

Penggunaan Pohon Merentang Minimum dalam Jaringan Komputer

Samuel 13518041¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

¹13518041@std.stei.itb.ac.id

Abstract—Teori mengenai pohon dan graf memiliki banyak manfaat dalam kehidupan manusia. Pohon dan graf dapat dijadikan pemodelan untuk masalah-masalah yang ditemui dalam kehidupan sehari-hari seorang manusia. Namun, penggunaan pohon dan graf dalam masalah-masalah tersebut mungkin tidak terlalu memberi manfaat yang besar. Namun, di bidang teknologi, terutama teknologi informasi, penggunaan pohon dan graf memberi peran yang sangat besar dalam pemecahan masalah-masalah di bidang tersebut. Salah satunya adalah di bidang jaringan komputer. Optimisasi jaringan komputer dapat dilakukan dengan bantuan pohon merentang minimum.

Keywords—algoritme Kruskal, pohon merentang minimum, simpul, dan sisi.

I. PENDAHULUAN

Di dalam bidang ilmu matematika, ada banyak sekali jenis pemodelan. Pemodelan menjadi sebuah langkah awal dalam pemecahan sebuah masalah di bidang ilmu matematika. Hal ini dilakukan dengan tujuan untuk mempermudah komputasi. Dengan pemodelan, kita bisa menemukan satu atau beberapa pola yang terdapat dalam masalah tersebut. Dengan begitu, kita bisa menemukan solusi yang tepat terhadap masalah tersebut.

Di zaman modern ini—yang sering disebut orang-orang sebagai era Revolusi Industri 4.0, salah satu teknologi yang mengalami perkembangan sangat pesat adalah teknologi di bidang informasi dan komunikasi. Dalam kurun waktu beberapa dekade saja, teknologi ini bisa berkembang untuk menghubungkan orang-orang yang ada di berbagai belahan dunia. Orang-orang yang saling terkoneksi ini membentuk sebuah jaringan informasi yang sangat luas. Dengan begitu, pertukaran informasi di antara orang-orang tersebut bukanlah menjadi sebuah hal yang sulit lagi.

Sebuah lembaga, instansi, atau perusahaan memiliki banyak sekali sumber daya di dalamnya. Perusahaan tersebut membuhkan kinerja yang maksimal dari setiap sumber daya yang dimiliki agar perusahaan tersebut dapat berjalan dengan maksimal dan memenuhi tujuan atau capaian yang sudah ditentukan sebelumnya. Oleh karena itu, aliran informasi dan komunikasi antara sumber daya menjadi sebuah hal yang sangat penting di dalam sebuah perusahaan. Dengan begitu, sebuah perusahaan membutuhkan koneksi yang baik antara sumber daya yang ada. Koneksi ini pada akhirnya akan membentuk sebuah jaringan. Salah satu jaringan yang terdapat dalam sebuah

perusahaan adalah jaringan komputer yang merupakan kumpulan koneksi dari komputer-komputer yang ada di perusahaan tersebut.

Pembuatan sebuah jaringan komputer tentunya memiliki biaya, sama seperti hal lainnya yang ada di Bumi ini. Oleh karena itu, jaringan ini harus dioptimisasi sedemikian rupa untuk menekan biaya tersebut. Meskipun begitu, kinerja jaringan harus tetap dibuat seoptimal mungkin. Di sinilah pohon merentang minimum (*minimum spanning tree*) berperan dalam pengoptimisasian tersebut.

II. LANDASAN TEORI

A. Graf (Graph)

Istilah graf pertama kali ditemukan pada jurnal seorang matematikawan bernama J.J. Sylvester pada tahun 1878. Judul dari jurnal tersebut adalah "On an application of the new atomic theory to the graphical representation of the invariants and covariants of binary quantics, – with three appendices".^[1]

Graf merupakan sebuah struktur diskrit yang terdiri atas simpul (*vertex*) dan sisi (*edge*).^[2] Karena graf merupakan sebuah struktur diskrit, anggotanya, yaitu simpul dan sisi, bukan merupakan benda-benda kontinu. Oleh karena itu, anggota dari sebuah graf direpresentasikan sebagai himpunan.

Notasi dari graf adalah

$$G = (V, E)$$

di mana G merupakan sebuah graf, V merupakan himpunan dari simpul, dan E merupakan himpunan dari sisi. Setiap sisi terasosiasi dengan satu atau dua buah simpul yang disebut sebagai titik akhir (*endpoint*). Sisi ini menghubungkan titik akhir tersebut.

Untuk memudahkan pemahaman, perhatikan ilustrasi berikut.

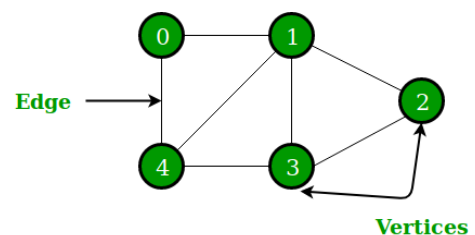


Figure 1. Contoh graf

(Sumber: <https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/>)

Misalkan graf di atas kita sebut dengan graf G , maka graf tersebut dapat disajikan sesuai dengan notasi sebelumnya.

$$V = \{0,1,2,3,4\}$$

$$E = \{(0,1), (0,4), (1,2), (1,3), (1,4), (2,3)\}$$

Graf dapat dibagi menjadi beberapa jenis berdasarkan sisinya.

1. Graf Sederhana (Simple Graph)

Graf ini dinamakan sederhana karena graf ini merupakan bentuk dasar graf. Sebuah graf sederhana hanya memiliki maksimal satu sisi untuk setiap sembarang dua simpul. Selain itu, pada graf sederhana, tidak ada sisi antara dua simpul yang sama, atau yang biasa disebut kalang (*loop*). Dengan begitu, setiap sisi pasti terhubung dengan dua simpul yang berbeda dan tidak ada sisi lain yang sama dengan simpul tersebut. Jika ada sisi yang terasosiasi dengan $\{u,v\}$, kita bisa juga mengatakan bahwa ada sisi $\{u,v\}$.

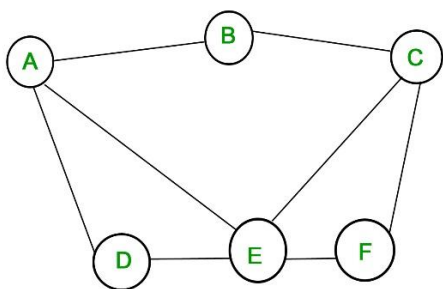


Figure 2. Contoh graf sederhana

(Sumber: <https://www.geeksforgeeks.org/mathematics-graph-theory-basics/>)

2. Graf Jamak (Multigraph)

Sebuah pemodelan bisa saja membutuhkan dua jalur yang berbeda antara dua buah benda. Oleh karena itu, di graf jamak, sebuah graf bisa memiliki beberapa sisi yang menghubungkan dua buah simpul yang sama. Jika ada m buah sisi yang menghubungkan sepasang simpul yang sama $\{u,v\}$, kita bisa juga mengatakan bahwa sisi $\{u,v\}$ memiliki kelipatan m .

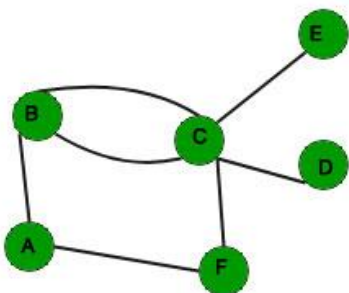
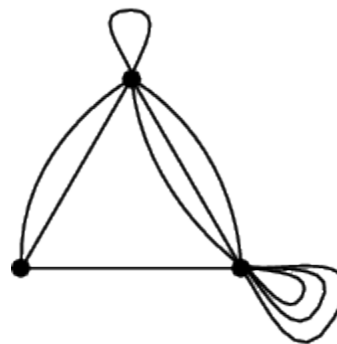


Figure 3. Contoh graf jamak

(Sumber: <https://www.geeksforgeeks.org/mathematics-graph-theory-basics/>)

Selain itu, sebuah graf jamak juga bisa memiliki sisi yang menghubungkan sebuah simpul dengan dirinya sendiri. Sisi seperti ini disebut kalang (*loop*). Terkadang, sebuah graf juga

bisa memiliki beberapa kalang yang sama. Graf seperti ini memiliki sebutan graf semu (*pseudograph*).



pseudograph

Figure 4. Contoh graf semu

(Sumber: <http://140.177.205.23/Pseudograph.html>)

Dari beberapa jenis graf yang sudah dijelaskan sebelumnya, ada sebuah kesamaan di antara mereka, yaitu graf-graf tersebut tidak memiliki arah. Oleh karena itu, di graf sederhana, jika terdapat sisi $\{u,v\}$, tidak ada sisi $\{v,u\}$ karena dua sisi ini merupakan sisi yang sama. Sedangkan di graf jamak, sisi $\{u,v\}$ dan sisi $\{v,u\}$ hanya akan menambah kelipatan sisi $\{u,v\}$ karena kedua sisi tersebut juga sama.

Terkadang, sebuah pemodelan koneksi juga membutuhkan arah jalur. Oleh karena itu, ada graf yang memiliki arah pada sisinya. Arah ini biasanya digambarkan dengan kepala panah pada ujung sisi tersebut. Graf ini disebut dengan graf berarah.

3. Graf Berarah Sederhana (Simple Directed Graph)

Graf ini mirip dengan graf sederhana. Di dalam graf ini, setiap sisi menghubungkan dua simpul yang berbeda dan tidak ada sisi lain lagi yang menghubungkan sepasang simpul tersebut dengan arah yang sama. Perbedaan graf ini dengan graf sederhana hanyalah setiap sisi di graf berarah sederhana memiliki arah. Sebuah sisi $\{u,v\}$ berbeda dengan sisi $\{v,u\}$. Oleh karena itu, Walaupun pada sepasang simpul yang berbeda terdapat dua sisi tetapi dengan arah yang berbeda, graf tersebut masih termasuk graf berarah sederhana.

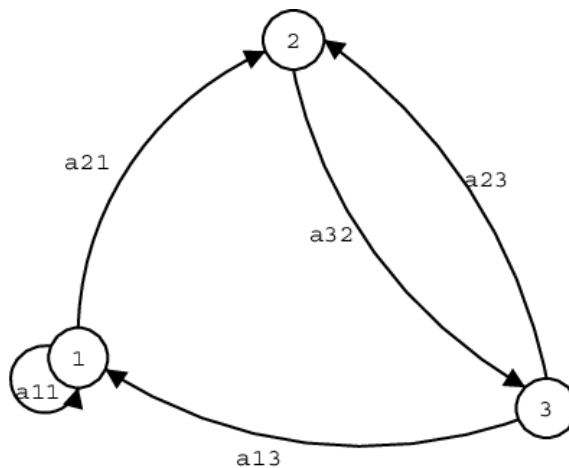


Figure 5. Contoh graf berarah sederhana

(Sumber: https://www.researchgate.net/figure/A-simple-directed-graph-digraph_fig3_268075859)

4. Graf Berarah Jamak (Directed multigraph)

Graf berarah jamak mirip dengan graf berarah sederhana. Sebuah graf dikatakan graf berarah jamak jika terdapat beberapa sisi berarah yang berasal dari simpul yang sama dan berakhir pada simpul yang sama—kedua simpul tersebut mungkin saja merupakan simpul yang sama sehingga membentuk kalang. Saat ada m buah sisi berarah yang terasosiasi dengan simpul $\{u,v\}$, maka sisi $\{u,v\}$ memiliki kelipatan sebanyak m .

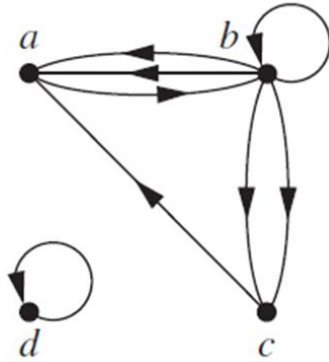


Figure 6. Contoh graf berarah jamak
(Sumber: <http://mymathangels.com/problem-84/>)

B. Pohon (Tree)

Sebuah graf sederhana yang tidak memiliki sirkuit disebut sebagai pohon.^[2] Selain itu, sebuah graf tidak berarah adalah pohon jika dan hanya jika ada satu sisi unik yang menghubungkan dua simpul.

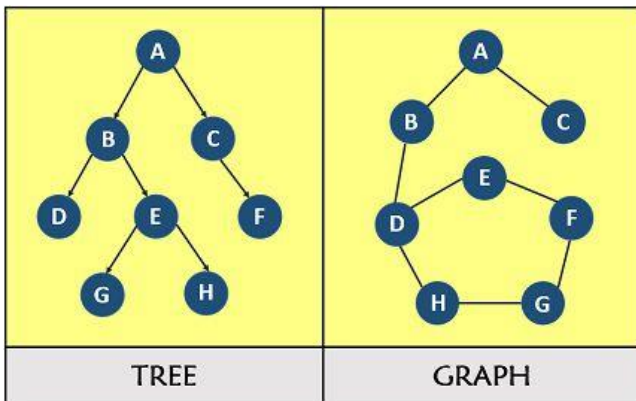


Figure 7. Perbedaan pohon (kiri) dan graf (kanan)
(Sumber: <https://techdifferences.com/difference-between-tree-and-graph.html>)

Berdasarkan gambar di atas, kita dapat melihat bahwa gambar yang di kanan merupakan graf yang bukan pohon karena adanya sirkuit $D-E-F-G-H-D$. Sedangkan, di gambar kiri, tidak ada sirkuit di mana pun. Oleh karena itu, gambar sebelah kiri merupakan graf sekaligus pohon.

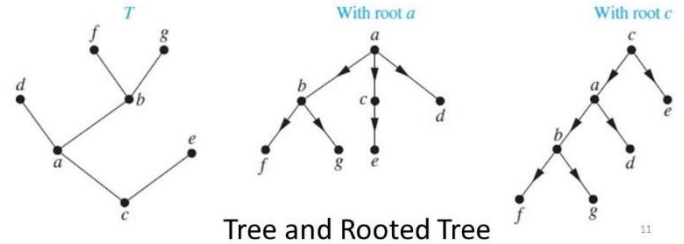
Pohon sudah mulai digunakan sejak tahun 1857 oleh seorang kimiawan Inggris bernama Arthur Cayley untuk menghitung jenis-jenis senyawa kimia tertentu. Selain itu, ternyata, pohon sangat berperan penting dalam bidang ilmu komputer. Sebuah pohon biasanya dimanfaatkan untuk pembuatan sebuah algoritme. Beberapa contoh algoritme terkenal yang memanfaatkan pohon dalam pembuatannya adalah pohon

Merkle dalam pembuatan fungsi hash dan pohon biner dalam pembuatan kode Huffman.

Di bawah ini, akan dibahas beberapa jenis pohon.

1. Pohon Berakar

Pohon ini memiliki karakteristik bahwa ada satu simpul yang dipilih sebagai akar (*root*). Kemudian, semua simpul lainnya dibuat menjalar dari akar tersebut.



Tree and Rooted Tree

Figure 8. Contoh pohon berakar
(Sumber: <https://slideplayer.com/slide/8471551/>)

Misalkan terdapat sebuah simpul v yang bukan akar dari sebuah pohon T . Maka, simpul u dikatakan *parent* (orangtua) dari v jika dan hanya jika ada sisi $\{u,v\}$ di mana u lebih dekat ke akar daripada v . Simpul v juga bisa disebut sebagai *child* (anak) dari simpul u . Beberapa simpul yang memiliki *parent* yang sama disebut sebagai *siblings* (saudara). Selain itu, simpul yang tidak memiliki *children* disebut sebagai *leaf* (daun). Sedangkan, simpul yang memiliki setidaknya satu *child* (bisa lebih) disebut sebagai *internal vertices* (simpul internal).

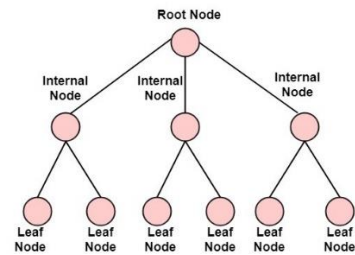


Figure 9. Hubungan antar simpul dalam pohon
(Sumber: <https://www.javatpoint.com/discrete-mathematics-introduction-of-trees>)

Sebuah pohon berakar bisa saja memiliki simpul internal yang anaknya memiliki urutan dari kiri ke kanan. Salah satunya adalah yang terdapat di pohon biner (*binary tree*). Pada pohon biner, anak pertama adalah anak kiri dan anak kedua adalah anak kanan. Dari setiap anak ini, bisa dibentuk subpohon lagi.

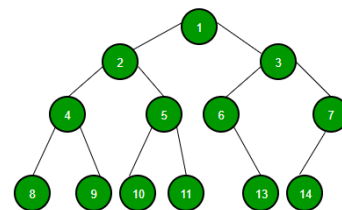


Figure 10. Contoh pohon biner
(Sumber: <https://www.geeksforgeeks.org/difference-between-binary-tree-and-binary-search-tree/>)

2. Pohon Merentang Minimum (Minimum Spanning Tree)

Sebuah graf sederhana G bisa memiliki sebuah subgraf berupa pohon merentang (*spanning tree*) T jika T mengandung semua simpul yang ada di graf G . Agar sebuah graf bisa memiliki pohon merentang, graf tersebut harus merupakan graf terhubung (*connected graph*). Graf terhubung merupakan graf yang setiap simpul memiliki lintasan ke semua simpul lainnya.

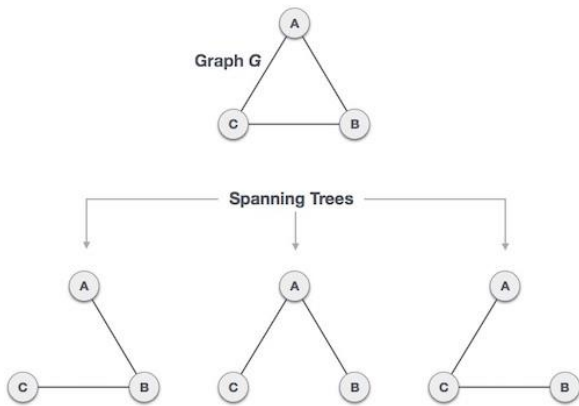


Figure 11. Contoh graf terhubung dan pohon merentangnya

(Sumber:

https://www.tutorialspoint.com/data_structures_algorithms/spanning_tree.htm)

Namun, graf terhubung tersebut bisa saja memiliki berat atau nilai di setiap sisinya. Graf seperti itu disebut sebagai *connected weighted graph*. Dengan begitu, berbagai macam *spanning tree* mampu menghasilkan berbagai macam total nilai sisinya. Dari hal inilah pengembangan pohon merentang minimum muncul. Pohon merentang minimum adalah sebuah pohon merentang dari sebuah *connected weighted graph* dengan total nilai sisinya merupakan nilai minimum yang mungkin.

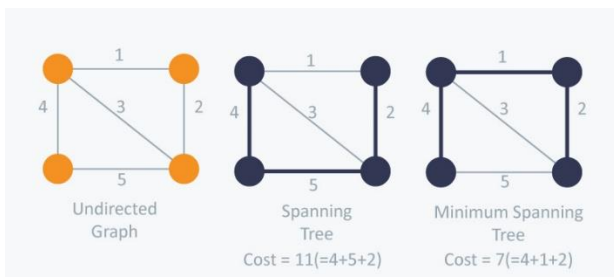


Figure 12. Contoh *spanning tree* dan *minimum spanning tree*

(Sumber:

<https://www.hackerearth.com/practice/algorithms/graphs/minimum-spanning-tree/tutorial/>)

Untuk memperoleh pohon merentang minimum ada beberapa algoritme yang bisa digunakan. Salah satu algoritme yang cukup terkenal adalah algoritme Kruskal.

Sesuai namanya, algoritme ini ditemukan oleh seorang matematikawan dari Amerika Serikat bernama Joseph Bernard Kruskal pada tahun 1956.^[3] Walaupun begitu, dia bukan yang pertama kali menemukan teknik ini. Dia mendapat referensi dari orang yang telah mendahuluinya, yaitu Otakar Borůvka.



Figure 13. Potret Joseph Bernard Kruskal, Jr. (Sumber: <https://alchetron.com/Joseph-Kruskal>)

Berdasarkan jurnal yang beliau buat^[3], beliau mengatakan untuk membangun sebuah pohon merentang minimum dari sebuah graf terhubung membutuhkan $n-1$ cara dengan n merupakan banyak simpul dari graf tersebut.

Langkah pertama dari algoritme Kruskal adalah untuk mengurutkan semua sisi graf secara tidak menurun. Kemudian, sisi-sisi tersebut diambil satu per satu untuk membangun sebuah pohon merentang minimum. Jika sisi yang diambil akan membentuk sirkuit, sisi tersebut diabaikan dan langsung berlanjut ke sisi selanjutnya. Untuk mempermudah pemahaman, perhatikan ilustrasi berikut.

Kruskal's Algorithm

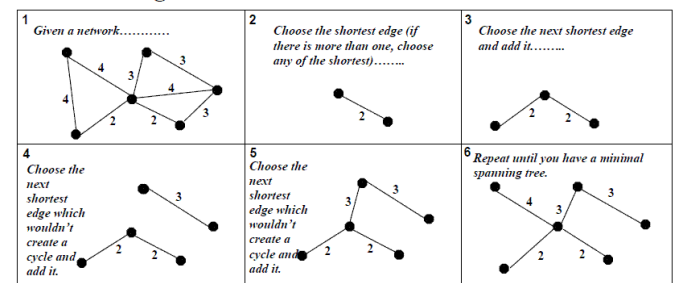


Figure 14. Langkah-langkah algoritme Kruskal

(Sumber: <https://stackoverflow.com/questions/1195872/when-should-i-use-kruskal-as-opposed-to-prim-and-vice-versa/18680823#18680823>)

C. Jaringan Komputer (Computer Network)

Beberapa komputer yang terkoneksi satu sama lain akan membentuk sebuah jaringan komputer. Di dalam jaringan komputer ini, setiap komputer bisa berkomunikasi dan bertukar informasi satu sama lain. Dengan begitu, sebuah jaringan komputer bisa meningkatkan efisiensi dalam penyebaran data.

Oleh karena itu, konektivitas menjadi salah satu hal yang penting untuk dibahas di dalam sebuah jaringan komputer. Konektivitas terjadi dalam berbagai tingkat.^[4] Di tingkat paling bawah, dua buah komputer atau lebih dapat terhubung langsung dengan sebuah medium fisik, seperti kabel. Kita menyebut medium fisik tersebut sebagai hubungan (*link*) dan komputer-komputer yang terhubung sebagai simpul (*nodes*).

Ada beberapa jenis kabel yang cukup umum digunakan sebagai medium fisik. Beberapa jenis kabel tersebut akan dijelaskan di bawah ini.

1. Coaxial Cable

Kabel jenis ini merupakan jenis kabel yang paling umum ada di masyarakat karena kabel ini yang lebih dulu ditemukan, yaitu pada tahun 1880 oleh seorang fisikawan, teknisi, dan matematikawan Inggris bernama Olivier Heaviside.^[5]

Kabel ini menyalurkan informasi melalui sinyal dengan frekuensi radio. Oleh karena itu, kabel ini terbuat dari logam agar dapat menyalurkan listrik dan data tersebut dalam gelombang elektromagnetik dengan cukup mudah, tepatnya kabel ini terbuat dari tembaga (*copper*).

Keuntungan dari kabel ini adalah kabel ini memiliki harga per jarak yang tidak begitu mahal. Selain itu, kabel ini sudah terdapat di mana-mana sehingga tidak perlu membangun infrastruktur baru jika ingin menggunakan kabel ini. Sayangnya, kabel ini juga memiliki beberapa kekurangan, yaitu penurunan kualitas dari informasi yang dibawa kabel cukup besar sehingga kabel ini kurang *reliable* untuk komunikasi jarak jauh. Selain itu, karena kabel ini mengantar informasi dalam bentuk gelombang elektromagnetik, informasi-informasi tersebut menjadi rentan juga terhadap interferensi dari gelombang-gelombang elektromagnetik.

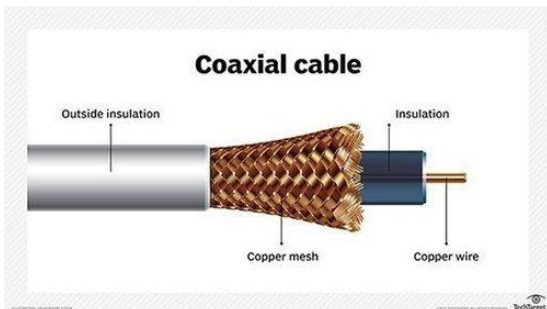


Figure 15. Contoh coaxial cable
(Sumber: <https://www.indiamart.com/proddetail/coaxial-cable-19889516473.html>)

2. Optical Fiber

Kabel ini merupakan jenis terobosan yang merupakan *cutting-edge technology* yang masih terus berkembang hingga sekarang. *Optical fiber* mulai dieksperimentasikan sekitar pertengahan abad 20, yaitu sekitar tahun 1940. Para ilmuwan saat itu mulai bereksperimen dengan transmisi citra melalui tabung sangat tipis. Beberapa tahun kemudian, beberapa ilmuwan berhasil meneruskan upaya tersebut dan menemukan citra juga bisa ditransmisikan melalui kumpulan *optical fiber*.

Inilah konsep dari *optical fiber* itu. Data-data atau informasi ditransmisikan melalui *optical fiber* dengan cara mengubah informasi tersebut ke dalam bentuk cahaya yang kemudian baru akan ditransmisikan melalui *optical fiber* tersebut. Karena informasi berada dalam bentuk cahaya sepanjang transmisi, kualitas dari informasi tersebut sangat terjaga karena tidak mudah untuk gelombang elektromagnetik yang berfrekuensi jauh lebih rendah untuk memberikan interferensi yang cukup signifikan terhadap cahaya tersebut. Selain itu, kecepatan transmisi juga meningkat jauh.

Sayangnya, kekurangan dari kabel ini adalah harga per jarak yang mahal karena bahan kabel ini terbuat dari silika

(bahan dasar kaca) sehingga biaya produksinya cukup mahal. Selain itu, infrastrukturnya juga belum tersebar merata sehingga jika ingin menggunakan medium fisik jenis ini, infrastrukturnya harus dibangun terlebih dahulu.

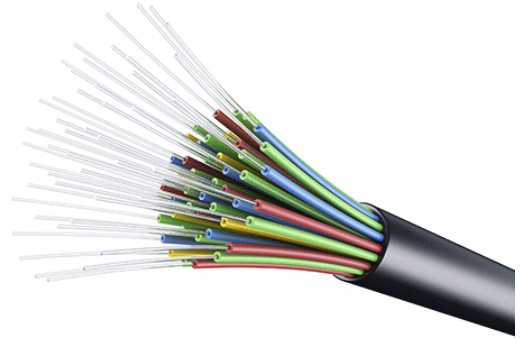


Figure 16. Contoh kabel optic
(Sumber: <https://www.indiamart.com/proddetail/4-core-steel-optical-fiber-cable-19629177148.html>)

Setelah kita melihat bagaimana jenis medium fisik yang digunakan sebagai konektivitas dalam sebuah jaringan komputer sangat mempengaruhi transmisi dari data atau informasi, maka kita sangat membutuhkan transmisi yang optimal dengan biaya yang optimal juga. Oleh karena itu, pohon merentang minimum memiliki peran yang amat besar dalam hal ini.

III. STUDI KASUS

Diberikan sebuah graf terhubung dengan berat yang merepresentasikan sebuah jaringan komputer di suatu tempat. Berat dari setiap sisinya menggambarkan jarak yang terdapat di antara kedua simpul. Oleh karena itu, kita harus mencari pohon merentang minimum (*minimum spanning tree*) dari graf tersebut untuk mengoptimalkan *link* yang terdapat di dalam jaringan komputer tersebut agar biaya yang dikeluarkan juga optimal. Graf tersebut berbentuk berikut ini.

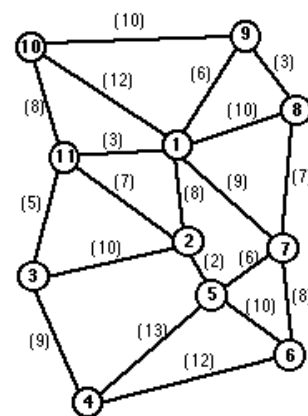


Figure 17. Kasus 1
(Sumber: https://www.me.utexas.edu/~jensen/exercises/mst_spt/mst_spt.html)

Lalu, dengan program untuk mencari MST yang dibuat dalam bahasa pemrograman C++ dengan algoritma Kruskal, data-data yang ada di graf tersebut dimasukkan ke dalam program tersebut. Kemudian, program melakukan komputasi untuk

menghasilkan MST. Setelah itu, keluaran dari program itu adalah seperti di bawah ini.

Node 1	Node 2	Weight
2	5	2
1	11	3
8	9	3
3	11	5
1	9	6
5	7	6
2	11	7
6	7	8
10	11	8
3	4	9

Total cost: 57

Figure 18. Output program untuk kasus 1

Program tersebut berhasil mengeluarkan data-data berupa sisi-sisi yang dipertahankan dan total berat dari sisi-sisi tersebut, yaitu 57. Berikut gambar dari pohon merentang minimum (MST) berdasarkan keluaran program tersebut.

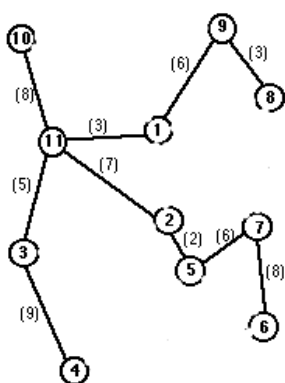


Figure 17. MST dari kasus 1

Dengan begitu, pohon merentang minimum di atas merupakan model dari koneksi optimal untuk jaringan komputer di kasus pertama karena jarak dan jumlah medium fisik dipotong sedemikian rupa untuk menghemat biaya tanpa menghilangkan koneksi dari sebuah komputer dari jaringan tersebut. Dengan begitu, total biaya yang dibutuhkan sekarang hanya 57.

Agar semakin jelas, diberikan kembali kasus kedua mengenai jaringan komputer juga dengan bentuk graf yang berbeda. Graf dari jaringan komputer tersebut adalah seperti berikut.

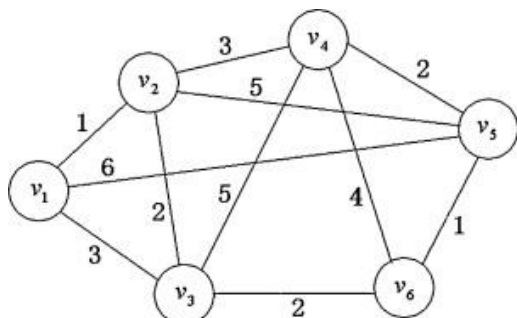


Figure 19. Kasus 2

(Sumber:

<https://www.sciencedirect.com/science/article/pii/S0303264713001676>)

Kemudian, dengan langkah yang sama seperti pada kasus sebelumnya, yaitu dengan memasukkan data-data dari graf tersebut ke dalam program, program berhasil menampilkan keluaran seperti di bawah ini.

Node 1	Node 2	Weight
1	2	1
5	6	1
2	3	2
3	6	2
4	5	2

Total cost: 8

Figure 20. Output program untuk kasus 2

Jika kita merekonstruksi kembali graf tersebut dan menghilangkan semua sisi-sisi pada graf tersebut kecuali yang tertulis di program, kita akan mendapat sebuah pohon merentang minimum seperti di bawah ini.

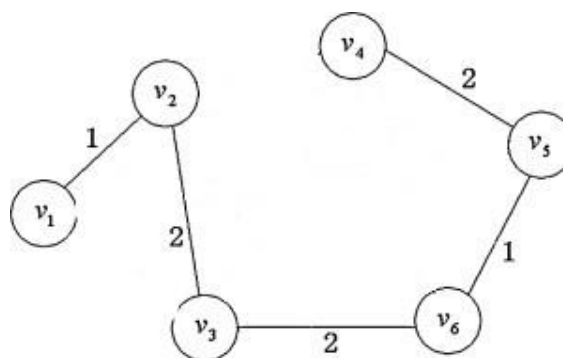


Figure 21. MST dari kasus 2

Dengan begitu, pohon merentang minimum dari graf untuk kasus kedua dapat diperoleh berdasarkan data-data dari keluaran program. Pohon merentang minimum di atas hanya memiliki total biaya sebesar 8 yang tentunya merupakan biaya paling rendah yang mungkin dari graf tersebut tanpa menghilangkan koneksi satu komputer pun dari jaringan tersebut.

IV. KESIMPULAN

Pemodelan menggunakan graf dan pohon memiliki peran yang sangat besar dalam pemecahan berbagai persoalan. Apalagi jika persoalan tersebut berhubungan dengan bidang ilmu matematika atau informatika. Di kesempatan kali ini, penulis berhasil menggunakan konsep dan teori seputar graf dan pohon, terutama pohon merentang minimum (*minimum spanning tree*) untuk memecahkan persoalan terkait optimisasi sebuah jaringan komputer agar memiliki biaya yang serendah mungkin tanpa menghilangkan satu buah komputer pun dari jaringan tersebut. Algoritma Kruskal yang dipakai untuk mencari pohon merentang minimum terbukti mampu menjadi bahan utama dalam solusi dari masalah-masalah tersebut.

V. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Tuhan Yang Maha Esa karena telah senantiasa memberkati dan menyertai penulis dalam masa penulisan makalah dengan judul "Penggunaan Pohon Merentang Minimum dalam Jaringan Komputer". Dengan begitu, makalah ini dapat diselesaikan dengan baik dan

dapat berguna bagi orang-orang yang akan membaca makalah ini di masa depan. Penulis juga tidak lupa untuk berterima kasih kepada para dosen pengampu mata kuliah IF2120 Matematika Diskrit pada semester I tahun ajaran 2019/2020, yaitu Fariska Zakhralativa, Dr. Rinaldi Munir, dan Harlili, M.Sc., atas berbagai ilmu yang telah diberikan mereka terkait bidang graf, pohon, dan ilmu-ilmu yang berhubungan dengan Matematika Diskrit lainnya yang membantu dalam penulisan makalah ini. Terakhir, penulis juga ingin berterima kasih kepada rekan-rekan satu angkatan jurusan Teknik Informatika yang memberi ide mengenai makalah ini dan yang kerap memberi semangat semasa penulisan makalah ini berlangsung.

REFERENCES

- [1] J.J. Sylvester, "On an application of the new atomic theory to the graphical representation of the invariants and covariants of binary quatics, – with three appendices", in *American Journal of Mathematics*, vol. 1, no. 1, Baltimore: The John Hopkins University Press, 1878, pp. 64–104.
- [2] K. H. Rossen, *Discrete Mathematics and Its Applications*, 7th ed., New York: McGraw-Hill, 2012.
- [3] J.B. Kruskal, Jr., "On the shortest spanning subtree of a graph and the traveling salesman problem", in *Proceedings of the American Mathematical Society*, vol. 7, no. 1, Providence: American Mathematical Society, 1956, pp. 48–50.
- [4] L.L. Peterson and B.S. Davie, *Computer Networks: A Systems Approach*, 4th ed., San Francisco: Elsevier, 2007.
- [5] P.J. Nahin, *Oliver Heaviside: The Life, Work, and Times of an Electrical Genius of the Victorian Age*, Baltimore: The John Hopkins University Press, 2002.

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, 5 Desember 2019



Samuel
13518041