

FGT Lecture 3

Sep. 26, Oct. 3.

NO. 3-1
DATE

Eulerian circuits

Note The first paper of Graph Theory deals with the existence of Eulerian circuits (eulerian circuits).

Walk: A sequence of vertices in G $\langle x_1, x_2, \dots, x_n \rangle$ such that $x_i x_{i+1}$ is an edge of G where $i=1, 2, \dots, n-1$.

Trail: A walk without repeating edges.

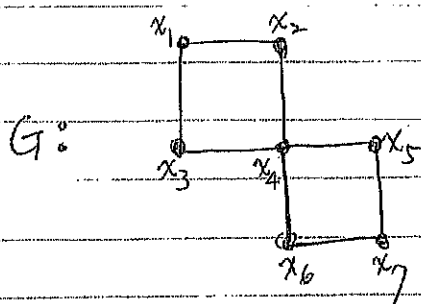
Circuit: A closed trail, i.e., the first vertex is also the last vertex. $\langle x_1, x_2, \dots, x_n \rangle$

Cycle: A circuit without repeating vertices. $(C_n) \Rightarrow \|C_n\| = n$.

Path: A trail (open) without repeating vertices. $(P_n) \Rightarrow \|P_n\| = n-1$.

Definition An eulerian circuit of a graph G (multi-graph)

is a circuit of G which contains all edges of G .



$\langle x_1, x_2, x_4, x_5, x_7, x_6, x_4, x_3, x_1 \rangle$

$= \langle x_1, x_2, x_4, x_5, x_7, x_6, x_4, x_3 \rangle$ is an eulerian circuit

of G .

(*) A graph is connected if for any two vertices x and y there exists a walk (path) from x to y . (可以用 path 取代?)
Sure!

Theorem (Euler, 1736 (1741))

A graph G contains an eulerian circuit if and only if G is connected and every vertex is of even degree.

(Remark: The graph considered by Euler is a multi-graph.)

Proof. (Many versions) $(\Rightarrow)$ is easy to see, we prove $(\Leftarrow)$.

1st proof: (Best) Let $T = \langle x_1, x_2, \dots, x_k \rangle$ be the longest trail in G starting from x_1 and ending at x_k . Since each vertex is an even vertex,

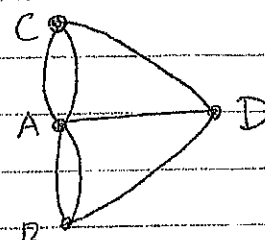
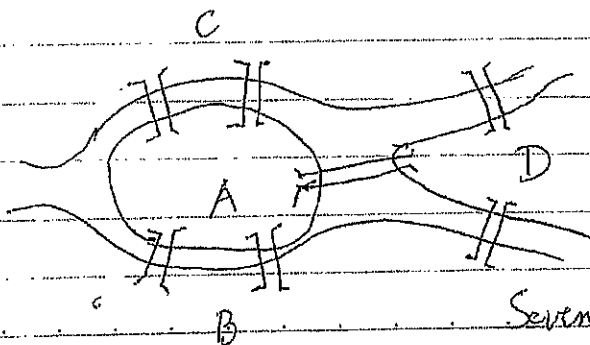
$x_1 = x_k$. Let E' be the set of edges of this trail. If $\|E'\| = \|G\|$,

then the trail is an eulerian circuit. Assume that $E'' = E(G) \setminus E' \neq \emptyset$.

Since G is connected, $V(E'') \cap V(T) \neq \emptyset$. Let x_i be a common vertex

and $x_i x_{i+1} \in E''$. Now, we have a longer trail starting from x_k ,

$\langle x_k, x_i, x_{i+1}, \dots, x_i, x_2, \dots, x_k \rangle$, a contradiction. Hence $E'' = \emptyset$ and T is an eulerian circuit. $\square$



Seven bridges on the Pregel in Königsberg

nd proof. (The main idea of Euler). By induction on $\|G\|$.

Since each vertex of G is even, $\delta(G) \geq 2$, G contains a cycle. Let C be a circuit in G with the maximal number of edges. (?)

If $\|C\| = \|G\|$, then C is an eulerian circuit of G . On the other hand,

$E(G) \setminus E(C) \neq \emptyset$, i.e., C is not an eulerian circuit of G . As G is connected,

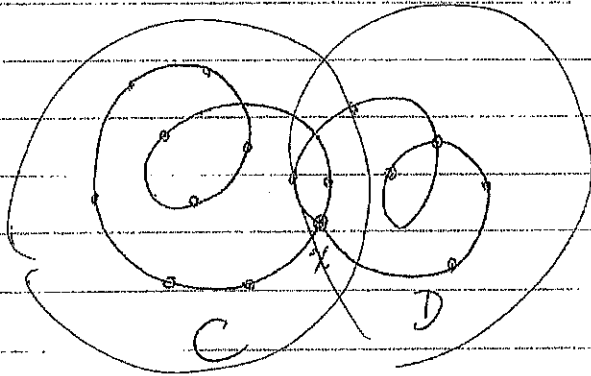
C contains a vertex x in a non-trivial component H of $G - E(C)$.

By induction H has an eulerian circuit D . Since $E(C) \cap E(D) = \emptyset$

and $V(C) \cap V(D) \supseteq \{x\}$, we obtain a larger circuit of G by

concatenating these two circuits. $\rightarrow$ Hence C is an eulerian

circuit of G . ■



滚雪球的概念
(棉花糖)。

Remark At the time of "Euler", mathematical induction is not known yet.

Different proofs ...?

Lemma

In a graph G , if $\delta(G)$ (minimum degree) ≥ 2 , then G contains a cycle of length at least $\delta(G)+1$.

Proof. Let $\langle x_1, x_2, \dots, x_t \rangle$ be a path of maximum distance from x_1 . Then, $N_G(x_1) \subseteq \{x_2, x_3, \dots, x_t\}$. Let x_t be the vertex with largest index. Clearly, $t \geq \delta(G)+1$. This concludes the proof.

(*) We also conclude that $l \geq \delta(G)+1$, hence G contains a path of length at least $\delta(G)$.

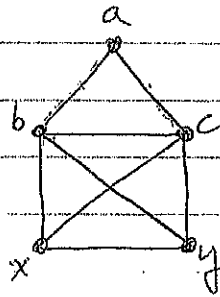
3rd proof of Eulerian circuit theorem

By induction of $\|G\|$. It is true if $\|G\|=2$. Assume that the assertion is true for G' with $\|G'\| < \|G\|$ and G' satisfies the sufficient conditions, G' is connected and every vertex is of even degree. Since $\delta(G) \geq 2$, G contains a cycle Z . Then,

$G-Z$ is either an empty graph or a graph of smaller size, such that each component of $G-Z$ satisfies the S.C. Now, by induction and the idea of Euler, we have the proof. (Put them together!) ▀

Sufficient Condition

Corollary A connected graph has an eulerian trail from a vertex x to a vertex $y \neq x$ if and only if x and y are the only odd vertices.



Eulerian Trail : $\langle x, b, a, c, x, y, b, c, y \rangle$

Corollary If a connected graph G has $2m$ odd vertices, then

($m \geq 1$)

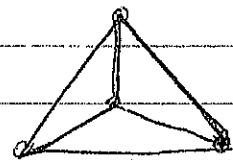
the edges of G can be partitioned into m trails such that each

1st part

trail has an even number of edges except possibly one.

2nd part

(No easy to prove! Especially the second part.)



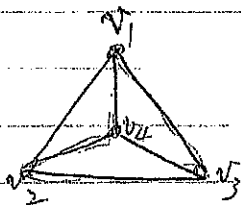
Digraph version

(To be continued!)

Proof of the first part.

Let the $2m$ odd vertices of G be $\{x_1, x_2, \dots, x_{2m-1}, x_{2m}\}$ and $V(G) = \{x_1, x_2, \dots, x_p\}$. Clearly $p \geq m$. Now, consider the graph $\tilde{G}$ with $V(\tilde{G}) = \{x_0, x_1, x_2, \dots, x_p\}$ and $E(\tilde{G}) = E(G) \cup \{x_0 x_i \mid i = 1, 2, \dots, 2m\}$. Then, $\tilde{G}$ is a connected even graph and thus $\tilde{G}$ has an eulerian circuit. Since the circuit passes x_0 exactly $2m$ times if we start at a vertex which is not x_0 , $\tilde{G} - x_0$ is decomposed into m trails which concludes the proof of the first part.

An example of 2nd part



$\langle v_1, v_4, v_2, v_3 \rangle$ and $\langle v_2, v_1, v_3, v_4 \rangle$
are two trails (Both have length 3)



$\langle v_1, v_4, v_2, v_1, v_3 \rangle$ and $\langle v_4, v_3, v_2 \rangle$

are two even trails

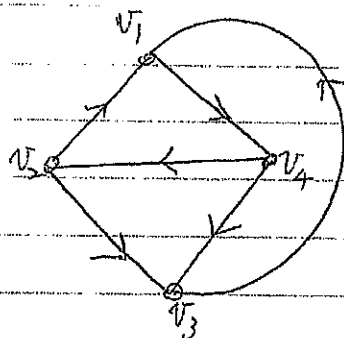
Digraph (Directed Graph) - Oct. 12

NO. 3-6
DATE

Definition $D =$ A digraph (V, A) is an ordered pair such that V is the set of vertices and A is a set of ordered pairs in (V, V) (arcs) or V^2 . $|A|$ is defined as the size of D (the number of arcs).

e.g.

$D:$



$$\begin{aligned} \deg_D^+(v_1) &= 1, \deg_D^-(v_1) = 2 \\ \deg_D^+(v_2) &= 2, \\ \deg_D^+(v_3) &= 1, \\ \deg_D^+(v_4) &= 2, \end{aligned}$$

$$V(D) = \{v_1, v_2, v_3, v_4\}$$

$$A(D) = \{(v_1, v_2), (v_1, v_3), (v_2, v_4), (v_3, v_4), (v_4, v_1), (v_2, v_4), (v_4, v_2)\}$$

$$N_D^+(v) = \{u \in V \mid (v, u) \in A\}, \quad N_D^-(v) = \{u \in V \mid (u, v) \in A\}$$

$\uparrow$ out-neighbor $\uparrow$ in-neighbor

$$\deg^+(v) = |N_D^+(v)|, \quad \deg^-(v) = |N_D^-(v)|$$

$\uparrow$ out-degree of v $\uparrow$ in-degree of v

Theorem Let $D = (V, A)$ be a digraph. Then

$$\sum_{v \in V(D)} \deg_D^+(v) = \sum_{v \in V(D)} \deg_D^-(v) = |A|.$$

Remark Walks, trails, circuits, cycles and paths in digraph D are defined according to the directions, i.e., in a walk $\langle v_1, v_2, v_3, \dots \rangle$

$(v_1, v_2), (v_2, v_3)$ are arcs in D . For clearness, we shall say a directed path in D instead of just a path in D .

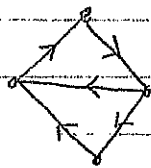
Definition (Connected digraphs)

A digraph D is connected if for any two distinct vertices x and y in $V(D)$, there exists a directed path from x to y (or) from y to x . If we replace "or" by "and", then the digraph is strongly connected.

weakly connected



weakly connected



strongly connected

Directed eulerian circuits

Theorem A digraph D has an eulerian (directed) circuit if and only if D is strongly connected and for each vertex $v \in V(D)$, $\deg^+(v) = \deg^-(v)$.

Proof. Similarly we can apply the idea used in graphs. ■

In fact, we can count the number of distinct directed eulerian circuits in an eulerian digraph. (A digraph with eulerian circuits is called an eulerian digraph.)

Theorem (BEST Theorem) ($V(D) = \{v_1, v_2, \dots, v_n\}$).

Let D be an eulerian digraph, $s(D)$ be the number of eulerian circuits, and $t_i(D)$ be the number of spanning trees oriented from v_i . Then $s(D) = t_i(D) \cdot \prod_{j=1}^n (\deg^+(v_j) - 1)!$ for every i , $1 \leq i \leq n$. (Note that $t_1(D) = t_2(D) = \dots = t_n(D)$.)

(The theorem was proved by de Bruijn, van Aardenne-Ehrenfest, Smith and Tutte.) We skip the details.

(*) Counting the number of distinct eulerian circuits in an eulerian graph (not directed) is a very difficult problem.

萬之徵答 problem 3*

There are many applications of "eulerian circuits", please refer to lecture notes "Chapter 2". For convenience, I attach them following this page.

Let G be an eulerian graph. Find the number of distinct eulerian circuits in G .

(Note) 知道不同歐拉迴路的個數有助於在應用於 DNA 排序問題時的實驗。

千之徵答 problem 3*

Let G be a connected q -regular eulerian graph, Find the number of distinct eulerian circuits in G .

2. De Bruijn 數列及郵差問題

這一節，我們將討論兩個尤拉圖概念的應用。

定義 2.1. (de Bruijn Sequence)

一個循環數列 $(a_1, a_2, \dots, a_{2^n})$ 稱為是 $(2, n)$ - de Bruijn 數列如果下列兩個條件滿足：

- (1) $a_i \in \{0, 1\}$, $i = 1, 2, \dots, 2^n$; 且
- (2) $(a_j, a_{j+1}, \dots, a_{j+n-1})$, $j = 1, 2, \dots, 2^n$, $(\text{mod } 2^n)$, 為相異的 2^n 個 n 維向量。

例1. $n = 1, 2, 3, 4$; $(2, n)$ -de Bruijn 數列分別為 (括號, 逗號省略)
 $01, 0110, 01110100, 0000100110101111$ 。

對於所有的 n 要建構一個 $(2, n)$ - de Bruijn 數列並不困難, 以下是荷蘭數學家 N. de Bruijn 用來尋找這種數列的有向圖。

定義 2.2. $((2, n)$ -de Bruijn 有向圖, $D_{2,n}$)

$D_{2,n}$ 為一加權有向圖, 它滿足下列兩條件:

- (1) $V(D_{2,n}) = (Z_2)^{n-1}$ 及
- (2) $(a_1, a_2, \dots, a_{n-1})$ 連到 $(a_2, a_3, \dots, a_{n-1}, a_n)$ 並且在這個弧上給予加權 $(a_1, a_2, \dots, a_n)$ 。

例2.

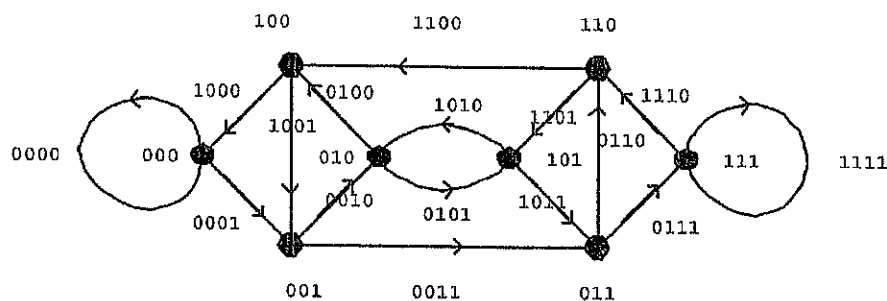


圖 5 : $D_{2,4}$

由定義, 我們可以證明 $D_{2,n}$ 為一有向的尤拉圖。

引理 2.3. $(2, n)$ -de Bruijn 有向圖為尤拉圖。

證明.

因為 $D_{2,n}$ 為強連通圖而且每一點的內度與外度均為 2, 故得證。 ■

引理 2.4. 在 $(2, n)$ -de Bruijn 有向圖上的加權全部不一樣。

證明.

由定義可得。 ■

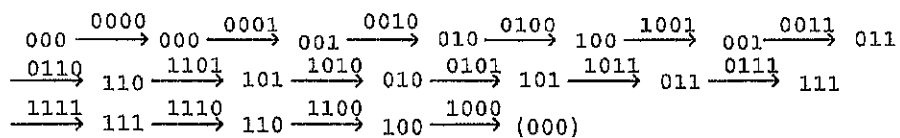
定理 2.5. 對於所有的 n , $(2, n)$ -de Bruijn 數列存在。

證明.

首先, 由引理 2.3, 一個 $(2, n)$ -de Bruijn 有向圖存在, 所以存在一個尤拉迴路; 現在, 令此迴路經過的邊為 $e_1, e_2, \dots, e_{2^n}$; 同時對於所有的 i 令 $l(e_i) = a_i$ 為 e_i 邊上加權的 最左邊那個數字, 於是我們得一個數列 $(a_1, a_2, \dots, a_{2^n})$, 這個數列就是一個 $(2, n)$ -de Bruijn 數列。(作業6.) ■

以下的例子也許有助於了解上述的證明。

在 $D_{2,4}$ 中的尤拉迴路可以是



$(2,4)$ -de Bruijn 數列為 0000100110101111, 而它所產生的 16 個不同之向量依次為 16 個加權向量。

De Bruijn 數列最出名的應用是在旋轉鼓 (Rotating Drum)。它可以利用連續的位置來判斷不同的輸入 (機械原理), 如下圖所示。

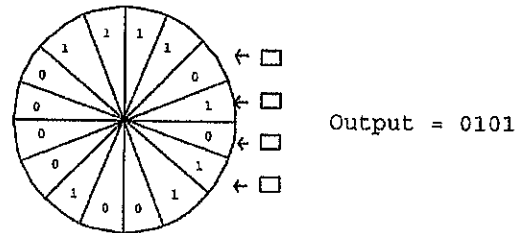


圖 6：旋轉鼓

另一個出名的應用是由華裔數學家管梅谷 (Meigu Guan) 所提出來, 後來 J. Edmonds 稱它為中國郵差問題。

定義 2.6. (郵差路線, Postman Tour)

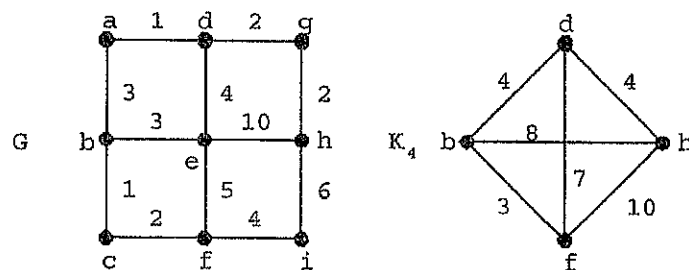
任給一個圖 G 所謂郵差路線是指在 G 中的一個封閉步行 (Closed Walk), 它通過 G 中的每一個邊至少一次。

定義 2.7. (最佳的郵差路線, Optimal Postman Tour)

在一個加權圖中, 一個郵差路線所經過的邊加權總和為最小時, 我們稱這路線為最佳郵差路線。

這個問題看來十分複雜。因為所考慮的圖千萬化, 然而, 在提出這個問題 (尋找最佳郵差路線) 之後不到十年, 即由 Edmonds 和 Johnson 所解決, 他們提供了一個多項式時間 (Polynomial-time) 的演算法來找出最佳路線。

在沒有介紹演算法之前, 我們先看一個例子。在 G 中有 4 個奇點, 現在利用這 4 點建構一個加權的 K_4 , 在邊上的加權為兩點在 G 中的最短距離 (加權)。接著在 K_4 中找到不相鄰的兩邊 (配對) 使得



它們的加權和最小, 此例中的 bf 即 dh , 最後再將原圖 G 改為 G^* , 如圖 7, G^* 的尤拉迴路即為所求。

演算法 (Edmonds 及 Johnson) (摘要)

- 證明.**

其它方面有不少應用, 例如在掃街路線的安排, 要怎樣行駛才最省經費; 在 RNA 的重組方面如何利用片斷資訊來得到完整的 RNA; 以及在資訊傳遞方面如何編碼才更有效率地把資料表現出來, 都可以利用有向的尤拉圖來達到目的; 由於篇幅有限在此省略, 請自行參考由 Gross 及 Yellen 所寫的書 "Graph Theory and its Application"。

DNA 排序的参考资料

Shortest Superstring Problem

Example set of strings: $\{001, 100, 101\}$

001100101 (Concatenation)

0010100 (Better)

Problem Given a set of strings $\vec{s}_1, \vec{s}_2, \dots, \vec{s}_n$, find the shortest string $\vec{s}$ such that $\forall i=1, 2, \dots, n, \vec{s}_i$ is a substring of $\vec{s}$.

Example set of strings $\{\text{CATGC}, \text{CTAAGT}, \text{GCTA}, \text{TTCATGCATC}\}$.

$\vec{s} : \text{GCTAAGTTCTATGCATC}$

(*) Greedy Algorithm (把可以接在一起的依次接好, 重叠部分处理好)

Step 1. ① + ⑤ CATGCATC

Step 2. ② + ③ GCTAAGT

Step 3. ④ + ⑥ TTCATGCATC

Step 4 ⑦ + ⑧ GCTAAGTTCTATGCATC

(Answer)

Prefix graph G of sequences

(*) $\text{overlap}(\vec{s}_i, \vec{s}_j)$

$\vec{s}_i$: ACGGCTAT

$\vec{s}_j$: CTATAG

$$\text{overlap}(\vec{s}_i, \vec{s}_j) = \text{CTAT}$$

(*) $\text{prefix}(\vec{s}_i, \vec{s}_j)$: 去掉 $\text{overlap}(\vec{s}_i, \vec{s}_j)$ 之後, $\vec{s}_i$ 的前段.

上例: $\text{prefix}(\vec{s}_i, \vec{s}_j) = \text{ACGG}$ (有向)

Definition (Prefix graph)

The prefix graph G of a set of strings $\{\vec{s}_1, \vec{s}_2, \dots, \vec{s}_n\}$

is a double-weighted complete digraph such that

$$V(G) = \{\vec{s}_1, \vec{s}_2, \dots, \vec{s}_n\}, \quad A(G) = \{(\vec{s}_i, \vec{s}_j) \mid 1 \leq i, j \leq n\} \text{ and}$$

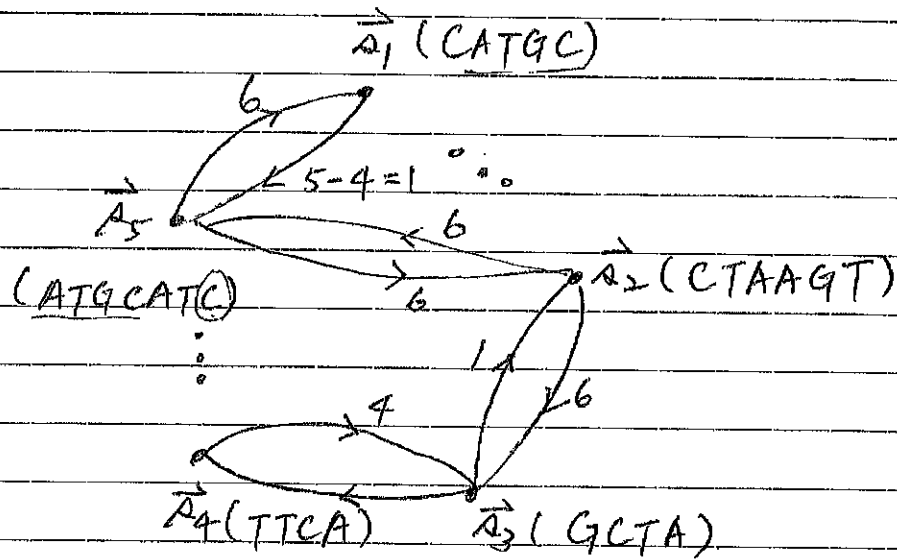
the weight of $(\vec{s}_i, \vec{s}_j) = |\vec{s}_i| - \text{overlap}(\vec{s}_i, \vec{s}_j)$, denoted by $w(\vec{s}_i, \vec{s}_j)$.

e.g. (上例)

$$w(\vec{s}_i, \vec{s}_j) = 4, \quad w(\vec{s}_j, \vec{s}_i) = 0. \quad (\text{双向边不重叠})$$

Example $\vec{s}_1 = \text{CATGC}$, $\vec{s}_2 = \text{CTAAGT}$, $\vec{s}_3 = \text{GCTA}$,
 $\vec{s}_4 = \text{TTCA}$ and $\vec{s}_5 = \text{ATGCATC}$.

G



(*) The goal is to find a directed Hamilton path with minimum total weight. (Very-hard problem)

An answer:

$$\vec{A}_3 \xrightarrow{1} \vec{A}_2 \xrightarrow{5} \vec{A}_4 \xrightarrow{2} \vec{A}_1 \xrightarrow{1} \vec{A}_5$$

$$\vec{A} = \text{GCTAAGTTTCATGCATC} \quad (\text{length } 16)$$

$$|\vec{A}_3| + |\vec{A}_2| + |\vec{A}_4| + |\vec{A}_1| + |\vec{A}_5| - |\text{overlap}(\vec{A}_3, \vec{A}_2)|$$

$$- |\text{overlap}(\vec{A}_2, \vec{A}_4)| - |\text{overlap}(\vec{A}_4, \vec{A}_1)| - |\text{overlap}(\vec{A}_1, \vec{A}_5)|$$

$$= 4 + 6 + 4 + 5 + 7 - 3 - 1 - 2 - 4 = 16.$$

Sequencing by Hybridization (SBH Problem)

(雜交定序)

Definition (l-mer composition)

For a string $\vec{s}$ of length n , the l -mer composition, or spectrum, of $\vec{s}$, is the multiset of $n-l+1$ l -mers (consecutive l letters, substring of length l) in $\vec{s}$, denoted by $\text{Spectrum}(\vec{s}, l)$.

For example

$\vec{s} = \text{ATGCTGCAAG}$, $|\vec{s}| = 10$

$l = 3$

$\Rightarrow \text{Spectrum}(\vec{s}, l) = \{ \text{ATG}, \text{TGC}, \text{CTG}, \text{TGC}, \text{GCA}, \text{CAA}, \text{AAG} \}$

GCT,

(*) SBH Problem

Two l -mers have overlap $l-1$, i.e., $|\text{overlap}(\vec{p}, \vec{q})| = l-1$
 $\vec{p}$ and $\vec{q}$ (連續) letters

Let S be a set of l -mers obtained from $\vec{s}$.

Reconstruct $\vec{s}$ from $S = \text{Spectrum}(\vec{s}, l)$.

Note: Clearly, we did not know the "order" of elements in S .

Idea: Two consecutive l -mers, $\vec{p}$ and $\vec{q}$ has $\text{overlap}(\vec{p}, \vec{q}) = l-1$.

(**) One $\vec{s}$ gives a unique $\text{Spectrum}(\vec{s}, l)$ for each $l=1, 2, \dots, n-1$.

(***) A given set of l -mers may produce two or more feasible strings, i.e., each of the strings has the same spectrum.

$(\Rightarrow)$ 唯一解答

$(\Leftarrow)$ 可能有多个答案

For example: $ATGCGTGGCA = \vec{s}_1$

$ATGGCGTGGCA = \vec{s}_2$

We may obtain the above two distinct strings from the set $\vec{s}?$

Spectrum $(\{?\}_3) = \{ ATG, TGG, TGC, GTG, GGC, GCA, GCG, CGT \}$
 $v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6 \quad v_7 \quad v_8$

$\vec{s}_1: v_1 - v_3 - v_7 - v_8 - v_4 - v_2 - v_5 - v_6 \quad ATGCGTGGCA$

$\vec{s}_2: v_1 - v_2 - v_5 - v_7 - v_8 - v_4 - v_3 - v_6 \quad ATGGCGTGGCA$

Observation

SBH problem is a particular (special) case of the Shortest Superstring problem, see it? However, in contrast to SSP, (1 保持最多的重叠部分)

there exists a simple linear-time algorithm for the SBH problem.

(*) Construct one $\vec{s}$ not all $\vec{s}$'s.
 (Greedy?)

Approach

① Hamiltonian Path problem (Hard problem), digraph

Let $V(G) = \{\vec{s}_1, \vec{s}_2, \dots, \vec{s}_n\}$ and $(\vec{s}_i, \vec{s}_j) \in A(G)$ iff

$\text{overlap}(\vec{s}_i, \vec{s}_j) = l - 1$. Then, a Hamiltonian path gives an $\vec{s}$.
 (directed)

② Eulerian Path Problem

From the set of l -mers we obtain a set T of $(l-1)$ -mers $\{\vec{k}_1, \vec{k}_2, \dots, \vec{k}_m\}$. Then, let $\tilde{G}$ be the graph obtained from letting $V(\tilde{G}) = T$ and $\vec{k}_i$ is incident to $\vec{k}_j$ if and only if $\text{overlap}(\vec{k}_i, \vec{k}_j) = l-2$ and $\vec{k}_i \wedge \vec{k}_j$ is an l -mer in S .

$\vec{k}_i \wedge \vec{k}_j$ is obtained by combining $\vec{k}_i$ and $\vec{k}_j$ together such that the overlapping portion (the suffix of $\vec{k}_i$ and prefix of $\vec{k}_j$) occurs exactly once. For example (See 5' for an example.)

$$\boxed{\text{ATG}} \wedge \boxed{\text{TGG}} = \text{ATGG}.$$

$$\begin{array}{c} \text{ATG} \\ \text{TGG} \end{array}$$

(*) Then, an eulerian path (directed) of $\tilde{G}$ gives an $\vec{s}$.

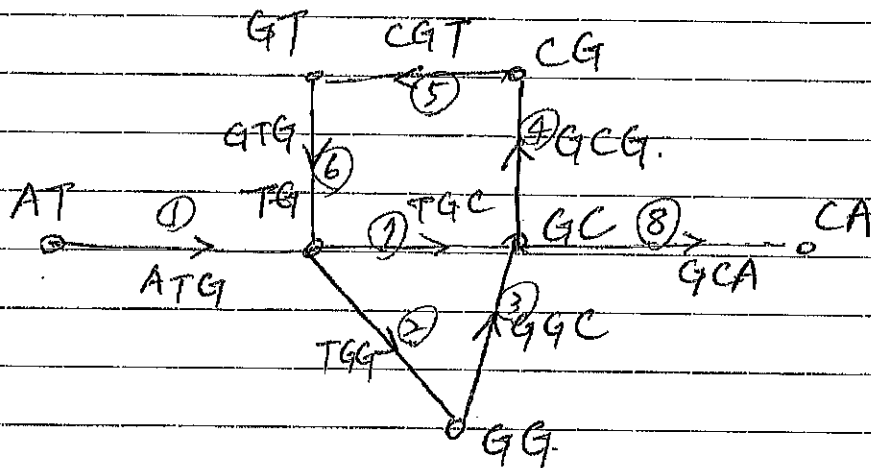
(Note) The number of ^{directed} eulerian circuits in a ^{connected} balanced

digraph can be enumerated. (BEST Theorem).
(directed eulerian graph)

Problem (Open) The number of eulerian circuits in an eulerian

graph can also be enumerated, but how?

(**) After the short 500-700 bp DNA reads are sequenced, biologists need to assemble them together to reconstruct the entire genomic DNA sequence. (Fragment Assembly Problem)



① → ② → ③ → ④ → ⑤ → ⑥ → ⑦ → ⑧

→ ATG GCGT GCA

① → ② → ④ → ⑤ → ⑥ → ② → ③ → ⑧

→ ATG CGTG GCA

(*) Eulerian path from AT to CA.