

Menentukan Pohon Merentang Minimum Dengan Algoritma Sollin

Rizky Delfianto – NIM 13507032

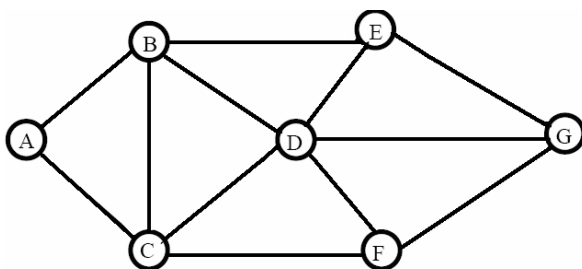
Jurusan Teknik Informatika ITB, Bandung, email: if17032@students.if.itb.ac.id

Abstract – Penentuan pohon merentang minimum dari suatu graf dapat dilakukan dengan beberapa cara. Cara-cara tersebut pada umumnya menggunakan tiga algoritma dasar, yaitu algoritma Borůvka, Prim, dan Kruskal atau menggunakan kombinasi maupun pengembangan dari salah satu atau beberapa algoritma dasar tersebut. Dalam kenyataannya, metode penentuan pohon merentang minimum yang sering dipelajari hanya yang menerapkan algoritma Prim dan Kruskal. Dari algoritma Borůvka dapat diturunkan sebuah algoritma yang dinamai seperti nama penemunya yaitu algoritma Sollin. Dan algoritma Sollin nampaknya juga kurang populer karena masih ada orang-orang yang berkemcimpung pada dunia pemrograman tapi belum mengetahui algoritma Sollin. Oleh karena itu, penulis dalam bahasan ilmiah ini ingin lebih mempopulerkan lagi algoritma Sollin, yang merupakan penurunan dari salah satu dari tiga algoritma dasar, yaitu algoritma Borůvka.

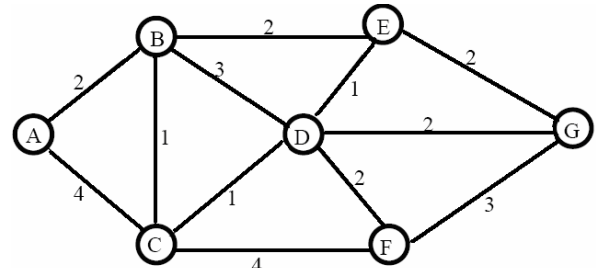
Kata Kunci: pohon, graf, pohon merentang minimum, Sollin, Borůvka, Prim, dan Kruskal.

1. PENDAHULUAN

Graf adalah himpunan atau kumpulan benda-benda yang dinamai verteks (node atau simpul) yang dihubungkan oleh edges (arc atau busur). Graf terdiri atas dua himpunan, himpunan verteks (V) yang merupakan himpunan simpul yang menyusun graf dan tidak kosong dan himpunan edges (E) yang merupakan himpunan busur atau sisi yang menghubungkan simpul-simpul pada graf dan himpunan ini boleh kosong. Dalam graf, terdapat istilah graf berbobot, yaitu graf yang tiap busur atau sisinya memiliki suatu bobot tertentu. Graf berbobot sering digunakan untuk memecahkan permasalahan yang timbul di sekitar kita.

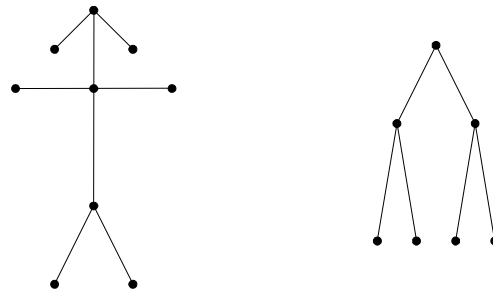


Gambar 1: Contoh Graf Tak-Berarah



Gambar 2: Contoh Graf Tak-Berarah yang Berbobot

Pohon (struktur data) adalah suatu struktur data yang dapat digunakan secara luas yang memiliki “bentuk” menyerupai sebuah pohon yang pada umumnya terdiri atas akar dan daun. Pohon dapat juga diartikan sebagai graf tak-berarah terhubung yang tidak mengandung sirkuit (lintasan yang berawal dan berakhir pada simpul yang sama). Pada pohon struktur data, terdapat beberapa terminology seperti akar, daun, simpul, anak, dan ayah. Akar adalah simpul awal yang terletak paling atas dari sebuah pohon atau dapat dikatakan pula sebagai simpul yang tidak memiliki ayah. Sedangkan simpul adalah alat penyimpan nilai dalam sebuah pohon. Daun adalah simpul yang berada pada tingkat terendah dari sebuah pohon atau dapat dikatakan tidak mempunyai anak.



Gambar 3 : Contoh Pohon

Dalam pokok bahasan pohon, ada yang disebut dengan pohon merentang. Pohon merentang dari sebuah graf adalah upagraf yang memiliki himpunan verteks yang sama dengan graf asalnya dan berupa pohon. Sebuah graf terhubung bisa memiliki lebih dari satu buah pohon merentang minimum. Dan pohon merentang dari sebuah graf berbobot yang memiliki bobot paling kecil dinamakan pohon merentang minimum.

2. ALGORITMA DASAR

Penentuan pohon merentang minimum dari suatu graf dapat dilakukan dengan menggunakan algoritma dasar yang ada. Tapi kebanyakan dari kita hanya mengetahui sebatas algoritma Prim dan algoritma Kruskal. Algoritma Sollin diturunkan dari algoritma Borůvka yang merupakan salah satu dari algoritma dasar dalam penentuan pohon merentang minimum.

2.1. Algoritma Prim

Algoritma Prim merupakan algoritma untuk mencari pohon merentang minimum dari sebuah graf berbobot langkah demi langkah dengan menganbil sisi atau busur dari graf yang mempunyai bobot minimum kemudian mengulang proses yang sama (mengambil sisi yang berbobot minimum) yang tidak membentuk sirkuit.

Secara terurut, algoritma Prim dapat dituliskan sebagai berikut :

1. Ambil sisi pada graf G yang memiliki bobot minimum, masukkan ke dalam T
2. Pilih sisi (u, v) yang memiliki bobot minimum dan bersisian dengan simpul di T , tetapi (u, v) tidak membentuk sirkuit di T . Masukkan (u, v) ke dalam T
3. Ulangi langkah 2 sebanyak $n-2$ kali

Algoritma jelasnya dapat dilihat pada kotak di bawah ini :

```
procedure Prim(input G : graf, output T : pohon)
{ Membentuk pohon merentang minimum T dari graf
terhubung-berbobot G.
Masukan: graf-berbobot terhubung  $G = (V, E)$ ,
dengan  $|V| = n$ .
Keluaran: pohon rentang minimum  $T = (V, E')$  }
Deklarasi
i, p, q, u, v : integer
Algoritma
Cari sisi (p,q) dari E yang berbobot terkecil
 $T \leftarrow \{(p,q)\}$ 
for i←1 to n-2 do
    Pilih sisi (u,v) dari E yang bobotnya terkecil
    namun bersisian dengan simpul di T
     $T \leftarrow T \cup \{(u,v)\}$ 
endfor
```

2.2. Algoritma Kruskal

Algoritma Kruskal adalah salah satu algoritma dasar yang dapat digunakan untuk menentukan pohon

merentang minimum dengan cara menurutkan terlebih dahulu sisi-sisi graf dari kecil ke besar. Kemudian, secara berulang sisi yang memiliki bobot terkecil dan tidak membentuk siklus di T dimasukkan ke himpunan T .

Algoritmanya dapat disusun ke dalam langkah-langkah terurut sebagai berikut :

1. Urutkan sisi-sisi graf dari kecil ke besar. T merupakan himpunan kosong
2. Pilih sisi (u, v) dengan bobot minimum yang tidak membentuk sirkuit di T . Masukkan ke dalam T
3. Ulangi langkah 2 sebanyak $n-1$ kali

Algoritma jelasnya dapat dilihat pada kotak di bawah ini :

```
procedure Kruskal(input G : graf, output T : pohon)
{ Membentuk pohon merentang minimum T dari graf
terhubung-berbobot G.
Masukan: graf-berbobot terhubung  $G = (V, E)$ ,
dengan  $|V| = n$ 
Keluaran: pohon rentang minimum  $T = (V, E')$ 
}
Deklarasi
i, p, q, u, v : integer
Algoritma
(Asumsi: sisi-sisi dari graf sudah diurut menaik
berdasarkan bobotnya – dari bobot kecil ke bobot
besar)
 $T \leftarrow \{\}$ 
while jumlah sisi  $T < n-1$  do
    Pilih sisi (u,v) dari E yang bobotnya terkecil
    if (u,v) tidak membentuk siklus di T then
         $T \leftarrow T \cup \{(u,v)\}$ 
    endif
endfor
```

2.3. Algoritma Sollin

Seperti sudah dikatakan sebelumnya, algoritma Sollin diturunkan dari algoritma Borůvka. Oleh karena itu, dalam pengaplikasiannya, algoritma Sollin berbeda dengan algoritma Borůvka. Algoritma Sollin menentukan pohon merentang minimum dengan cara melakukan penghapusan sisi-sisi yang tidak menyebabkan graf menjadi tidak terhubung atau

membentuk sirkuit. Penghapusan tersebut dimulai dari sisi atau busur yang memiliki bobot terbesar hingga terkecil. Penghapusan dilakukan setelah sebelumnya sisi-sisi pada graf diurutkan berdasarkan bobotnya dari besar ke kecil. Sedangkan algoritma Borůvka menambahkan sisi dengan bobot terkecil ke himpunan T tanpa memperhatikan sisi yang sebelumnya sudah ditambahkan pada himpunan T. Dan kemudian menggabungkan sisi-sisi tersebut hingga terbentuk pohon merentang.

Algoritma Sollin dapat dituliskan secara terurut sebagai berikut :

1. Urutkan sisi-sisi pada graf berdasarkan bobotnya dari besar ke kecil, misalkan jumlah sisiya n
2. Lakukan penghapusan setiap sisi yang tidak menyebabkan graf menjadi tidak terhubung sampai didapat n – 1 sisi yang tersisa

Algoritma Borůvka dapat dituliskan dalam langkah-langkah terurut sebagai berikut :

1. Mulai dengan graf berbobot dan himpunan T yang kosong
2. Sementara simpul graf G yang terhubung oleh T disjoint :
 1. Mulai dengan himpunan sisi E yang kosong
 2. Untuk tiap komponen :
 1. Mulai dengan himpunan sisi S yang kosong
 2. Untuk tiap simpul di G :
 1. Tambahkan sisi dengan bobot terkecil pada suatu simpul ke simpul lain dengan elemen terpisah ke himpunan S
 3. Tambahkan sisi dengan bobot minimum di S ke himpunan T
 3. Tambahkan hasil dari E ke himpunan T
3. T adalah pohon merentang minimum dari graf G

Berikut adalah algoritma jelasnya dari algoritma Sollin:

```

procedure Sollin(input G : graf, output T : pohon)
{ Membentuk pohon merentang minimum T dari graf terhubung –berbobot G.
Masukan: graf-berbobot terhubung G = (V, E), dengan |V|= n
Keluaran: pohon rentang minimum T = (V, E')
}

```

Deklarasi

Algoritma

(Asumsi: sisi-sisi dari graf sudah diurut menurun berdasarkan bobotnya – dari bobot besar ke bobot kecil)

T ← {E}

while masih ada sisi yang belum diperiksa do

Pilih sisi (u,v) dari E yang bobotnya terbesar

if (u,v) tidak menyebabkan G tidak terhubung bila dihapus then

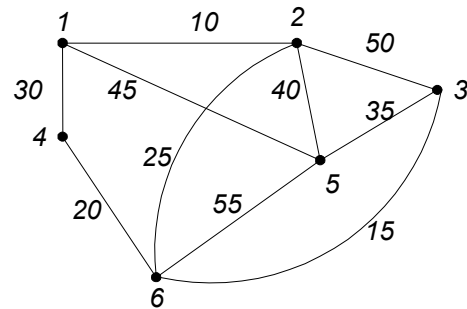
T ← T - {(u,v)}

endif

endfor

Berikut adalah contoh penggunaan algoritma Sollin dalam menentukan pohon merentang minimum dari sebuah graf berbobot.

Contoh:



Gambar 4: Graf G

Akan kita selesaikan dengan metode Sollin.

Mula-mula kita buat dulu tabel yang berisikan sisi-sisi yang terurut dari besar ke kecil.

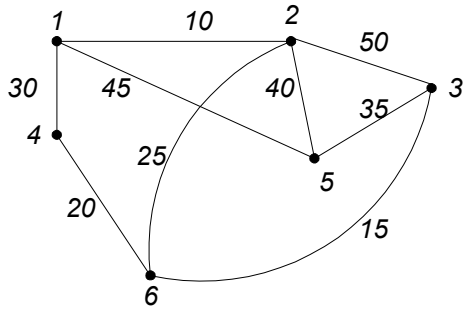
Tabel 1. Sisi-Sisi Graf

Bobot	Sisi
55	(5,6)
50	(2,3)
45	(1,5)
40	(2,5)
35	(3,5)
30	(1,4)
25	(2,6)
20	(4,6)
15	(3,6)
10	(1,2)

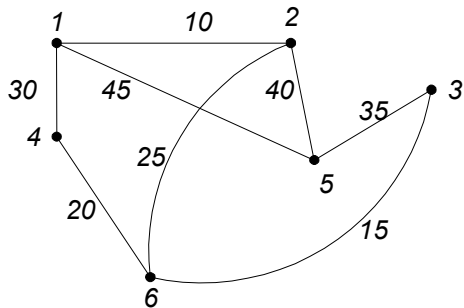
Kemudian kita lakukan penghapusan sisi dimulai dari yang memiliki bobot terbesar dan tidak membuat graf

menjadi tidak terhubung.

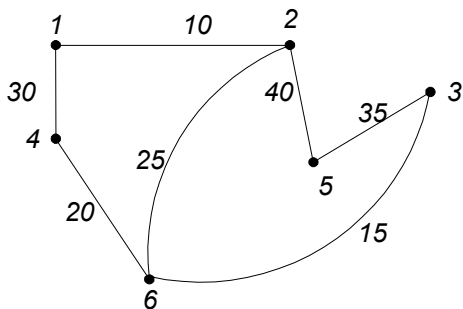
Sisi (5,6) dapat dihapus karena tidak menyebabkan graf menjadi tidak terhubung.



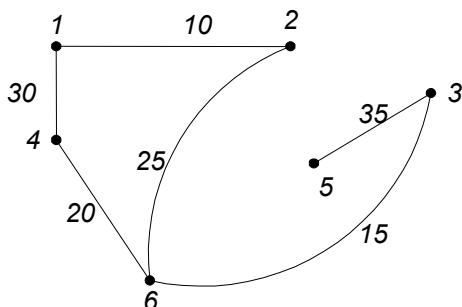
Sisi (2,3) dapat dihapus karena tidak menyebabkan graf menjadi tidak terhubung.



Sisi (1,5) dapat dihapus karena tidak menyebabkan graf menjadi tidak terhubung.

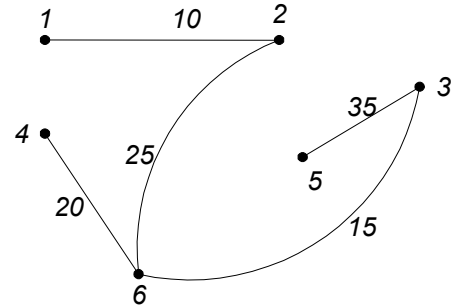


Sisi (2,5) dapat dihapus karena tidak menyebabkan graf menjadi tidak terhubung.



Sisi (3,5) tidak dapat dihapus karena akan menyebabkan graf menjadi tidak terhubung (simpul nomor 5 akan menjadi simpul terencil).

Sisi (1,4) dapat dihapus karena tidak menyebabkan graf menjadi tidak terhubung.



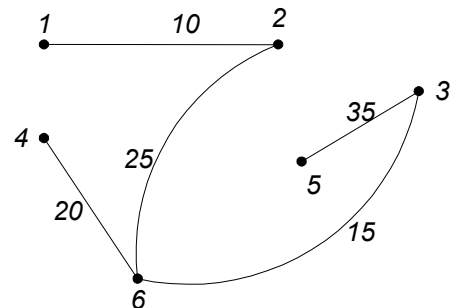
Sisi (2,6) tidak dapat dihapus karena akan menyebabkan graf menjadi tidak terhubung (simpul 1 dan 2 akan terpisah dengan simpul yang lain).

Sisi (4,6) tidak dapat dihapus karena akan menyebabkan graf menjadi tidak terhubung (simpul nomor 4 akan menjadi simpul terencil).

Sisi (3,6) tidak dapat dihapus karena akan menyebabkan graf menjadi tidak terhubung (simpul 3 dan 5 akan terpisah dengan simpul yang lain).

Sisi (1,2) tidak dapat dihapus karena akan menyebabkan graf menjadi tidak terhubung (simpul nomor 1 akan menjadi simpul terencil).

Akhirnya diperoleh hasil pohon merentang minimum sebagai berikut :



Gambar 5 : Pohon Merentang Minimum dari Graf G

3. HASIL DAN PEMBAHASAN

Dari pseudo-code algoritma dasar dan algoritma Sollin, dapat dilihat algoritma Sollin dapat dikatakan mendekati algoritma dari Kruskal. Hanya perbedaannya, pada Sollin, sisi diurutkan mengecil dan yang dilakukan bukan menambahkan tapi menghapus sisi dari himpunan T yang tidak menyebabkan graf menjadi tidak terhubung. Dapat dilihat pula bahwa algoritma Sollin adalah algoritma yang sederhana dan mudah dimengerti. Hal ini membuat penulis semakin penasaran kenapa algoritma Sollin ini kurang populer dibandingkan algoritma Kruskal.

4. KESIMPULAN

Kesimpulan yang dapat diperoleh antara lain adalah :

1. Algoritma Sollin masih kalah populer dibandingkan algoritma dasar seperti Prim dan Kruskal
2. Algoritma Sollin merupakan turunan dari algoritma dasar Borůvka dan mendekati algoritma Kruskal
3. Perbedaan antara Sollin dan Kruskal terletak pada pengurutan sisi yang mengecil dan proses penghapusan sisi dari himpunan. Berbeda dengan Kruskal yang mengurutkan sisi membesar dan proses yang dilakukan adalah penambahan sisi ke himpunan

DAFTAR REFERENSI

- [1] Munir, Rinaldi. (2006). Bahan Kuliah IF2153 Matematika Diskrit. Departemen Teknik Informatika, Institut Teknologi Bandung.
- [2] Prim's algorithm - Wikipedia, the free encyclopedia.

http://en.wikipedia.org/prim's_algorithm.htm . Tanggal akses : 19 Desember 2008, pukul 14.47

[3] Kruskal's algorithm - Wikipedia, the free encyclopedia.

http://en.wikipedia.org/kruskal's_algorithm.htm . Tanggal akses: 19 Desember 2008, pukul 14.48

[4] Boruvka's algorithm - Wikipedia, the free encyclopedia.

http://en.wikipedia.org/boruvka's_algorithm.htm . Tanggal akses : 19 Desember 2008, pukul 14.47

[5]

<http://mufidnilmada.staff.gunadarma.ac.id/Downloads/files/1866/span+tree.doc>. Tanggal akses : 12 Desember 2008, pukul 9.13

[6] <http://www.ics.uci.edu/~eppstein/161/960206.html> Tanggal akses : 20 Desember 2008, pukul 16.21

[7] <http://one.indoskripsi.com/judul-skripsi-tugas.../algoritma/buatlah-minimum-spanning-tree-dengan-metode-solin-kruskal>. Tanggal akses : 12 Desember 2008, pukul 9.01

[8]

<http://syakur.staff.gunadarma.ac.id/Downloads/files/5850/Minimum+Spanning+Tree.pdf>. Tanggal akses : 12 Desember 2008, pukul 9.26

[9] Pohon (struktur data) - Wikipedia, the free encyclopedia.

[http://id.wikipedia.org/wiki/Pohon_\(struktur_data\)](http://id.wikipedia.org/wiki/Pohon_(struktur_data)). Tanggal akses : 19 Desember 2008, pukul 14.46

[10] Teori Graf - Wikipedia, the free encyclopedia.

http://id.wikipedia.org/wiki/Teori_graf. Tanggal akses : 12 Desember 2008, pukul 8.56