

BANYAK SIRKUIT EULER PADA GRAF BERARAH

NANDA BUNGA

0 3 0 4 0 1 0 4 2 Y



UNIVERSITAS INDONESIA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

DEPARTEMEN MATEMATIKA

DEPOK

2009

BANYAK SIRKUIT EULER PADA GRAF BERARAH

**Skripsi ini diajukan sebagai salah satu syarat
untuk memperoleh gelar Sarjana Sains**

Oleh:

NANDA BUNGA

0 3 0 4 0 1 0 4 2 Y



DEPOK

2009

SKRIPSI : BANYAK SIRKUIT EULER PADA GRAF BERARAH
NAMA : NANDA BUNGA
NPM : 030401042Y

SKRIPSI INI TELAH DIPERIKSA DAN DISETUJUI
DEPOK, 1 DESEMBER 2009

DRA. DENNY R. SILABAN, M.KOM.

PEMBIMBING I

DR. KIKI ARIYANTI S.

PEMBIMBING II

Tanggal Lulus Ujian Sidang Sarjana: 16 Desember 2009

Penguji I : Dra. Denny R. Silaban, M.Kom.

Penguji II : Dhian Widya, S.Si., M.Kom.

Penguji III : Dra. Siti Aminah, M.Kom.

Kata Pengantar

Puji syukur kepada Allah SWT atas segala berkah dan karunia-Nya sehingga penulisan skripsi ini selesai tepat waktu. Selesaiannya skripsi ini tidak lepas dari bimbingan, dorongan, dan bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Orang tua penulis, (“Papa, Mama *Thanks for all your support, luv you both.*”)
2. Kakak – kakak penulis, Da Aldo dan Da Nikko terima kasih doa dan dukungannya.
3. Ibu Denny R. Silaban dan Ibu Kiki A. Sugeng selaku pembimbing skripsi penulis yang dengan telah sabar meluangkan waktunya untuk memberi bimbingan, saran, pengarahan, dan bantuan lainnya sehingga skripsi ini dapat diselesaikan dengan baik.
4. Ibu Denny R. Silaban selaku pembimbing akademis penulis selama menempuh kuliah di Departemen Matematika UI.
5. Pak Djati Kerami, Bu Siti Aminah, dan Bu Dhian Widya yang selalu meluangkan waktu untuk menghadiri SIG 1, SIG 2, dan menjadi penguji untuk kolokium penulis.

6. Seluruh dosen Departemen Matematika atas segala ilmu yang telah penulis terima selama menjadi mahasiswa Matematika UI.
7. Mba Santi dan Pak Saliman terima kasih yang selalu menjadi tempat untuk bertanya oleh penulis dalam hal administrasi perkuliahan,
8. Seluruh karyawan Matematika UI, baik TU maupun Perpustakaan Departemen Matematika yang telah banyak memberikan bantuannya.
9. Kakak ipar penulis, kak Yuli terima kasih atas doa dan dukungannya, terutama untuk 2 keponakanku tersayang kakak Zee dan adik Keiko.
10. Keluarga Jimelvi, terima kasih atas dukungan dan komputer yang telah dipinjamkan kepada penulis selama kuliah, serta Greffio terima kasih telah memperbolehkan penulis menggunakan printernya.
11. Nenek dan Keluarga Besar Nursehan (“Terima kasih doa dan dukungannya”).
12. Agha, terima kasih atas dukungan dan semangat yang diberikan kepada penulis.
13. Teman – teman seperjuangan penulis Alberta, Anggha, Novi, Inne, Anggi, TH, skripsi kita selesai juga.
14. Keluarga besar Matematika 2004, terutama untuk las, Spina terima kasih atas bantuannya.
15. Untuk keluarga besar distro d4ngk4l : Zhesa, Kiki, Rhe2, Jerry, Utha, Avi, Tian, Risman, Rian, Bunda, Uthi, dan semua yang tak bisa disebutkan satu per satu “Terima kasih atas dukungannya” .

16. Abang Goiz, Silvy, Nina, Astri, Anton, Yana terima kasih (“walaupun kita sudah jarang berkumpul lagi, tetapi kalian semua tetap *support* aku melalui telepon”).
17. Teman – teman angkatan 2001, 2002, 2003, 2005, 2006.
18. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah membantu baik secara langsung maupun tidak langsung dalam pembuatan skripsi ini, penulis mengucapkan terima kasih.

Semoga Allah SWT, memberikan rahmat-Nya sebagai balasan atas bantuan yang diberikan kepada penulis. Penulis mohon maaf apabila terdapat kekurangan dalam skripsi ini. Semoga skripsi ini bermanfaat bagi pembaca.

Depok, Desember 2009

Penulis

Nanda Bunga

ABSTRAK

Salah satu permasalahan dalam genetika adalah mencari barisan DNA lengkap dari jaringan tertentu. Metode yang dapat digunakan untuk masalah ini adalah metode *Sequencing by Hybridization* (SBH). Dalam SBH terdapat dua tahapan yaitu tahap biokimia dan tahapan komputasional. Pada tahapan biokimia akan diperoleh l -spektrum. Selanjutnya, l -spektrum disusun untuk memperoleh barisan DNA lengkap pada tahapan komputasional. Pencarian barisan DNA lengkap dapat dimodelkan dengan masalah pencarian sirkuit Euler pada graf DNA. Dalam skripsi ini akan dibahas Teorema Matriks Pohon untuk menentukan banyaknya pohon rentangan terhadap simpul v_i pada graf berarah bisa dengan menggunakan kofaktor matriks. Selanjutnya, dibahas Teorema BEST yang digunakan untuk menghitung banyaknya sirkuit Euler pada sembarang graf berarah dengan menggunakan banyaknya pohon rentangan berarah terhadap suatu simpul v_i serta derajat simpul-simpulnya pada graf berarah.

Kata kunci: graf de Bruijn; graf DNA; *Sequencing by Hybridization*; sirkuit

Euler; pohon rentangan berarah terhadap simpul v_i .

vi + 42 hlm.

Bibliography: 4 (1996-2008)

DAFTAR ISI

	Halaman
KATA PENGANTAR	i
ABSTRAK	iv
DAFTAR ISI	v
DAFTAR GAMBAR	vi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan.....	3
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Sistematika Penulisan	3
BAB 2 LANDASAN TEORI	5
2.1 <i>Sequencing by Hybridization</i>	5
2.2 Graf Berarah	8
2.3 Matriks	15
2.4 Graf de Bruijn	18
BAB 3 Teorema BEST.....	22
3.1 Teorema Matriks Pohon	22
3.2 Teorema BEST	32
BAB 4 KESIMPULAN	41
DAFTAR PUSTAKA	42

DAFTAR GAMBAR

Gambar	Halaman
2.1 Contoh Graf berarah	9
2.2 Contoh jalan (jalur atau lintasan).....	11
2.3 (a) Graf lintasan berarah, (b) Graf bukan lintasan berarah, (c) Graf lingkaran berarah, (d) Graf bukan lingkaran berarah	11
2.4 (a) Graf dengan sirkuit Euler tunggal, (b) Graf dengan sirkuit Euler tidak tunggal	12
2.5 (a) Graf berarah, (b) Pohon, (c) Pohon rentangan	13
2.6 (a) Graf berarah G , (b) Pohon rentangan terhadap simpul v_1	14
2.7 Graf berarah G	16
2.8 Graf de brujin dengan 1 sirkuit Euler	19
2.9 Graf de brujin dengan 2 sirkuit Euler	20
3.1 (a) Graf berarah G , (b) Graf $G-v_1v_2$, (c) Graf G/v_1v_2	23
3.2 (a) Graf berarah G , (b) Pohon rentangan berarah terhadap simpul v_1 di G	32
3.3 Pohon rentangan berarah terhadap simpul v_1 di G	38
3.4 Graf berarah G	39

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bioinformatika merupakan kajian keilmuan yang menggabungkan biologi, matematika dan komputasi. Salah satu yang menjadi permasalahan dalam bioinformatika adalah bagaimana mengenali urutan struktur suatu barisan DNA. Barisan DNA terdiri dari empat karakter huruf yaitu *A (Adenine)*, *G (Guanine)*, *T (Thymine)*, dan *C (Cytosine)*. Keempat karakter tersebut merupakan asam nukleat. Salah satu metode yang dapat digunakan untuk mengenali urutan struktur barisan DNA adalah *Sequencing by Hybridization (SBH)*. Metode SBH terbagi menjadi dua tahapan, yaitu tahapan biokimia dan tahapan komputasional. Tahapan biokimia merupakan tahapan awal dimana DNA diekstrak sehingga diperoleh fragmen-fragmen dari asam nukleat dengan panjang tertentu. Fragmen ini disebut oligonukleotida dan himpunan dari semua oligonukleotida disebut spektrum. Tahapan komputasional merupakan tahapan terakhir dimana barisan DNA dicari dengan menggunakan spektrum yang diperoleh pada tahapan biokimia.

Apabila diberikan jaringan DNA, maka DNA itu akan diekstrak pada tahapan biokimia dan akan diperoleh fragmen-fragmen DNA dengan panjang

tertentu. Himpunan dari fragmen-fragmen DNA atau spektrum, akan digunakan untuk tahapan selanjutnya yaitu tahapan komputasional.

Pada tahapan komputasional, untuk penentuan urutan barisan DNA yang utuh dari fragmen-fragmen yang telah didapatkan pada tahapan biokimia dapat digunakan teori graf sebagai alat bantu. Untuk dapat melakukan hal itu, akan dibentuk suatu graf dengan busur berarah sebanyak elemen pada spektrum. Dalam hal ini, setiap busur berarah menghubungkan dua simpul yang dibentuk dengan aturan tertentu. Graf yang terbentuk mengikuti aturan pembentukan graf khusus yang disebut sebagai graf de Bruijn. Aturan pembentukan simpul dan busur ditetapkan dengan aturan yang mengakibatkan kemungkinan rangkaian DNA yang utuh dapat diperoleh dengan mencari sirkuit Euler pada graf tersebut. Oleh karena itu, Masalah pencarian barisan DNA yang utuh dapat dimodelkan menjadi masalah pencarian sirkuit Euler pada graf DNA. Sirkuit Euler merupakan jalur tertutup yang melewati semua busur pada graf.

Graf DNA yang dibangun dari spektrum DNA untuk memperoleh barisan DNA yang utuh, kemudian masalah pencarian barisan DNA dimodelkan menjadi masalah pencarian sirkuit Euler dan menghitung berapa banyak sirkuit Euler pada graf DNA. Teorema yang akan dibahas, yaitu Teorema BEST, menetapkan banyaknya sirkuit Euler pada sembarang graf berarah.

1.2 Permasalahan

Permasalahan dalam skripsi ini adalah berapa banyakkah sirkuit Euler pada sembarang graf berarah?

1.3 Tujuan

Skripsi ini bertujuan untuk membahas berapa banyak sirkuit Euler yang dapat dibentuk dari sembarang graf berarah menggunakan Teorema BEST.

1.4 Batasan Masalah

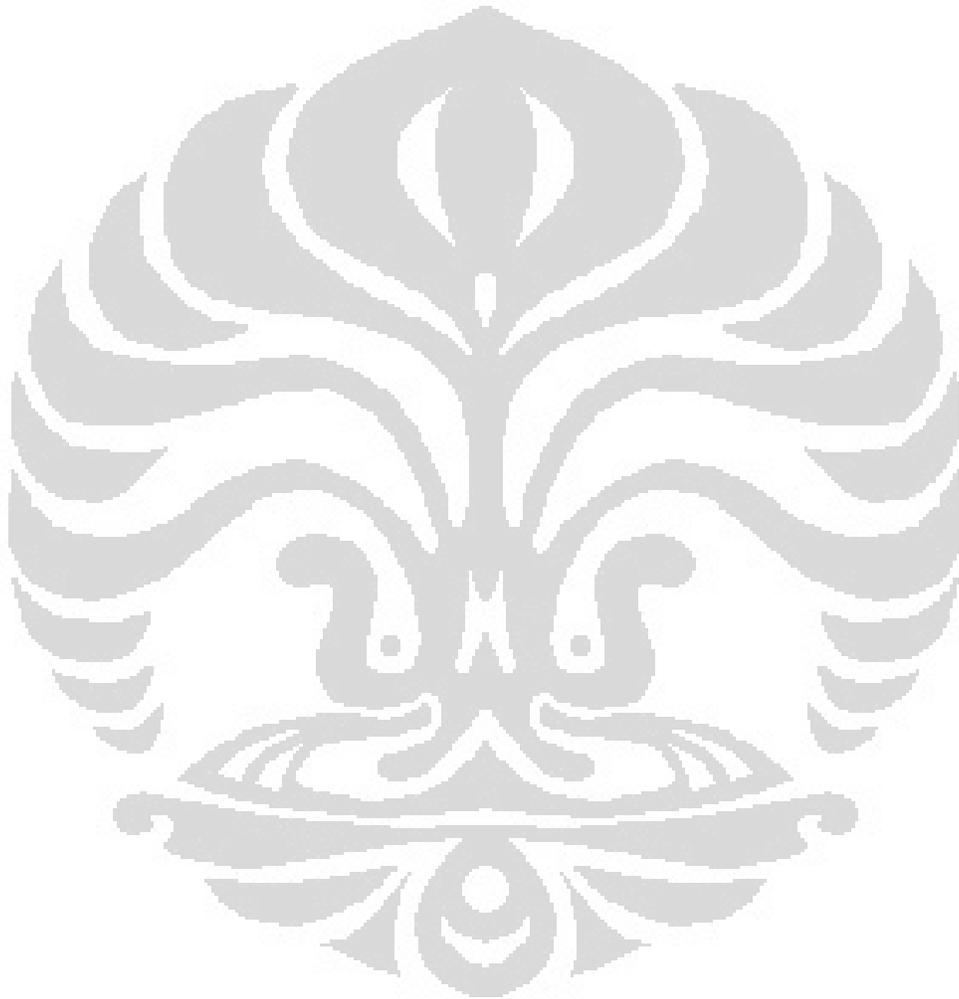
Dalam skripsi ini, hanya dibahas mengenai berapa banyak sirkuit Euler yang dapat dibentuk dari sembarang graf berarah. Bagaimana membentuk sirkuit Eulernya tidak dibahas.

1.5 Sistematika Penulisan

Skripsi ini terbagi menjadi empat bab. Dalam Bab II, akan dibahas mengenai teori graf berarah dan metode *Sequencing by Hybridization* sebagai salah satu metode untuk penentuan urutan suatu barisan DNA. Bab III membahas Teorema Matriks Pohon untuk menentukan banyaknya pohon

rentangan terhadap simpul v_1 pada graf berarah bisa dengan menggunakan kofaktor matriks Laplacian dan Teorema BEST yang dapat digunakan untuk menghitung banyaknya sirkuit Euler pada sembarang graf berarah G .

Kesimpulan dari skripsi ini akan diberikan pada Bab IV.



BAB II

LANDASAN TEORI

Pada bab ini, akan diberikan teori dasar dan definisi yang berhubungan dengan Teorema BEST yang akan digunakan pada pada bab III. Akan diberikan juga metode *Sequencing by Hybridization* (SBH) yang merupakan metode dalam penentuan (urutan) barisan DNA melalui proses perakitan sekumpulan fragmen-fragmen DNA yang tumpang tindih (*overlapping*).

2.1 Sequencing by Hybridization

DNA tersusun dari molekul-molekul yang lebih sederhana yang disebut *nukleotida*. Suatu rantai DNA tersusun atas fosfat, gula dan 3×10^9 pasang nukleotida. Nukleotida yang menyusun DNA terdiri dari empat unsur yang direpresentasikan oleh: *A* (*Adenine*), *G* (*Guanine*), *T* (*Thymine*) dan *C* (*Cytosine*). Molekul DNA seperti dua pita yang terangkai (*double helix*) yang dihubungkan oleh suatu ikatan hidrogen. Jika suatu saat DNA itu dipanaskan maka ikatan hidrogen akan hilang dan menyisahkan dua pita tunggal. Satu pita tunggal dibentuk oleh barisan nukleotida. Barisan nukleotida atau yang dikenal dengan barisan DNA tidak mudah dikenali. Dibutuhkan suatu proses untuk mengenali barisan DNA.

Salah satu metode untuk mengenali barisan DNA adalah *Sequencing by Hybridization* (SBH) yang diperkenalkan pada tahun 1988 [Pev01]. Metode SBH ini terbagi menjadi dua tahap, tahap biokimia dan tahap komputasional. Pada tahap biokimia, dihasilkan potongan-potongan pendek barisan DNA dengan panjang tertentu yang disebut oligonukleotida atau fragmen. Himpunan dari semua fragmen disebut spektrum. Misalkan diberikan suatu DNA, maka DNA itu akan diekstrak dan direpresentasikan dalam barisan nukleotida yang disebut barisan DNA. Pada tahap biokimia akan diekstrak himpunan fragmen berukuran l yang terkandung pada barisan DNA, namun urutannya belum diketahui pada barisan tersebut. Untuk suatu barisan s dengan panjang n , terdapat sebanyak $n - l + 1$ fragmen dengan panjang l karena untuk barisan dengan panjang n jika dibentuk suatu fragmen pendek dengan panjang l dimana barisannya harus berurut maka akan ada sebanyak $n - l$ fragmen ditambah dengan satu fragmen dengan panjang l yang terakhir. Sehingga banyaknya elemen pada himpunan fragmen (spektrum) adalah $n - l + 1$ dengan kondisi bahwa tidak ada fragmen yang muncul lebih dari sekali dalam barisan.

Pada tahapan komputasional, fragmen-fragmen pendek ini disusun untuk membentuk barisan DNA. Terdapat beberapa metode untuk penyusunan fragmen menjadi barisan DNA. Di antaranya dengan menggunakan metode TSP (*Traveling Salesman Problem*) yakni dengan mendefinisikan overlap (u_i, u_j) sebagai panjang dari bagian paling kanan u_i yang

sama dengan bagian paling kiri u_j , bentuk graf dengan n simpul yang mewakili n untai u_1, u_2, \dots, u_n dan busurnya adalah overlap (u_i, u_j) antara simpul s_i dan simpul s_j dan solusinya adalah jarak terpendek untuk mengunjungi tiap simpul tepat satu kali. Metode lainnya adalah SSP (*Shortest Superstring Problem*) yakni suatu masalah dimana jika diberikan sekumpulan untai s_1, s_2, \dots, s_n , rangkai untai–untai tersebut sehingga menghasilkan satu untai s yang mengandung s_1, s_2, \dots, s_n sebagai subuntai, sedemikian sehingga panjang dari s minimum. Pada skripsi ini akan dibahas metode pembentukan barisan DNA menggunakan sirkuit Euler.

Pada metode menggunakan sirkuit Euler akan digunakan bantuan teori graf untuk mendapatkan barisan DNA yang utuh. Untuk dapat melakukan hal itu, akan dibentuk suatu graf dengan busur berarah sebanyak elemen pada spektrum dan busur berarah yang menghubungkan antar dua simpul dibentuk dengan aturan tertentu. Graf tersebut disebut graf de Bruijn. Pembentukan busur dan simpul ditetapkan sedemikian rupa sehingga kemungkinan barisan DNA dapat diperoleh dengan mencari sirkuit Euler pada graf DNA. Oleh karena itu, masalah mencari barisan DNA dari spektrum dimodelkan menjadi masalah mencari sirkuit Euler pada graf.

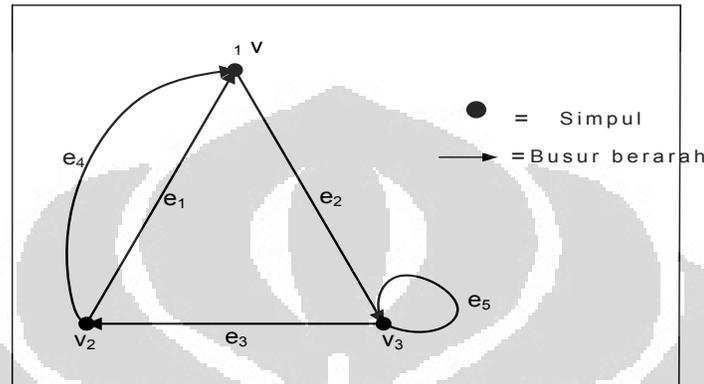
2.2 Graf Berarah

Graf berarah atau *digraph* $G = (V, E)$ adalah graf dengan himpunan tak kosong simpul $V(G)$ dengan himpunan **busur berarah** $E(G)$. Busur berarah $e \in E$ dapat merupakan pasangan terurut (u, v) dimana $u, v \in V$. Graf berarah dapat direpresentasikan secara visual dengan suatu **simpul** dapat digambarkan sebagai lingkaran kecil atau titik sedangkan busur berarah $(u, v) \in E$ digambarkan sebagai garis berarah dari u ke v dengan u disebut **titik pangkal** dan v disebut **titik ujung** dari busur berarah tersebut. Untuk pembahasan selanjutnya kata busur merujuk kepada busur berarah dan graf merujuk kepada graf berarah. Misalkan $e = (u, v)$ dimana $u, v \in V$ dan busur e menghubungkan simpul u dengan simpul v , maka u dan v dikatakan saling **bertetangga** (*adjacent*). Busur e dikatakan **hadir** (*incident*) pada simpul u dan simpul v . Suatu graf dikatakan memiliki **busur sejajar** jika terdapat dua atau lebih busur berarah yang memiliki titik pangkal dan titik ujung yang sama. Busur yang menghubungkan suatu simpul dengan dirinya sendiri disebut **gelang** (*loop*).

Banyaknya busur yang hadir pada simpul v dikatakan **derajat** (*degree*) dan dilambangkan dengan $d(v)$. Derajat pada graf berarah dibagi dua, yaitu **derajat masuk** (*indegree*) dan **derajat keluar** (*outdegree*) sesuai dengan banyaknya busur yang mempunyai arah masuk atau keluar dari suatu simpul yang secara berturut-turut dinotasikan dengan $d^-(v)$ dan $d^+(v)$. Jumlah

derajat semua simpul pada suatu graf adalah genap, yaitu dua kali jumlah sisi pada graf tersebut. Jika $G = (V, E)$ merupakan suatu graf, maka dapat ditulis:

$$\sum_{v \in V} d(v) = 2|E|.$$



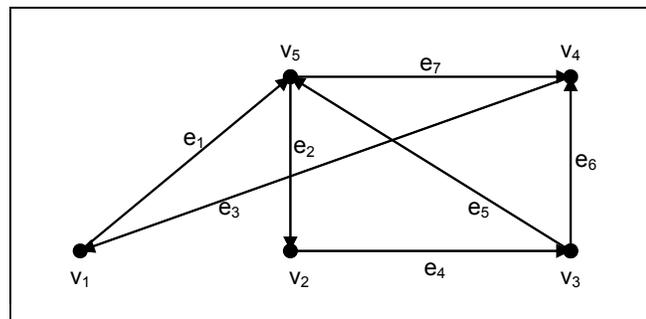
Gambar 2.1 Contoh Graf berarah

Pada Gambar 2.1, diberikan contoh graf berarah dengan tiga simpul $V = \{v_1, v_2, v_3\}$ dan lima busur $E = \{e_1, e_2, e_3, e_4, e_5\}$. Simpul v_1 dan v_3 dikatakan saling bertetangga karena ada busur e_2 yang menghubungkan kedua simpul tersebut, begitu juga dengan simpul v_2 dan v_3 . Busur e_2 dikatakan hadir pada simpul v_1 dan v_3 sedangkan e_3 dikatakan hadir pada simpul v_3 dan v_2 . Busur e_3 menghubungkan simpul v_3 dan v_2 dimana v_3 adalah titik pangkal dan v_2 adalah titik ujung. Busur e_1 dan e_4 adalah busur sejajar karena kedua busur itu sama-sama memiliki titik pangkal v_2 dan titik ujung v_1 . Busur e_5 menghubungkan simpul v_3 dengan dirinya sendiri disebut gelang (*loop*). Derajat masuk dari simpul v_1 adalah dua, karena busur yang hadir dan mempunyai arah masuk pada simpul v_1 adalah busur e_1 dan e_4 . Derajat keluar dari simpul v_1 adalah satu, karena busur yang hadir dan mempunyai arah keluar pada simpul v_1

adalah busur e_2 .

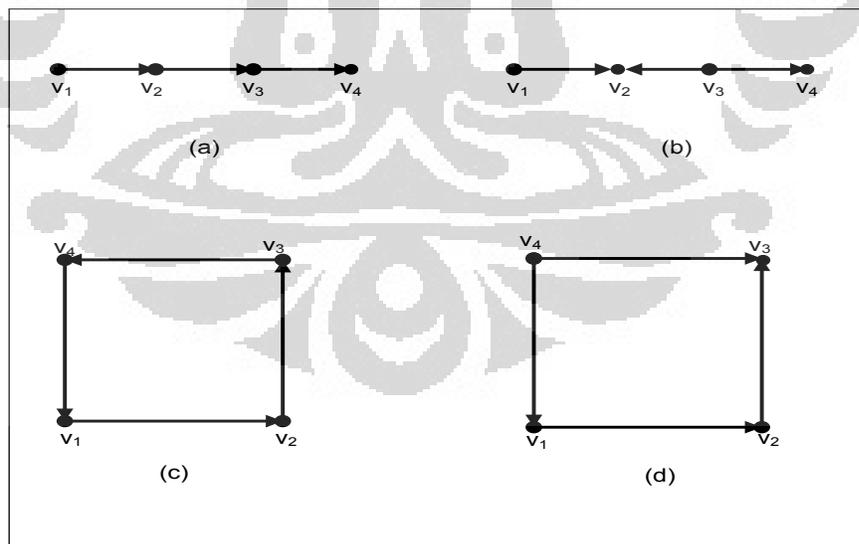
Jalan berarah (*diwalk*) adalah sembarang perjalanan berarah pada graf yang menghasilkan suatu daftar barisan dari simpul dan busur yang membentuk suatu sambungan yang tidak putus. **Jalur berarah** (*ditrail*) adalah suatu jalan berarah dengan tidak ada busur yang berulang. **Lintasan berarah** (*dipath*) P_n merupakan suatu jalur dimana tidak ada simpul yang berulang dan merupakan suatu barisan n simpul dan $n-1$ busur $v_1, e_1, v_2, \dots, v_{n-1}, e_{n-1}, v_n$ dengan $e_i = \{v_i, v_{i+1}\} \in E, 1 \leq i \leq n-1$. Panjang jalan (jalur atau lintasan) didefinisikan sebagai banyaknya busur pada jalan (jalur atau lintasan) tersebut. Jika pada suatu jalur ada busur e_n yang menghubungkan v_1 dengan v_n maka jalur disebut *tertutup*. Jalur tertutup disebut sirkuit, sedangkan lintasan tertutup disebut **lingkaran**. Lintasan berarah tertutup dapat disebut sebagai **lingkaran berarah** (*dicycle*). Panjang lingkaran atau sirkuit didefinisikan sebagai banyaknya busur pada lingkaran atau sirkuit tersebut. Suatu graf dikatakan **terhubung** (*connected*), apabila untuk setiap pasang simpul u dan v di G , terdapat lintasan tak berarah dari simpul u menuju ke v .

Pada Gambar 2.2 diberikan contoh graf berarah dengan lima simpul $V = \{v_1, v_2, v_3, v_4, v_5\}$ dan tujuh busur berarah $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$. Jalan $(v_1, e_1, v_5, e_2, v_2, e_4, v_3, e_5, v_5, e_7, v_4)$ adalah suatu jalur dengan panjang lima dari simpul v_1 menuju simpul v_4 . Akan tetapi jalur ini bukan lintasan karena simpul v_5 dilalui dua kali. Jalan $(v_1, e_1, v_5, e_2, v_2, e_4, v_3, e_6, v_4)$ adalah suatu lintasan dengan lima simpul (P_5) dan panjang lintasan empat.



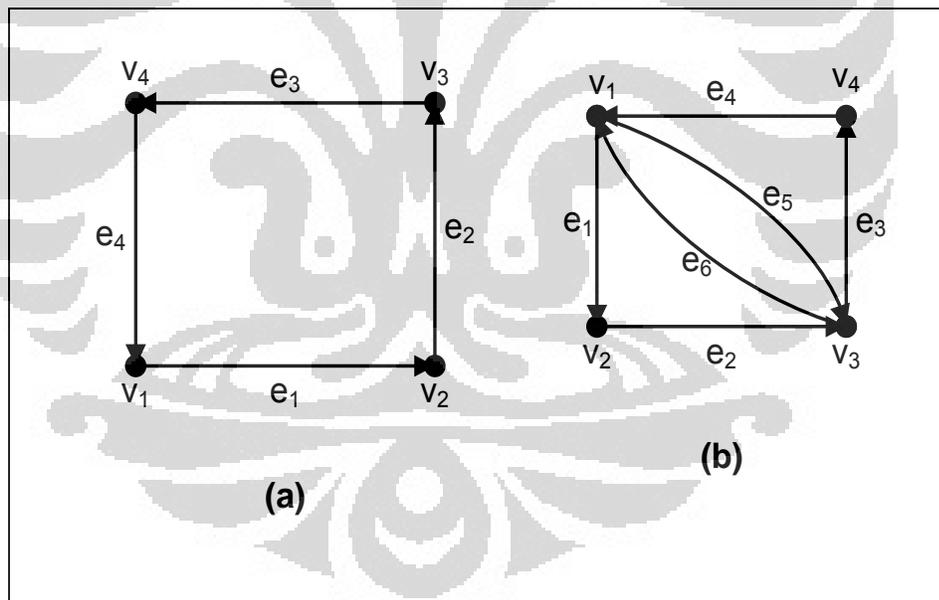
Gambar 2.2 Contoh jalan (jalur atau lintasan)

Pada Gambar 2.3, diberikan contoh lintasan dengan empat simpul (P_4) dan panjang lintasan tiga (a), sedangkan (b) bukan merupakan suatu lintasan karena tidak dapat dibuat suatu barisan berarah dengan empat simpul dan tiga busur. Gambar 2.3 (c) merupakan lingkaran dengan empat simpul (C_4) dan panjang lingkaran empat, sedangkan (d) bukan merupakan suatu lingkaran berarah.



Gambar 2.3 (a) Graf lintasan berarah, (b) Graf bukan lintasan berarah, (c) Graf lingkaran berarah, (d) graf bukan lingkaran berarah

Jalur Euler adalah jalan yang melewati setiap busur pada graf tepat satu kali. **Sirkuit Euler** adalah jalur tertutup yang melewati semua busur pada graf. Tidak setiap graf berarah G mempunyai sirkuit Euler. Jika suatu graf G mempunyai sirkuit Euler, sirkuit Euler pada G bisa tidak tunggal. Sebagai contoh pada Gambar 2.2 tidak mempunyai sirkuit Euler karena tidak dapat dibuat suatu perjalanan yang mengunjungi setiap busur tepat satu kali. Gambar 2.4 (a) sirkuit Euler tunggal yaitu $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_1)$. Pada Gambar 2.4 (b) terdapat dua sirkuit Euler yaitu $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_1, e_5, v_3, e_6, v_1)$, dan $(v_1, e_1, v_2, e_2, v_3, e_6, v_1, e_5, v_3, e_3, v_4, e_4, v_1)$.



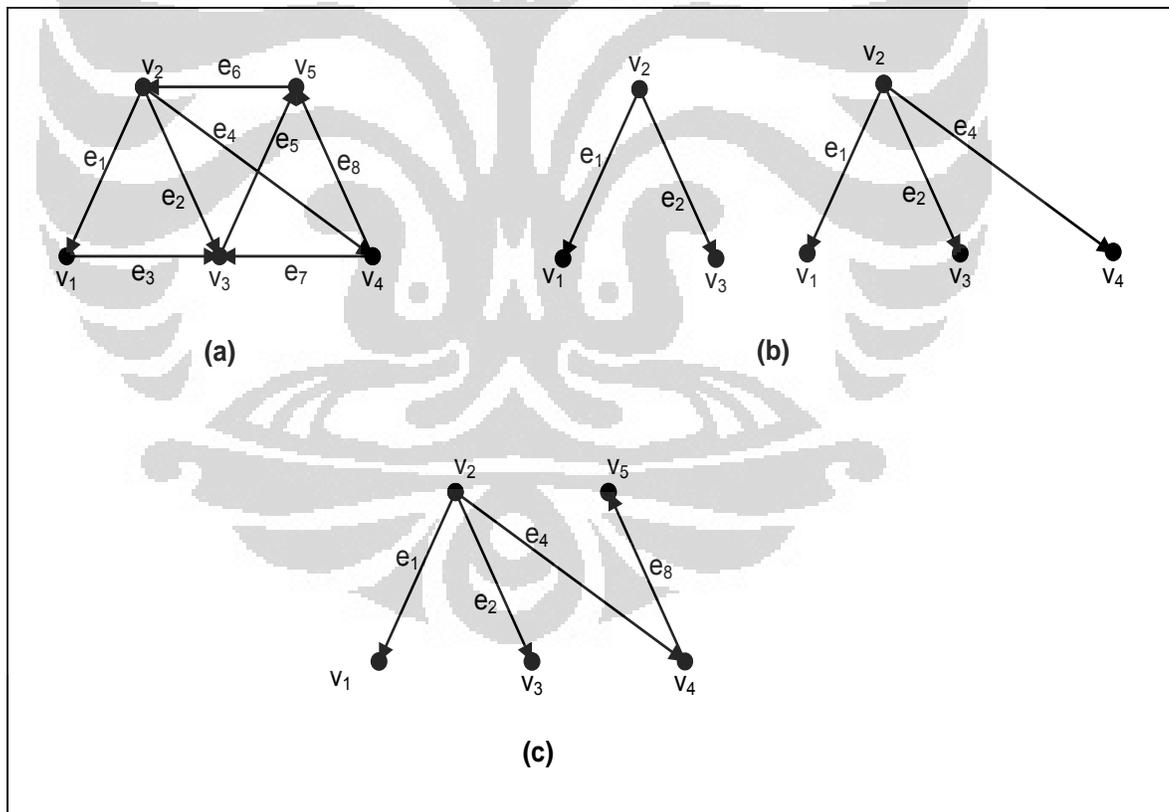
Gambar 2.4 (a) Graf dengan sirkuit Euler tunggal, (b) Graf dengan sirkuit Euler tidak tunggal

Teorema 2.1 [Bol98]

Graf berarah G memiliki sirkuit Euler jika dan hanya jika G terhubung dan setiap simpul memiliki derajat masuk dan derajat keluar sama.

Bukti dari teorema ini dapat dilihat pada [Bol98].

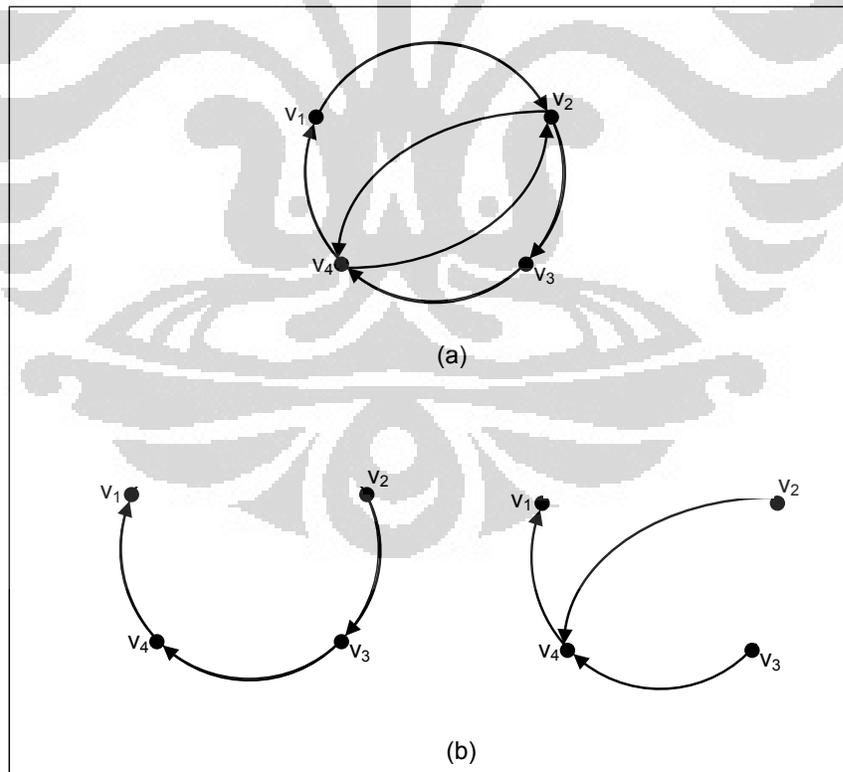
Pohon (Tree) adalah graf terhubung yang tidak mengandung lingkaran, dengan banyak simpul n dan busur $n-1$. **Pohon rentangan (Spanning tree)** adalah suatu subgraf dari graf G yang mengandung semua simpul dari G (tidak perlu memuat seluruh busur dari G), dan merupakan suatu pohon.



Gambar 2.5 (a) Graf berarah, (b) Pohon, dan (c) Pohon rentangan

Sebagai contoh Gambar 2.5 (a) diberikan graf berarah G dengan lima simpul $V = \{v_1, v_2, v_3, v_4, v_5\}$ dan delapan busur $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$. Pohon yang dapat dibuat dari graf G diberikan pada Gambar 2.5 (b) dan pohon rentangan diberikan pada Gambar 2.5 (c).

Pohon rentangan berarah terhadap simpul v_i adalah pohon rentangan dengan aturan untuk setiap $j \neq i$, terdapat lintasan berarah dari simpul v_j ke simpul v_i . Sebagai contoh pada Gambar 2.6 (a), diberikan graf berarah G dengan empat simpul $V = \{v_1, v_2, v_3, v_4\}$ dan enam busur $E = \{e_1, e_2, e_3, e_4, e_5, e_6\}$, sementara pada Gambar 2.6 (b) diberikan dua pohon rentangan berarah terhadap simpul v_1 dari graf G .



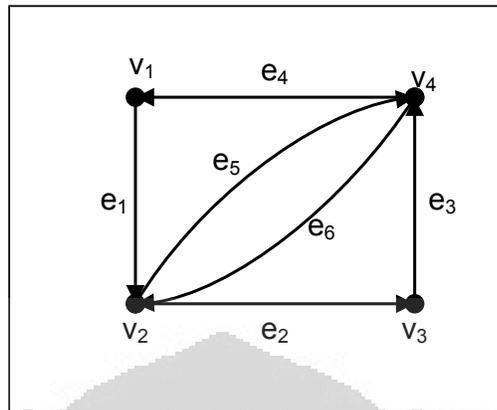
Gambar 2.6 (a) Graf berarah G , (b) Pohon rentangan terhadap simpul v_1

2.3 Matriks

Informasi dari graf berarah juga dapat diberikan dalam berbagai matriks, antara lain matriks ketetanggaan (*adjacency*), matriks diagonal dan matriks Laplacian. **Matriks ketetanggaan** adalah matriks yang berisi daftar banyaknya busur yang menghubungkan setiap pasang simpul di graf berarah. Untuk matriks $n \times n$ di mana n adalah banyaknya simpul pada graf G , jika a_{ij} adalah jumlah busur yang meninggalkan simpul v_i dan masuk ke simpul v_j pada graf G , maka a_{ij} adalah entri dari matriks pada baris ke- i dan kolom ke- j . Jika pada G tidak ada gelang maka diagonal pada matriks ketetanggaan adalah nol.

Matriks yang mendaftarkan derajat masuk pada setiap simpul dalam graf G merupakan **matriks diagonal**. Selain diagonal semua entri bernilai nol.

Matriks Laplacian adalah matriks yang didapat dengan menggabungkan kedua informasi yang diberikan oleh kedua matriks sebelumnya, yaitu matriks ketetanggaan $A(G)$ dan matriks diagonal $D(G)$. Matriks Laplacian $L(G) = D(G) - A(G)$.



Gambar 2.7 Graf berarah G

Sebagai contoh, maka matriks ketetanggaan dari G (Gambar 2.7) adalah

$$A(G) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Perhatikan bahwa semua elemen diagonal dari $A(G)$ bernilai nol, karena tidak ada gelang di graf G. Entri a_{12} bernilai 1 karena simpul v_1 hanya mengirimkan satu busur berarah ke simpul v_2 . Entri a_{23} dan a_{24} bernilai 1 karena simpul v_2 hanya mengirimkan satu busur berarah ke simpul v_3 dan simpul v_4 . Entri a_{34} bernilai 1 karena simpul v_3 hanya mengirimkan satu busur berarah ke simpul v_4 . Entri a_{41} dan a_{42} bernilai 1 karena simpul v_4 hanya mengirimkan satu busur berarah ke simpul v_1 dan simpul v_2 . Selain itu untuk a_{13} , a_{14} , a_{21} , a_{31} , a_{33} , dan a_{43} bernilai nol karena tidak ada busur yang menghubungkan simpul.

Matriks diagonal $D(G)$ dari G adalah

$$D(G) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

Sementara matriks Laplacian $L(G)$ dari G adalah

$$L(G) = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 2 & -1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & -1 & 0 & 2 \end{bmatrix}$$

Ada beberapa karakteristik dari matriks Laplacian yang berlaku untuk semua matriks pada graf berarah. Pertama, diagonal masih merupakan derajat masuk setiap simpul di graf G . Dalam contoh ini tidak ada gelang, sehingga diagonal tetap sama. Selain itu, bernilai -1 untuk simpul v_i yang mengirim busur ke v_j dimana i, j adalah entri dari matriks.

Banyaknya pohon rentangan berarah terhadap suatu simpul pada graf berarah G dapat dihitung dari matriks Laplacian. Penghitungan ini melibatkan kofaktor dari matriks. Kofaktor dari suatu matriks didapatkan dengan mengeluarkan baris ke- i dan kolom ke- i dari matriks Laplacian, dan kemudian cari nilai determinan dari sisa matriks Laplacian tersebut. Maka dapat ditunjukkan sebagai berikut:

$$k = \det_{i,i} L(G)$$

Mempertimbangkan matriks Laplacian untuk graf G dari Gambar 2.7, kofaktor baris pertama dan kolom pertama yang didapat adalah matriks sebagai berikut:

$$\begin{bmatrix} 2 & -1 & -1 \\ 0 & 1 & -1 \\ -1 & 0 & 2 \end{bmatrix}$$

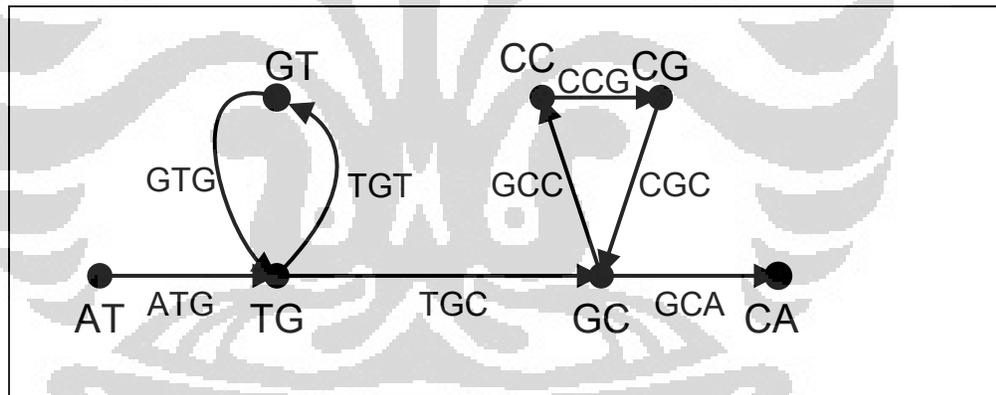
Maka hasil determinan dari matriks ini menghasilkan nilai $k = 2$.

2.4 Graf de Bruijn

Graf de Bruijn adalah graf berarah yang simpul-simpulnya dinyatakan dengan barisan simbol-simbol alfabet, dan busur-busurnya dinyatakan dengan barisan-barisan yang tumpang tindih (*overlapping*).

Aturan pembentukan graf de Bruijn ini digunakan untuk pembentukan graf DNA. Pembangunan graf DNA tergantung pada fragmen DNA dengan panjang l . Setiap simpul memiliki panjang label fragmen $l-1$ dan busur berarah berada diantara kedua simpul yang memiliki panjang label fragmen l , dimana simbol-simbol dengan panjang label fragmen $l-1$ berasal dari simpul. Dengan demikian, simbol-simbol yang tumpang tindih diantara simpul-simpul dengan panjang label fragmen $l-1$ merupakan busur. Sebagai contoh akan dibangun graf DNA dengan menggunakan aturan graf de Bruijn. Dengan

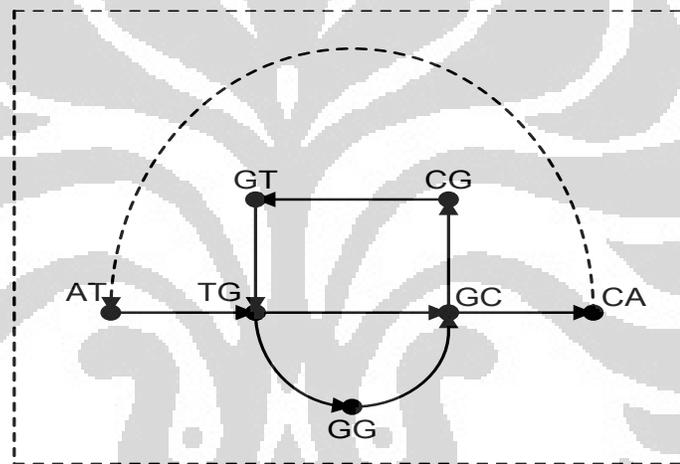
himpunan fragmen $S = \{ ATG, TGT, GTG, TGC, GCC, CCG, CGC, GCA \}$ dimana fragmen dari DNA tersebut memiliki panjang label tiga. Dengan demikian, busur memiliki panjang label fragmen tiga dan simpul memiliki panjang label fragmen dua. Dengan aturan tumpang-tindih pada graf de Bruijn, busur ATG akan menghubungkan simpul AT dan simpul TG . Ini berlaku untuk setiap busur. Maka setelah seluruh proses selesai, graf DNA yang dihasilkan dapat dilihat pada Gambar 2.8 dengan barisan DNA adalah $ATGTGCCGCA$. Dalam hal ini hanya akan terbentuk sirkuit Euler unik dengan menghubungkan simpul awal dan akhir yaitu simpul AT dan simpul CA .



Gambar 2.8 Graf de Bruijn dengan 1 sirkuit Euler

Sekarang misalkan akan dibangun graf de Bruijn yang lain, dengan himpunan fragmen $S = \{ ATG, TGG, TGC, GTG, GGC, GCA, GCG, CGT \}$, dimana fragmen dari DNA tersebut memiliki panjang label fragmen tiga. Dengan demikian, busur memiliki panjang label fragmen tiga dan simpul memiliki panjang label fragmen dua. Dengan aturan tumpang-tindih pada graf de Bruijn, busur ATG akan menghubungkan simpul AT dan simpul TG .

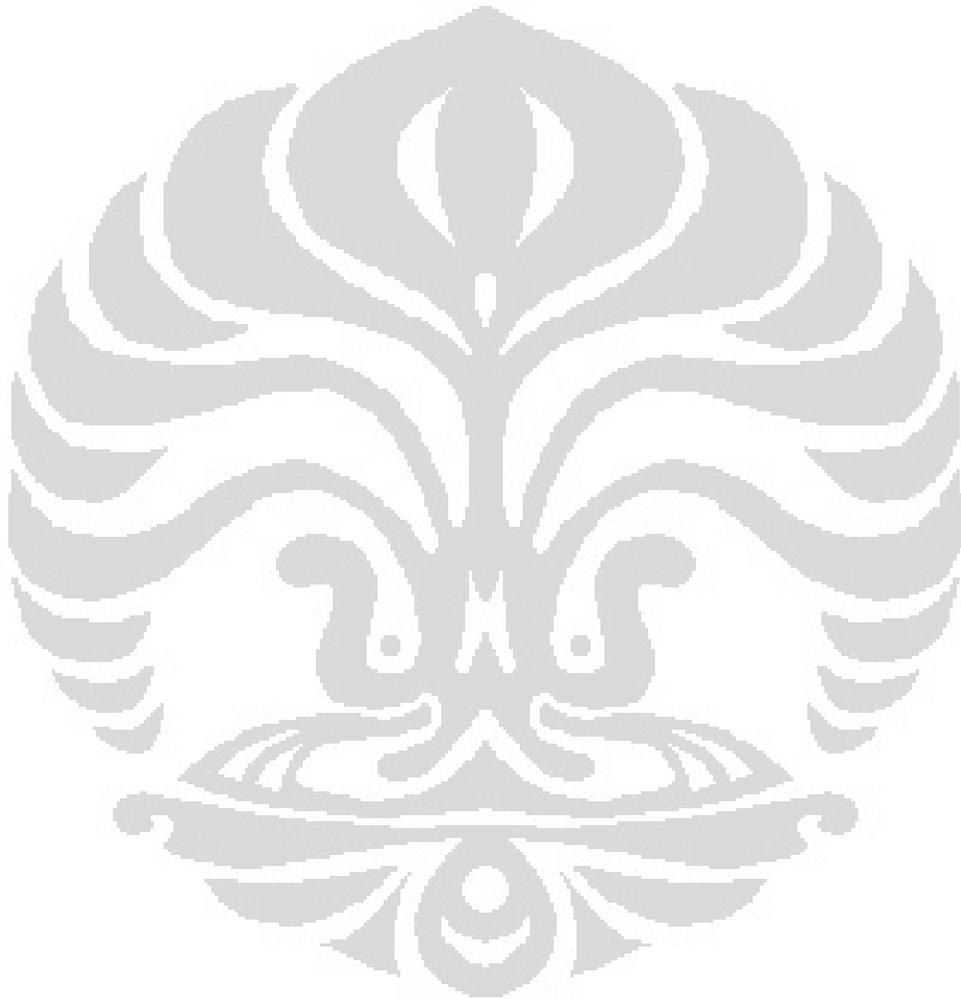
Ini berlaku untuk setiap busur. Maka setelah proses selesai, graf yang dihasilkan ternyata lebih kompleks dari graf sebelumnya. Secara khusus, terdapat dua jalur Euler dengan barisan yang berbeda $ATGGCGTGCA$ dan $ATGCGTGGGCA$, yang dapat dilihat pada Gambar 2.9 dan perlu dicatat bahwa disini ditambahkan busur yang menghubungkan simpul pertama dengan simpul terakhir (AT dan CA) agar membuat sirkuit Euler berarah. Dengan demikian sirkuit Euler tidak unik.



Gambar 2.9 Graf de Bruijn dengan 2 sirkuit Euler

Oleh karena itu, jika diberikan sembarang barisan DNA maka jalur Euler yang dihasilkan tidak unik, sehingga mengakibatkan sirkuit Euler yang ditemukan juga tidak unik. Maka pada skripsi ini akan dibahas berapa banyak sirkuit Euler yang dapat diperoleh dari suatu graf berarah yang terdapat pada teorema BEST. Untuk mendapatkan banyaknya sirkuit Euler pada graf G , teorema BEST bergantung pada banyaknya pohon rentangan terhadap simpul v_i pada graf berarah G serta derajat simpul-simpulnya.

Kemudian, banyaknya pohon rentangan pada graf berarah G dihitung menggunakan kofaktor matriks Laplacian yang terdapat pada Teorema Matriks Pohon.



BAB III

TEOREMA BEST

Bab ini khusus membahas pembuktian Teorema BEST yang digunakan untuk menghitung banyaknya sirkuit Euler pada graf berarah G . Semua teorema pada bab ini mengacu pada [KJ08]. Sebelum membahas Teorema BEST akan dijelaskan terlebih dahulu mengenai Teorema Matriks Pohon (*Matrix Tree Theorem*).

3.1 Teorema Matriks Pohon

Pada bagian ini diberikan teorema matriks pohon yang akan digunakan untuk menghitung banyaknya pohon rentangan berarah terhadap simpul tertentu.

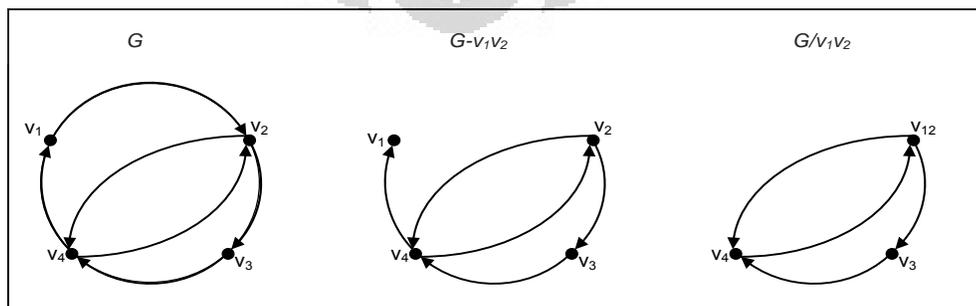
Teorema 3.1 Teorema Matriks Pohon

Diberikan graf berarah terhubung G dengan himpunan simpul $V(G) = \{v_1, \dots, v_n\}$ dan $t_i(G)$ merupakan himpunan pohon rentangan berarah terhadap simpul v_i maka $|t_i(G)|$ sama dengan kofaktor baris ke- i dan kolom ke- i dari $L(G)$. Dengan kata lain

$$|t_i(G)| = \det_{i,i} L(G)$$

Bukti. Digunakan induksi terhadap jumlah busur m dalam graf. Diasumsikan graf G terhubung dan m lebih besar dari satu, sebab graf tidak terhubung tidak mempunyai pohon rentangan, dan graf dengan satu busur hanya mempunyai satu pohon rentangan.

Misalkan diberikan dua simpul v_1 dan v_2 yang saling bertetangga di graf G . Akan dibentuk dua graf baru dari graf G yang memiliki busur kurang dari m . Graf pertama dibentuk dengan menghapus semua busur berarah yang meninggalkan simpul v_1 dan menuju simpul v_2 pada graf G . Himpunan busur ini dinyatakan dengan v_1v_2 dan graf yang dihasilkan disebut $G-v_1v_2$. Graf yang kedua dibentuk dengan mengkontraksi busur v_1v_2 , sehingga menghasilkan graf G/v_1v_2 . Dalam hal ini, simpul v_1 dan v_2 akan bergabung menjadi simpul baru yang disebut v_{12} . Busur yang sebelumnya menuju simpul v_1 dan v_2 akan digabung menjadi satu busur dengan bobot totalnya adalah penjumlahan dari tiap-tiap busur. Selanjutnya bobot semua busur yang meninggalkan setiap simpul yang telah dihilangkan ke dalam simpul v_i dan bergabung dengan cara yang sama. Kedua graf $G-v_1v_2$ dan graf G/v_1v_2 akan memiliki busur kurang dari m .



Gambar 3.1 (a) Graf berarah G , (b) Graf $G-v_1v_2$, dan (c) Graf G/v_1v_2

Dapat dilihat pada Gambar 3.1(b) dan 3.1(c) dimana kedua graf merupakan contoh dua graf baru yang dibentuk dari graf berarah pada Gambar 3.1(a). Terlihat bahwa graf $G - v_1v_2$ diperoleh dengan hanya menghapus satu busur berarah dari simpul v_1 ke simpul v_2 , dan graf G/v_1v_2 diperoleh dengan menyatukan busur v_4v_1 dan v_4v_2 menjadi satu busur dari v_4 ke simpul baru v_{12} . Tidak ada busur ganda yang meninggalkan v_{12} , karena tidak ada simpul di graf G yang menerima busur dari v_1 dan v_2 sekaligus.

Pada saat graf baru dibentuk maka hubungan antara jumlah pohon rentangan adalah

$$|t_i(G)| = |t_i(G - v_1v_2)| + a_{12} |t_i(G / v_1v_2)|$$

dimana a_{12} menyatakan banyaknya busur v_1v_2 yang ada di G .

Lebih jelasnya, jumlah dari pohon rentangan berarah terhadap simpul v_i sama dengan pohon rentangan di graf $G - v_1v_2$ ditambahkan dengan hasil kali dari pohon rentangan di graf G/v_1v_2 dan banyaknya busur berarah v_1v_2 . Suku pertama pada sisi kanan memperhitungkan semua pohon rentangan berarah terhadap v_i dimana busur-busur v_1v_2 telah dihapus. Suku kedua mempertimbangkan semua pohon rentangan yang tidak mengandung v_1v_2 . Karena itu, busur- busur ini dapat dinyatakan dengan a_{12} . Dengan menghapus semua busur v_1v_2 akan menyisakan pohon rentangan berarah terhadap simpul v_i dalam graf dimana v_1v_2 telah dihapus. Graf ini adalah G/v_1v_2 .

Setelah dibentuk hubungan pohon rentangan di G , $G-v_1v_2$, dan G/v_1v_2 , selanjutnya akan dibangun hubungan yang mirip hubungan dengan kofaktor pada matriks Laplacian di masing-masing graf sebagai berikut:

$$\det_{i,i} L(G) = \det_{i,i} L(G-v_1v_2) + a_{12} \det_{i,i} L(G/v_1v_2)$$

Untuk melengkapi bukti ini, maka harus dibuktikan bahwa rumus di atas akan selalu benar. Untuk melakukan hal itu dapat dibangun kesamaan antara unsur-unsur yang ada di sisi kanan persamaan dengan unsur-unsur dari sisi kanan pada persamaan sebelumnya dengan berdasarkan $|t_i(G)|$. Perlu diingat bahwa teorema dibuktikan dengan menggunakan induksi terhadap jumlah busur m dan persamaan di atas berdasarkan pada graf dengan jumlah busur kurang dari m .

Karena persamaan di atas menghubungkan antara kofaktor matriks Laplacian graf G dengan kofaktor matriks Laplacian graf $G-v_1v_2$ dan G/v_1v_2 , maka persamaan ini akan dibuktikan dengan terlebih dahulu membangun matriks Laplacian untuk setiap graf yang terlibat yaitu graf G , $G-v_1v_2$, dan G/v_1v_2 .

Seperti telah diberikan pada Bab II, matriks Laplacian adalah selisih dari matriks ketetanggaan dengan matriks diagonal sehingga matriks Laplacian dari graf berarah G adalah

$$L(G) = \begin{bmatrix} d_1 & -a_{12} & -a_{13} & -a_{14} & \cdots & -a_{1n} \\ -a_{21} & d_2 & -a_{23} & -a_{24} & \cdots & -a_{2n} \\ -a_{31} & -a_{32} & d_3 & -a_{34} & \cdots & -a_{3n} \\ -a_{41} & -a_{42} & -a_{43} & d_4 & & \vdots \\ \vdots & \vdots & \vdots & & \ddots & \vdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & \cdots & d_n \end{bmatrix}$$

Karena graf $G-v_1v_2$ diperoleh dengan menghapus semua busur berarah v_1v_2 dan diketahui bahwa a_{12} adalah banyaknya busur berarah dari v_1 ke v_2 , maka matriks Laplacian dari graf $G-v_1v_2$ adalah

$$L(G-v_1v_2) = \begin{bmatrix} d_1 & -a_{12} & -a_{13} & -a_{14} & \cdots & -a_{1n} \\ -a_{21} & d_2 - a_{12} & -a_{23} & -a_{24} & \cdots & -a_{2n} \\ -a_{31} & -a_{32} & d_3 & -a_{34} & \cdots & -a_{3n} \\ -a_{41} & -a_{42} & -a_{43} & d_4 & & \vdots \\ \vdots & \vdots & \vdots & & \ddots & \vdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & \cdots & d_n \end{bmatrix}$$

Perlu diperhatikan bahwa hanya dua komponen dari matriks ini yang berbeda dengan matriks $L(G)$, yaitu pada entri (1,2) dan (2,2) yang mewakili busur berarah dari vektor v_1 ke v_2 , dan derajat masuk dari v_2 . Derajat masuk dari v_2 akan berkurang karena busur-busur yang berasal dari v_1 , dinyatakan a_{12} dihapus. Selanjutnya, jelas nilai entri (1,2) adalah nol.

Matriks Laplacian untuk graf G/v_1v_2 sedikit berbeda. Karena jumlah simpul berkurang satu dari awal graf G , maka dimensi $L(G/v_1v_2)$ lebih kecil dari dimensi $L(G)$, yaitu menjadi $(n-1) \times (n-1)$, sehingga $L(G/v_1v_2)$ adalah

$$L(G/v_1v_2) = \begin{bmatrix} d_1 - a_{21} + d_2 - a_{12} & -(a_{13} + a_{23}) & -(a_{14} + a_{24}) & \cdots & -(a_{1n} + a_{2n}) \\ -(a_{31} + a_{32}) & d_3 & -a_{34} & \cdots & -a_{3n} \\ -(a_{41} + a_{42}) & -a_{43} & d_4 & & -a_{4n} \\ \vdots & \vdots & & \ddots & \vdots \\ -(a_{n1} + a_{n2}) & -a_{n3} & -a_{n4} & \cdots & d_n \end{bmatrix}$$

Membandingkan matriks ini dengan sebelumnya, dapat dilihat bahwa entri (1,1) adalah jumlah entri-entri (1,1), (1,2), (2,1), dan (2,2) di $L(G-v_1v_2)$. Hal ini dikarenakan simpul v_1 dan v_2 digabung serta busur berarah yang menghubungkannya dihapus. Untuk entri-entri yang lain pada baris pertama dan kolom pertama, setiap entri adalah penjumlahan dari entri-entri dari dua baris pertama dan dua kolom pertama dari matriks $L(G)$. Selanjutnya untuk entri sisanya sama dengan dua matriks sebelumnya.

Karena matriks Laplacian untuk ketiga graf berarah G , $G-v_1v_2$, dan G/v_1v_2 telah dibentuk, maka akan dilihat kofaktor dari matriks-matriks tersebut. Untuk suatu matriks M nyatakan $k_1(M)$ sebagai kofaktor yang diperoleh dengan menghapus baris pertama dan kolom pertama dari M , maka k_1 untuk ketiga graf sebagai berikut:

$$k_1(G) = \det \begin{bmatrix} d_2 & -a_{23} & -a_{24} & \cdots & -a_{2n} \\ -a_{32} & d_3 & -a_{34} & \cdots & -a_{3n} \\ -a_{42} & -a_{43} & d_4 & & -a_{4n} \\ \vdots & \vdots & & \ddots & \vdots \\ -a_{n2} & -a_{n3} & -a_{n4} & \cdots & d_n \end{bmatrix} \quad (1)$$

$$k_1(G - v_1v_2) = \det \begin{bmatrix} d_2 - a_{12} & -a_{23} & -a_{24} & \cdots & -a_{2n} \\ -a_{32} & d_3 & -a_{34} & \cdots & -a_{3n} \\ -a_{42} & -a_{43} & d_4 & & -a_{4n} \\ \vdots & \vdots & & \ddots & \vdots \\ -a_{n2} & -a_{n3} & -a_{n4} & \cdots & d_n \end{bmatrix} \quad (2)$$

dan

$$k_1(G / v_1v_2) = \det \begin{bmatrix} d_3 & -a_{34} & \cdots & -a_{3n} \\ -a_{43} & d_4 & & -a_{4n} \\ \vdots & & \ddots & \vdots \\ -a_{n3} & -a_{n4} & \cdots & d_n \end{bmatrix} \quad (3)$$

Misalkan $k_1(G/v_1v_2) = D_0$ dan D_i sebagai determinan dari matriks yang diperoleh dengan menghapus baris ke-1 dan kolom $i+1$ dari matriks Laplacian (L) dengan $i = 1, \dots, n-1$ maka didapat persamaan sebagai berikut :

$$k_1(G) = d_2 \cdot D_0 - [(-a_{23} \cdot D_1) + (-a_{24} \cdot D_2) + \dots + (-a_{2n} \cdot D_{n-1})]$$

dan

$$k_1(G - v_1v_2) = (d_2 - a_{12}) \cdot D_0 - [(-a_{23} \cdot D_1) + (-a_{24} \cdot D_2) + \dots + (-a_{2n} \cdot D_{n-1})]$$

Dari $k_1(G)$, $k_1(G - v_1v_2)$, dan $k_1(G/v_1v_2)$ diperoleh

$$k_1(G) = k_1(G - v_1v_2) + a_{12}k_1(G / v_1v_2)$$

Dengan menggunakan determinan matriks Laplacian, persamaan diatas dapat juga ditulis menjadi

$$\det_{i,i} L(G) = \det_{i,i} L(G - v_1v_2) + a_{12} \det_{i,i} L(G / v_1v_2)$$

Dengan demikian berdasarkan induksi pada jumlah m busur dalam graf G , dapat ditemukan persamaan yang menyatakan bahwa banyaknya

pohon rentangan terhadap simpul v_1 dan kofaktor untuk dua graf dengan jumlah kurang busur m adalah sama, yaitu

$$|t_i(G - v_1v_2)| + a_{12} |t_i(G / v_1v_2)| = \det_{i,i} L(G - v_1v_2) + a_{12} \det_{i,i} L(G / v_1v_2)$$

Sehingga dapat disimpulkan bahwa

$$|t_i(G)| = \det_{i,i} L(G)$$

Karena itu, banyaknya pohon rentangan terhadap simpul v_1 untuk sirkuit Euler pada graf berarah sama dengan kofaktor dari matriks Laplacian.

Untuk mengilustrasikan teorema matriks pohon ini akan diberikan contoh.

Contoh 3.1

Misalkan dengan menggunakan graf G pada Gambar 3.1, maka matriks ketetanggaan untuk graf G adalah

$$A(G) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Matriks diagonal untuk graf G adalah

$$D(G) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

Matriks Laplacian untuk graf G adalah

$$L(G) = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 2 & -1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & -1 & 0 & 2 \end{bmatrix}$$

Kemudian matriks Laplacian untuk graf $G - v_1 v_2$ adalah

$$L(G - v_1 v_2) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & -1 & 0 & 2 \end{bmatrix}$$

Dapat dilihat dua komponen dari matriks ini berbeda dengan matriks $L(D)$, yaitu pada entri (1,2) dan (2,2). Untuk entri (1,2) bernilai nol karena busur yang menghubungkan simpul v_1 ke simpul v_2 dihapus, kemudian untuk entri (2,2) bernilai satu karena derajat masuk untuk simpul v_2 berkurang satu busur yang berasal dari simpul v_1 yang dinyatakan a_{12} dihapus.

Matriks Laplacian untuk graf $G / v_1 v_2$ adalah

$$L(G / v_1 v_2) = \begin{bmatrix} 2 & -1 & -1 \\ 0 & 1 & -1 \\ -2 & 0 & 2 \end{bmatrix}$$

Disini dapat dilihat bahwa matriks Laplacian untuk graf $G / v_1 v_2$ sedikit berbeda. Karena jumlah simpul berkurang satu dari awal graf G , maka dimensi $L(G / v_1 v_2)$ lebih kecil dari dimensi $L(G)$, yaitu menjadi $(n-1) \times (n-1) = 3 \times 3$, dimana nilai entri (1,1) adalah jumlah entri-entri (1,1), (1,2), (2,1), dan (2,2) di $L(G - v_1 v_2)$. Dikarenakan simpul v_1 dan v_2 digabung serta busur

berarah yang menghubungkannya dihapus. Untuk entri-entri yang lain pada baris pertama dan kolom pertama, setiap entri adalah penjumlahan dari entri-entri dari dua baris pertama dan dua kolom pertama dari matriks $L(G)$.

Selanjutnya untuk entri sisanya sama dengan dua matriks sebelumnya.

Kemudian akan dicari kofaktor k_1 dari matriks masing-masing graf G , $G-v_1v_2$, dan G/v_1v_2 , yang diperoleh dengan menghapus baris pertama dan kolom pertama dari setiap matriks, kemudian menghitung determinan matriksnya. Maka akan didapat sebagai berikut:

$$k_1(G) = \det \begin{bmatrix} 2 & -1 & -1 \\ 0 & 1 & -1 \\ -1 & 0 & 2 \end{bmatrix}$$

$$k_1(G-v_1v_2) = \det \begin{bmatrix} 1 & -1 & -1 \\ 0 & 1 & -1 \\ -1 & 0 & 2 \end{bmatrix}$$

$$\text{dan } k_1(G/v_1v_2) = \det \begin{bmatrix} 1 & -1 \\ 0 & 2 \end{bmatrix}$$

Nilai dari ketiga kofaktor matriks ini adalah

$$k_1(G) = 2(2-0) - [-1(-1)] + [-1(1)] = 4 - 2 = 2$$

$$k_1(G-v_1v_2) = 1(2-0) - [-1(-1)] + [-1(1)] = 0$$

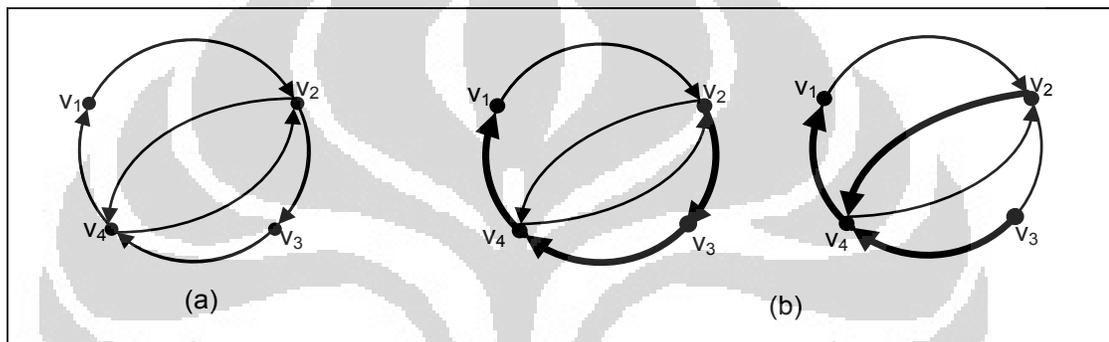
dan

$$k_1(G/v_1v_2) = 2.$$

Sehingga dapat ditunjukkan bahwa

$$k_1(G) = k_1(G-v_1v_2) + a_{12}k_1(G/v_1v_2) = 0 + 1(2) = 2.$$

Pada teorema matriks pohon telah dibuktikan bahwa banyaknya pohon rentangan berarah terhadap suatu simpul v_i pada graf G sama dengan kofaktor dari matriks Laplacian. Oleh karena itu, dapat disimpulkan bahwa dengan menggunakan Gambar 3.1(a) terdapat dua buah pohon rentangan berarah terhadap simpul v_i di G .



Gambar 3.2 (a) Graf berarah G , (b) Pohon rentangan berarah terhadap simpul v_1 yang ditunjukkan dalam graf di sebelah kanan yang bergaris hitam tebal.

3.2 Teorema BEST

Teorema BEST adalah teorema yang menyatakan banyaknya sirkuit Euler dari sembarang graf berarah, dimana kata **BEST** merupakan singkatan dari nama-nama de **B**ruijn, Van **A**ardenne **E**hrenfest, **S**mith, dan **T**utte.

Teorema 3.2 Teorema BEST [KJ08]

Diberikan graf berarah terhubung G dan himpunan simpul tak kosong

$V(G) = \{v_1, \dots, v_n\}$ dengan $d^+(v_i) = d^-(v_i)$ untuk setiap i , untuk $i = 1, \dots, n$.

Banyaknya sirkuit Euler $s|(G)|$ dinyatakan sebagai berikut:

$$|s(G)| = |t_i(G)| \prod_{j=1}^n (d^+(v_j) - 1)!$$

dimana $|t_i(G)|$ adalah banyaknya pohon rentangan terhadap setiap simpul v_i di G .

Bukti. Diberikan multigraf berarah G dengan himpunan simpul tak kosong $V(D) = \{v_1, \dots, v_n\}$ dimana derajat keluar dan derajat masuk dari sebuah simpul itu sama, yang dinotasikan dengan $d^+(v_i) = d^-(v_i)$. Jika kondisi ini terpenuhi, maka akan ada setidaknya satu sirkuit Euler berdasarkan Teorema 2.1.

Selanjutnya, E adalah himpunan dari sirkuit Euler yang berarah dan E_i merupakan himpunan jalur Euler yang mulai dan berakhir pada simpul v_i . Jadi jalur E_i harus mengunjungi setiap busur pada graf tepat satu kali. Karena banyak kali sirkuit Euler akan melalui simpul v_i setara dengan derajat keluar (atau derajat masuk) dari v_i , maka banyaknya jalur Euler yang mulai dan berakhir pada simpul v_i dapat dinyatakan sebagai berikut:

$$|E_i| = d^+(v_i) |E| = d^-(v_i) |E| \quad (1)$$

Persamaan (1) menyatakan bahwa banyaknya jalur Euler yang mulai dan berakhir pada simpul v_i sama dengan derajat keluar atau derajat masuk dari simpul v_i dikalikan dengan banyaknya sirkuit Euler di G .

Sekarang, misalkan T_i merupakan pohon rentangan berarah terhadap simpul v_i dan diketahui pemetaan fungsi $f_i : E_i \rightarrow T_i$. Pemetaan ini memiliki

input yaitu himpunan jalur Euler yang dimulai dan diakhiri pada simpul v_i di G dan memiliki *output* yaitu himpunan pohon rentangan berarah terhadap simpul v_i di G . Untuk penyederhanaan bukti digunakan $i=1$ sehingga

$$f_1 : E_1 \rightarrow T_1.$$

Misalkan S adalah jalur Euler dari himpunan E_1 dimana pemetaan ini membentuk suatu himpunan busur e_j dengan langkah-langkah berikut. Langkah pertama yaitu busur yang keluar dari simpul v_1 di S tidak digunakan, berarti yang akan digunakan $j = 2, \dots, n$ untuk setiap busur e_j . Selanjutnya, setiap busur e_j di S akan melewati simpul v_j untuk terakhir kali dan tidak akan kembali lagi ke dalam jalur Euler. Maka hasil dari pemetaan ini merupakan pohon T_1 berarah dengan simpul v_1, \dots, v_n dan busur e_2, \dots, e_n yang ada di himpunan pohon rentangan berarah terhadap simpul v_1 . Gambar 3.2 menunjukkan sebuah contoh pemetaan yang ditentukan berdasarkan sirkuit Euler di graf G .

Selanjutnya klaim bahwa $T \in T_1$, maka harus dibuktikan bahwa (a) T merupakan pohon dan (b) T adalah graf berarah terhadap simpul v_1 .

Untuk menunjukkan (a), pertama-tama anggap bahwa T memuat suatu lingkaran C , yang tidak akan memuat v_1 . Karena C adalah sebuah lingkaran, maka setiap simpul C memiliki derajat keluar sama dengan satu. Asumsikan busur e_g adalah busur terakhir pada jalur Euler S di lingkaran C , yang menghubungkan simpul v_g dan v_h . Jika hal ini terjadi, maka S kembali

ke simpul v_h walaupun S itu sudah melewati simpul v_h untuk terakhir kalinya setelah dilewati busur e_h sebelumnya. Hal ini bertentangan dengan pemetaan yang berkaitan, dimana dinyatakan bahwa busur e_j mengikuti kunjungan terakhir ke simpul v_j , dan tidak akan kembali lagi ke simpul v_j . Oleh karena itu, T adalah pohon.

Kemudian, akan dibuktikan (b) dengan mempertanyakan apakah T adalah graf berarah terhadap simpul v_1 . Langkah pertama yang dilakukan adalah mengasumsikan bahwa T memuat lintasan $v_k v_{k-1} \dots v_1$. Busur yang hadir antara simpul v_1 dan v_2 adalah e_2 , karena tidak ada busur e_1 . Selanjutnya, busur yang hadir antara simpul v_3 dan v_2 adalah e_2 atau e_3 , tetapi karena e_2 telah ditetapkan menjadi busur yang hadir antara simpul v_1 dan v_2 maka busur tersebut haruslah busur e_3 . Selanjutnya dengan cara yang sama dilakukan di antara semua simpul di lintasan, dibuktikan bahwa lintasan $v_k v_{k-1} \dots v_1$ merupakan graf berarah terhadap simpul v_1 .

Dengan demikian, graf berarah T ada di himpunan pohon rentangan berarah terhadap simpul v_i dari graf asli G . Karena itu, dapat disimpulkan bahwa $T \in T_1$.

Selanjutnya, untuk mendapatkan pemetaan yang diinginkan $f_1 : E_1 \rightarrow T_1$, maka harus ditetapkan bahwa $f_1(s) = T$. Kemudian, dengan mulai bekerja dari $f_1^{-1}(T)$, yang artinya diketahui pohon rentangan T pada simpul v_1 akan menimbulkan jalur Euler.

Sehingga, dengan menggunakan $f_1(s) = T$ dan $S \in E_1$, dapat dibuat sebuah jalur Euler S dengan cara sebagai berikut: Dimulai dari sebuah busur yang meninggalkan simpul v_1 . Setelah kembali ke simpul v_1 , maka selanjutnya jalur akan dilanjutkan melalui busur lain yang belum digunakan dan harus keluar pada simpul v_1 . Apabila tidak ada busur lain yang bisa keluar dari simpul v_1 maka terbentuk jalur Euler. Sesampainya di simpul v_j ($j > 1$), busur yang meninggalkan simpul v_j bukan busur e_j ataupun busur yang terdapat pada pohon. Setelah busur ini habis, maka selanjutnya biarkan busur e_j melalui simpul v_j .

Berdasarkan fakta bahwa derajat masuk dan derajat keluar dari setiap simpul v_j di G adalah sama, maka telah berhasil ditemukan sebuah jalur Euler S di himpunan E_1 dengan $f_1(s) = T$. Perlu diketahui bahwa untuk beberapa pohon rentangan akan ada lebih dari satu jalur Euler S yang dapat dihasilkan.

Menggunakan graf G (Gambar 3.2) sebagai contoh, akan ditunjukkan proses untuk pembentukan jalur Euler dengan menggunakan pohon rentangan (ditunjukkan dengan garis tebal dalam Gambar 3.2). Dimulai dengan busur yang meninggalkan v_1 menuju v_2 . Setelah di v_2 , harus menggunakan busur menuju simpul v_4 , karena busur yang menuju v_3 , busur e_j , telah digunakan dalam pohon T . Sebagai catatan, di v_4 ada dua busur yang keluar, karena salah satu busur menuju v_1 juga bagian dari pohon, maka harus menggunakan busur yang kembali ke simpul v_2 . Setelah kembali

ke v_2 , busur yang tersisa hanya satu busur dari pohon, sehingga busur ini yang digunakan. Hal yang sama berlaku untuk v_4 , karena satu-satunya busur yang tersisa juga dari T , melengkapi jalur Euler S .

Prinsip utama dalam proses ini adalah sebagai berikut: setelah memasuki sembarang simpul pada graf, maka pada saat keluar dari simpul tersebut menggunakan busur yang tidak berasal dari pohon sebisa mungkin, sampai hanya tersisa sebuah busur pada pohon tersebut. Dengan melakukannya secara berulang-ulang untuk setiap simpul maka akan dihasilkan sebuah jalur Euler pada graf.

Secara umum, banyaknya jalur Euler S dapat dihasilkan dari pohon rentangan T berarah terhadap simpul v_1 (dinotasikan dengan $|f_1^{-1}(T)|$) sebagai berikut:

$$|f_1^{-1}(T)| = d^+(v_1)! \prod_{j=2}^n (d^+(v_j) - 1)! \quad (2)$$

Faktorial dari derajat keluar pada simpul v_1 mempertimbangkan beberapa cara berbeda yang dapat dilalui jalur Euler (yang menjelaskan mengapa ukuran dari E_1 lebih besar daripada E), dan produk faktorial mempertimbangkan bahwa terdapat satu busur yang berkurang, dimana busur tersebut keluar dari simpul v_j pada saat jalur Euler S melaluinya.

Selain itu, setelah perhatian difokuskan pada himpunan sirkuit Euler E dan mengabaikan himpunan simpul v_i dimana sirkuit Euler dimulai, maka

banyaknya sirkuit Euler adalah sebagai berikut:

$$|E| = |T_1| \prod_{j=1}^n (d^+(v_j) - 1)! \quad (3)$$

Disini produk faktorial untuk derajat keluar semua simpul di G dikalikan dengan jumlah pohon rentangan T_1 berarah terhadap simpul v_1 .

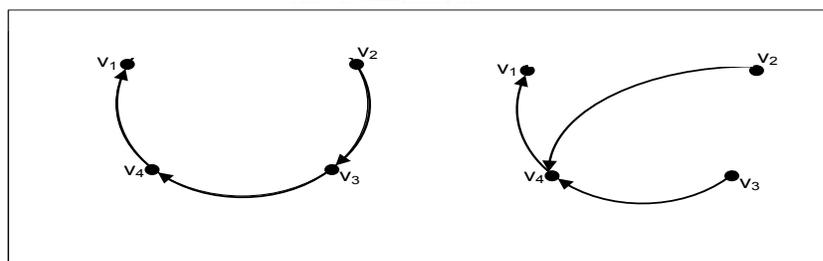
Dengan rangkaian persamaan (2) dan (3) telah dibuktikan dan diperiksa ketepatan rumus yang terkait dengan Teorema BEST.

Teorema BEST bergantung pada banyaknya pohon rentangan terhadap simpul v_i serta derajat simpul-simpulnya, untuk mendapatkan banyaknya sirkuit Euler pada graf G . Kemudian, untuk menentukan banyaknya pohon rentangan terhadap simpul v_i pada graf berarah G bisa menggunakan kofaktor matriks Laplacian yang terdapat pada Teorema Matriks Pohon yang telah dibahas terlebih dahulu.

Untuk mengilustrasikan teorema BEST ini akan diberikan contoh.

Contoh 3.2

Pada contoh 3.1 didapat banyaknya pohon rentangan berarah terhadap simpul v_1 di G adalah dua, yang dapat ditunjukkan pada Gambar 3.3.



Gambar 3.3 Pohon rentangan berarah terhadap v_1

Diketahui bahwa teorema BEST bergantung pada banyaknya pohon rentangan terhadap simpul v_i serta derajat simpul-simpulnya. Untuk mendapatkan banyaknya sirkuit Euler pada graf G , menggunakan teorema BEST

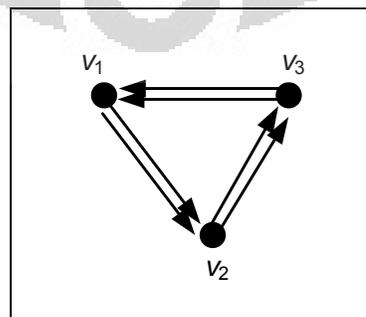
$$\begin{aligned} |s(G)| &= |t_i(G)| \prod_{j=1}^n (d^+(v_j) - 1)! \\ &= |2| (1-1)!(2-1)!(1-1)!(2-1)! \\ &= 2 \end{aligned}$$

Maka didapat bahwa banyaknya sirkuit Euler pada graf G yang terdapat pada Gambar 3.1 adalah dua, yaitu $v_1 v_2 v_3 v_4 v_2 v_4 v_1$ dan $v_1 v_2 v_4 v_2 v_3 v_4 v_1$.

Untuk memperjelas pembahasan pada Bab ini, akan diberikan contoh 3.3.

Contoh 3.3

Diberikan graf berarah G seperti pada Gambar 3.4, maka akan dicari banyaknya sirkuit Euler pada graf tersebut dengan menggunakan Teorema BEST.



Gambar 3.4 Graf berarah G

Pertama-tama akan dicari banyaknya pohon rentangan terhadap simpul tertentu dengan menggunakan teorema matriks pohon. Untuk itu akan dibuat matriks ketetanggaan, matriks diagonal, dan matriks Laplacian dari graf G yaitu:

$$A(G) = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 2 \\ 2 & 0 & 0 \end{bmatrix}$$

$$D(G) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

$$L(G) = \begin{bmatrix} 2 & -2 & 0 \\ 0 & 2 & -2 \\ -2 & 0 & 2 \end{bmatrix}$$

Kemudian cari kofaktor k_1 dari graf G adalah

$$k_1(G) = \det \begin{bmatrix} 2 & -2 \\ 0 & 2 \end{bmatrix}$$

Nilai kofaktor dari matriks $L(G)$ adalah

$$k_1(G) = 2(2) - 0(-2) = 4$$

Artinya banyak pohon rentangan berarah terhadap suatu simpul v_i pada graf G adalah empat (Teorema 3.1), dan menggunakan teorema BEST (Teorema 3.2) didapatkan banyaknya sirkuit Euler pada graf G adalah

$$\begin{aligned} |s(G)| &= |t_i(G)| \prod_{j=1}^n (d^+(v_j) - 1)! \\ &= |4| (2-1)!(2-1)!(2-1)! \\ &= 4 \end{aligned}$$

Sehingga banyaknya sirkuit Euler pada graf G (Gambar 3.4) adalah empat.

BAB IV

KESIMPULAN

Teorema BEST digunakan untuk mengetahui berapa banyak sirkuit Euler pada sembarang graf berarah. Dalam hal ini juga diketahui bahwa banyaknya sirkuit Euler tidak unik dan perlu diingat bahwa untuk memperoleh sirkuit Euler pada graf berarah harus memenuhi kondisi bahwa derajat masuk dan derajat keluar dari setiap simpul pada graf berarah adalah sama. Dalam pembuktian teorema BEST diperlukan informasi banyaknya pohon rentangan terhadap suatu simpul tertentu. Banyaknya pohon rentangan yang ada diperoleh dari teorema matriks pohon. Banyaknya sirkuit Euler pada suatu graf berarah adalah banyaknya pohon rentangan terhadap suatu simpul tertentu pada graf dikali dengan perkalian dari derajat-derajat masuk dikurang satu dari setiap simpul pada graf berarah tersebut.

DAFTAR PUSTAKA

- [Bol98] Bollobas, B. 1998. *Modern Graph Theory*. Springer. Memphis.
- [KJ08] Kaptcianos, Jonathan. 2008. A Graph theoretical Approach to DNA fragment Assembly, *American Journal of Undergraduate Research* vol. 7 number 1:1-12.
- [Pev01] Pevzner, Pavel .A, *et al.* 2001. An Eulerian Path Approach to DNA Fragment Assembly. *PNAS* vol. 98 number 17: 9748-9753.
- [Wil96] Wilson, Robin .J. 1996. *Introduction to Graph Theory*. Longman.