

PEMODELAN GRAF: NETWORKS

Algoritma Dijkstra - Floyd

C. Kuntoro Adi, SJ
Universitas Sanata Dharma
MATEMATIKA DISKRIT 2020/2021

Referensi:

Rosen, 2019, Discrete Mathematics and Its Applications, Bab 11
Taha, 2017, Operation Research an Introduction, Bab 6

1

Pokok Bahasan

1. Pengantar
2. Minimal spanning tree
3. Route terpendek: Dijkstra
4. Route terpendek: Floyd

2

2. ROUTE TERPENDEK: ALGORITMA DIJKSTRA

Ada dua model dalam menentukan jarak terpendek

1. Dijkstra Algorithm: untuk menentukan jarak terpendek suatu sumber (asal) terhadap node lain yang ada dalam jaringan
2. Floyd Algorithm: dipergunakan untuk menentukan jarak atau route terpendek setiap node dalam suatu jaringan

3

Algoritma Dijkstra

Catatan awal

- Misalkan u_i = jarak terpendek suatu node ke node i
- $d_{ij} (\geq 0)$ = jarak antara node i ke node j
- Label $[u_j, i] = [u_i + d_{ij}, i]$
- Dibedakan: label permanen, label temporer

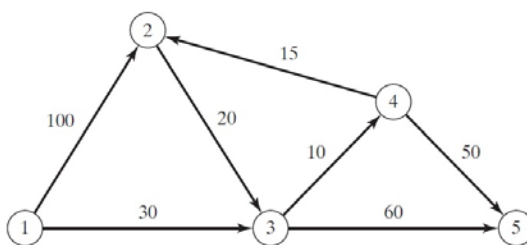
4

Langkah Algoritma

1. Beri label node sumber dengan label permanen $[0, -]$, set $i=1$
2. Langkah i :
 1. Tentukan label temporer $[u_i + d_{ij}, i]$ untuk setiap node j yang bisa disambungkan dengan i
 2. Jika node j sudah berlabel $[u_j, k]$ lewat node k , dan jika $u_i + d_{ij} < u_j$, ganti label $[u_j, k]$ dengan $[u_i + d_{ij}, i]$
 3. Jika semua node telah memiliki label permanen, stop. Otherwise: pilih label $[u_r, s]$ dengan jarak terpendek dari semua label temporer. Set $i=r$; ulang lagi langkah i

5

Contoh



(Taha, 2017)

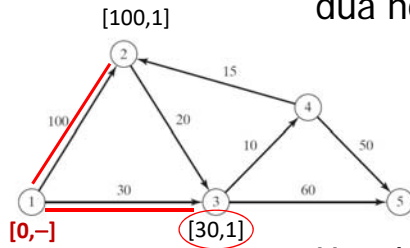
Jaringan di Gambar menunjukkan jarak (satuan kilometer) beberapa kota. Temukan jarak terpendek antara kota 1 dengan 4 kota lainnya.

6

How?

- Iterasi 0 Assign node 1 dengan label permanen $[0, -]$
- Iterasi 1

Node 2 dan 3 terhubung ke node 1. Label untuk dua node ini akan menjadi sebagai berikut:



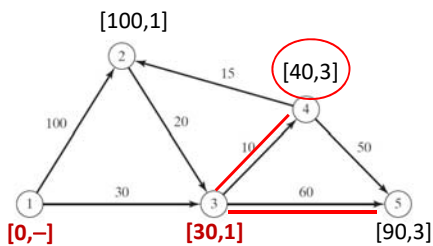
Node	Label	Status
1	$[0, -]$	Permanent
2	$[0 + 100, 1] = [100, 1]$	Temporary
3	$[0 + 30, 1] = [30, 1]$	Temporary

Untuk dua label $[100, 1]$ dan $[30, 1]$, node 3 memiliki jarak lebih pendek ($u_3 = 30$), oleh karena itu labelnya menjadi permanen

7

- Iterasi 2

Node 4 dan 5 dapat terhubung lewat node 3. Label pada node akan menjadi sebagai berikut



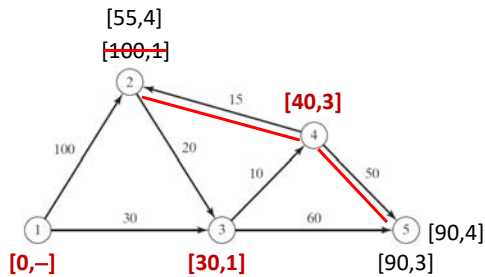
Node	Label	Status
1	$[0, -]$	Permanent
2	$[100, 1]$	Temporary
3	$[30, 1]$	Permanent
4	$[30 + 10, 3] = [40, 3]$	Temporary
5	$[30 + 60, 3] = [90, 3]$	Temporary

Label temporer $[40, 3]$ di node 4 akan menjadi label permanen ($u_4 = 40$)

8

• Iterasi 3

Node 2 dan 5 dapat terhubung lewat node 4. Label pada node diupdate menjadi sebagai berikut



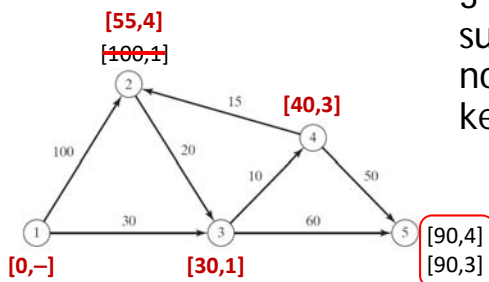
Node	Label	Status
1	$[0, -]$	Permanent
2	$[40 + 15, 4] = [55, 4]$	Temporary
3	$[30, 1]$	Permanent
4	$[40, 3]$	Permanent
5	$[90, 3]$ or $[40 + 50, 4] = [90, 4]$	Temporary

Di node 2, label baru $[55, 4]$ menggantikan label temporer $[100, 1]$ karena memberi route lebih pendek. Node 5 memiliki dua label. Label di node 2 $[55, 4]$ berubah menjadi label permanen ($u_2 = 55$)

9

• Iterasi 4

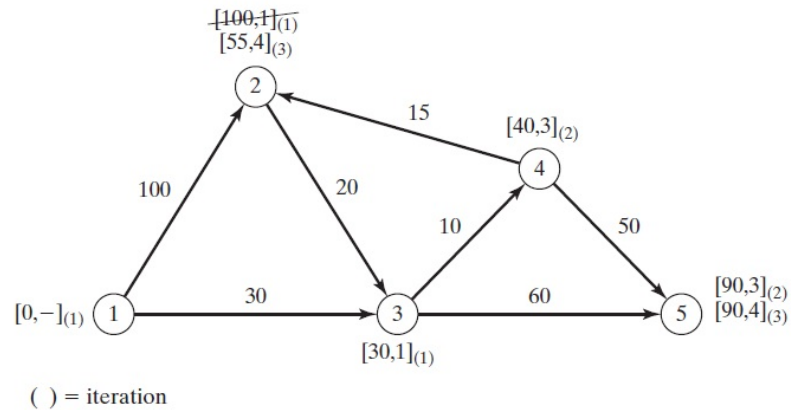
Node permanen yang bisa terhubung dengan node 2 hanyalah node 3. Node 3 tidak bisa dilabel ulang mengingat sudah permanen. Di iterasi ini semua node sudah memiliki label permanen kecuali label 5.



Label 5 ditandai permanen, dan proses iterasi berhenti.

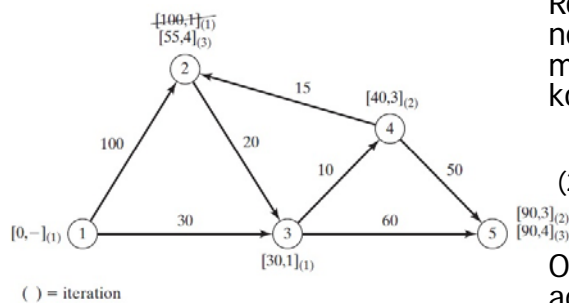
10

Prosedur pelabelan Dijkstra



11

Route terpendek? Back-tracking



Route terpendek (misalnya dari node 1 ke node 2) bisa dilacak secara "back-tracking" mulai dari node 2 dengan menggunakan kombinasi node dan label sebagai berikut:

(2) → [55, 4] → (4) → [40, 3] → (3) → [30, 1] → (1)

Oleh karena itu route yang ditemukan adalah sebagai berikut

(1) → (3) → (4) → (2)

dengan total jarak = 55 km

12

Soal Latihan

13