi pada setiap tahap yang berurutan pada program dinamis saling berkaitan sehingga jelas dalam pencarian solusi dari suatu sub masalah yang merupakan hasil dekomposisi dari masalah yang besar lebih efektif dibandingkan dengan tanpa menggunakan program dinamis.

Pada persoalan ini, jika kita ingin mengetahui jarak suatu pos dari kaki gunung (simpul awal) akan lebih mudah didapatkan karena tinggal melihat saja jarak yang merupakan solusi pada daerah (tahap) selanjutnya sehingga tidak perlu menelusuri dari kaki gunung lagi untuk mendapatkan jarak pada pos tersebut.

Kelemahan penggunaan program dinamis pada persoalan ini adalah jika terdapat data jarak yang tidak valid sehingga solusi yang ditemukan bukan merupakan lintasan terpendek, bahkan lebih buruk lagi jika tidak diketahui data jaraknya. Hal ini bisa menyebabkan tidak berjalannya program dinamis karena komponen jarak (cost) merupakan komponen yang terpenting dalam pembuatan keputusan dan penemuan solusi yang optimum.

Berikut ini adalah contoh jalur pendakian gunung merbabu yang memiliki data jarak yang bisa diperhitungkan dalam pencarian lintasan terpendek dari kaki gunung merbabu sampai puncaknya.

Jarak & Waktu Tempuh			
BASE CAMP	→	CAMP 1 PENDING	1200M 1 JAM
CAMP 1	→	CAMP 2 PERENG PUTIH	1000M 1 JAM
CAMP 2	→	CAMP 3 GUMUK MENTUL	575 M 30 MENIT
CAMP 3	→	CAMP 4 LEMPONG SAMPAN	785 M 1 JAM
CAMP 4	→	PUNCAK I WATU GUBUG	724 M 1 JAM
PUNCAK I	→	PUNCAK II WATU TULIS	435 M 45 MENIT
PUNCAK II	→	PUNCAK III CEGER SAPI	700 M 45 MENIT
PUNCAK III	→	PERTIGAAN	557 M 15 MENIT
PERTIGAAN	→	PUNCAK IV SYARIF	130 M 10 MENIT
PERTIGAAN	→	PUNCAK V ONDO RANTE	143 M 15 MENIT
PUNCAK V	→	PUNCAK VI KENTHENG SONGO	300 M 30 MENIT
PUNCAK VI	→	PUNCAK VII TRIANGGULASI	75 M 5 MENIT
TOTAL JELAJAH SEVEN SUMMIT 6674 M 7 JAM 20 MENIT			

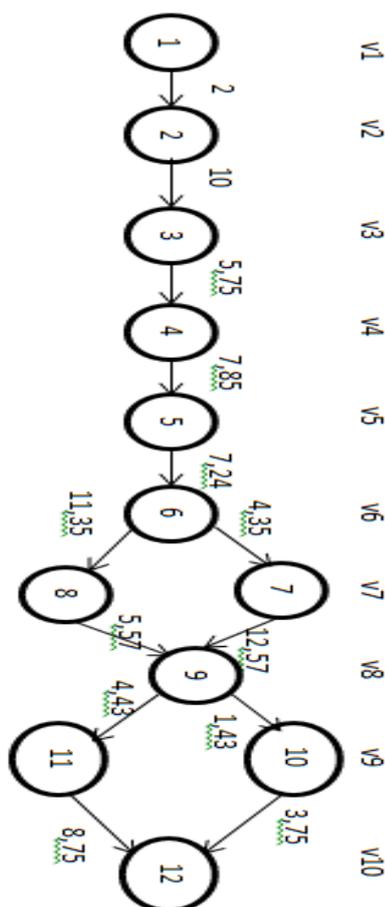
Sumber:

[http://mangkatnekat.files.wordpress.com/2012/01/383921\\_1713987466532\\_1743560892\\_925690\\_544250897\\_n.jpg](http://mangkatnekat.files.wordpress.com/2012/01/383921_1713987466532_1743560892_925690_544250897_n.jpg)

Setelah diketahui data jarak di atas maka dari setiap tempat yang diketahui jaraknya dari tempat sebelumnya dapat dibuat simpul-simpul kemudian diperhitungkan jarak antar tempatnya, jika terdapat lebih dari satu jarak antar tempat yang kurang dari 500 meter maka tempat-tempat tersebut dapat dijadikan satu daerah (satu tahap). Lalu yang dijadikan simpul pertama di sini adalah simpul base camp, dan simpul akhirnya adalah puncak VII Trianggulasi. Dan, tempat yang lain dijadikan simpul dengan terbagi menjadi 10 tahap dengan asumsi pendaki tidak mencapai puncak IV karena setelah dari puncak IV tidak diketahui data jarak menuju puncak VI. Berikut ini representasi dari graf berarah dari jalur pendakian gunung Merbabu.

Keterangan:

- Base camp = simpul 1
- Camp 1 = simpul 2
- Camp 2 = simpul 3
- Camp 3 = simpul 4
- Camp 4 = simpul 5
- Puncak I = simpul 6
- Puncak II = simpul 7
- Puncak III = simpul 8
- Pertigaan = simpul 9
- Puncak V = simpul 10
- Puncak VI = simpul 11
- Puncak VII = simpul 12



Dari graf berarah di atas dapat dicari lintasan pendek dari lintasan satu menuju lintasan dua belas dengan menggunakan program dinamis baik itu dengan pendekatan maju maupun pendekatan mundur. Dengan mengikuti cara seperti pada sampel maka bisa didapatkan solusi yang optimal, yaitu jalur lintasan terpendek yang sesuai dengan perhitungan cost minimum dari tiap tahap dan dengan pengambilan keputusan yang optimal pada tiap tahap.

Ini baru salah satu contoh gunung yang diperlihatkan di sini. Masih banyak berbagai gunung lain yang memiliki rute yang beragam, entah itu jalurnya memiliki banyak alternatif atau pun hanya memiliki satu jalur saja yang pendakian. Jika terdapat banyak jalur alternatif maka penggunaan program dinamis dalam pencarian jalur lintasan terpendek akan sangat berguna karena akan banyak kemungkinan keputusan yang diambil pada setiap jalan yang dilewati.

### Kesulitan

Tidak mudah dalam mencari data jarak pada jalur pendakian gunung karena memang biasanya hanya terdapat satu jalur saja yang biasa didaki oleh para pendaki gunung sehingga tidak ada alternatif lain dalam jalur pendakian gunung, dan dengan begitu biasanya tidak terlalu diperlukan program dinamis dalam pencarian rutenya karena hanya memang terdapat satu jalur.

## IV. KESIMPULAN

Dengan menggunakan program dinamis dapat dicari lintasan terpendek dari jalur pendakian suatu gunung dengan syarat terpenuhinya semua data jarak sehingga program dinamis dapat berjalan dengan semestinya.

### DAFTAR PUSTAKA

- [1] Ir. Rinaldi Munir, M.T. *Diktat Kuliah IF2211 Strategi Algoritma*. Bandung:ITB, 2008.
- [2] Ir. Rinaldi Munir, M.T. *Struktur Diskrit*. Bandung: ITB, 2008.
- [3] <http://gangsarnovianto.blogspot.com/2014/01/teori-program-dinamis.html>, diakses 19 Mei 2014, 21:00

### 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, 18 Mei 2013

A handwritten signature in black ink, appearing to read 'Mochamad Lutfi Fadlan', with a stylized flourish at the end.

Mochamad Lutfi Fadlan  
13512087