Graph Theory - Lecture 5
Euler Graphs, Hamiltonian Graphs
and
Interval Graphs

Chintan Kr Mandal
Department of Computer Science and Engineering
Jadavpur University, India

1 Euler Graphs

Eulerian trail An **Eulerian trail / Euler walk** in an undirected graph $\mathbf{G}(\mathbf{V},\mathbf{E})$ is a walk that uses each edge exactly once.

If such a walk exists, the graph is called **traversable / semi-eulerian**.

Eulerian cycle An **Eulerian cycle / Eulerian circuit / Euler tour** in an undirected graph is a cycle that uses each edge exactly once.

If such a cycle exists, the graph is called **Eulerian / unicursal**.

Eulerian orientation An **Eulerian orientation** of an undirected graph $\mathbf{G}(\mathbf{V},\mathbf{E})$ is an assignment of a direction to each edge of $\mathbf{G}(\mathbf{V},\mathbf{E})$ such that the $d^{+}(v) = d^{-}(v) \ \ \forall v \in V(G)$ where $d^{+}(v) = OutDegree(v)$ and $d^{-}(v) = InDegree(v)$.

Such an orientation exists for any undirected graph in which every vertex has even degree, and may be found by constructing an Euler tour in each connected component of $\mathbf{G}(\mathbf{V},\mathbf{E})$ and then orienting the edges according to the tour.

Every Eulerian orientation of a connected graph is a strong orientation, an orientation that makes the resulting directed graph strongly connected.

Definition 1.1 (Euler Graph)**.** *A connected graph is* **Eulerian** *if it has a* **closed trial containing all the edges**.

Note : The definition and properties of Eulerian trails, cycles and graphs are valid for multigraphs as well.
Check "Note – Eulerian circuits and directed graphs - Lincoln Lu" for applications

1.1 Konisberg Problem

Euler's Solution

Leonard Euler solved this problem by the simple observation:

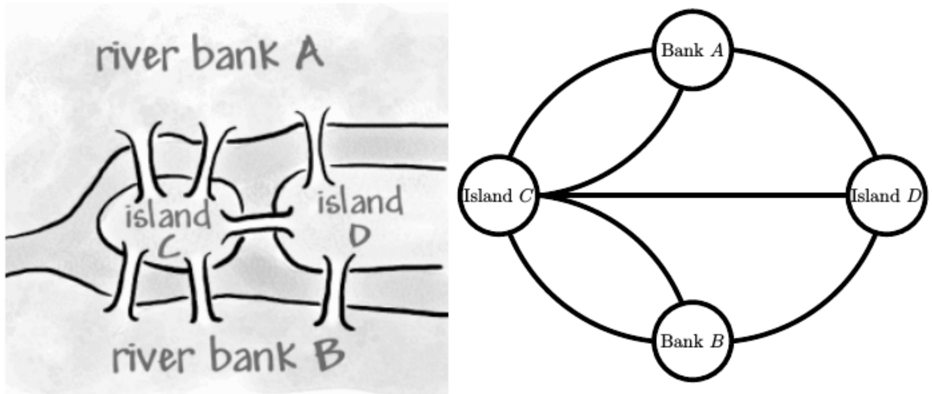


Figure 1: Kȯnigsberg bridge problem

This would only be possible if whenever you cross into a part of the city you must be able to leave it by another bridge.

Rephrasing this problem in the language of graph theory, we get the problem of finding an *Eulerian trail* in the connected graph

Theorem 1.1. *A connected graph is Eulerian if and only if each vertex has even degree.*

Proof. □

Solution :

Since the degrees of all the vertices in the graph in the Konigsberg bridge problem are **not even**: The answer is that it is not possible to cross each of the seven bridges of Konigsberg exactly once and return to the starting point.

Problem. *The Konigsberg bridge problem could have been solved if* **one** *bridge was removed* **and** *another added.*

Which bridge would you remove and where would you add a bridge?

1.2 Fleury's Algorithm : Finding the Euler Path

Let $\mathbf{G}(\mathbf{V}, \mathbf{E})$ be a connected graph. If $\mathbf{G}(\mathbf{V}, \mathbf{E})$ is Eulerian, then Fluery's Algorithm will produce an **Eulerian trial** in $\mathbf{G}(\mathbf{V}, \mathbf{E})$.

STEP 1. We begin the algorithm with vertex **A**

STEP 2. Starting at **A**, choose **AB, BC, CD**. This gives the following $\mathbf{G}(\mathbf{V}, \mathbf{E})$ Figure 2b (with the current vertex circled)

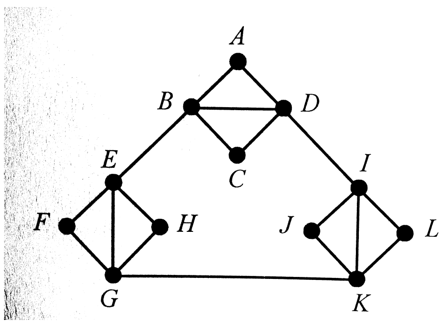
STEP 3. The edge **DA** is a ***bridge***; choosing **DB, BE, EF, FG** to produce the following graph: Figure 2c

STEP 4. Edge **GK** is a ***bridge***. Choosing **GE, EH, HG, GK, KI** to give the following : Figure 2d

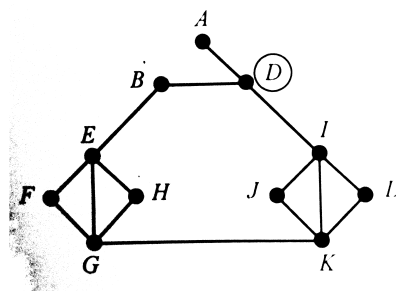
Algorithm 1 Fleury's Algorithm

Require: Let $\mathbf{G(V,E)}$ be an **Eulerian Graph**

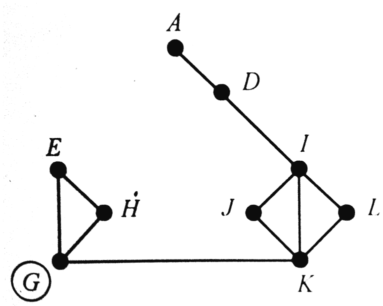
1: Choose any vertex $v \in V(G)$.
2: $currentVertex = v$, $currentTrial = \phi$ ▷ $currentTrial$ is a sequence of edges, $e \in E(G)$
3: Select any edge e **incident** to $currentVertex$. Choose a **bridge** only if there is no alternative
4: Add: $currentTrial = currentTrial \bigcup e$
5: Delete e from $\mathbf{G(V,E)}$. Delete any **isolated vertices**
6: Repeat Steps 2 - 5 until all edges have been deleted from $\mathbf{G(V,E)}$.
7: The final $currentTrial$ is an **Eulerian trial** in $\mathbf{G(V,E)}$



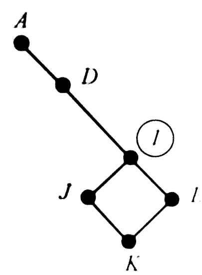
(a) Step 1



(b) Step 2



(c) Step 3



(d) Step 4

Figure 2: Fluery's Algorithm : Example

STEP 5. Edge **ID** is a ***bridge***. Choosing **IJ** followed by **JK, KL, LI, ID, DA**.

The complete trial is **ABCDBEFGEHGKIJKLIDA**

Theorem 1.2. *If* $\mathbf{G(V,E)}$ *is Eulerian, then any circuit constructed by Fluery's Algorithm is Eulerian.*

Proof. Let $\mathbf{G(V,E)}$ be an Eulerian graph. Let $C_p = v_0 e_1 \dots e_p v_p$ be the trial constructed by Fluery's Algorithm.

Then, clearly, the final vertex, v_p must be degree 0 in the graph G_p, and hence $v_p = v_t$ and C_p is the desired circuit of $\mathbf{G(V,E)}$. Now, to see that C_p is the desired circuit, suppose instead that C_p is not an Eulerian circuit of $\mathbf{G(V,E)}$. Thus, there must be edges of $\mathbf{G(V,E)}$ not in C_p.

Let S be the set of vertices of positive degree in G_p. Hence, $S \bigcap V(C_p)$ is non-empty since $\mathbf{G(V,E)}$ is connected and $v_p \in \overline{S} = V - S$. Let i be the largest integer such that $v_i \in S \bigcap C_p$ but $v_{i+1} \in \overline{S}$. Since, C_p ends in $\overline{S}$, it follows that $i < p$.

From the definition of $\overline{S}$, each edge of G_i that joins S and $\overline{S}$ is on C_p; thus the edge e_{i+1} is the only edge from S to $\overline{S}$ in the graph $\mathbf{G(V,E)}$ G_i. But then, $\underline{e_{i+1} \text{ is a } \textit{\textbf{bridge}} \text{ in } G_i}$.

Suppose, that e is any other edge of G_i that is incident to v_i.

Then from Algorithm 1: Step 3, it follows that e must also be a ***bridge*** of G_i (and hence of the graph H_i, induced by S in G_p). Since, $H_i \subseteq H_p$ (the graph induced by S in G_p), it follows that e is also a ***bridge*** in H_p.

Further, since e_{i+1} is a ***bridge*** of G_i and v_i is the last vertex on C_p that is also in S, we see that $H_i = H_p$ and that $deg_{H_p}(v) = deg_{G_p}(v) \ \ \forall v \in H_p$. Thus, every vertex in H_p has even degree, which implies that H_p has even degree, which implies that H_p contains *no* ***bridges***, a contradiction. $\square$

1.3 Eulerian Digraphs

Definition 1.2 (Eulerian Trial)**.** *An* **Eulerian Trial** *in a* $\mathbf{D(V,E)}$ *is a trail containing all edges*

An **Eulerian circuit** is a *closed trail* containing all edges.
A $\mathbf{D(V,E)}$ is Eulerian if it has an Eulerian circuit.

Theorem 1.3. *If* $\mathbf{D(V,E)}$ *is a digraph with* $\delta^+(D(V,G)) \geq 1$*, the* $\mathbf{D(V,E)}$ *contains a cycle. The same conclusion holds when* $\delta^-(D(V,G)) \geq 1$

Proof. Let P be a maximal path in $\mathbf{D(V,E)}$; and let $u \in P$ be the **last vertex** of P. Since, P cannot be extended, every successor of u must already be a vertex of P. Since, $\delta^+(D(V,G)) \geq 1$, u has a successor v on P.

The edge $\overrightarrow{uv}$ completes a cycle with the portion of P from v to u. $\square$

Theorem 1.4. *A* $\mathbf{D(V,E)}$ *is Eulerian iff* $d^+(v) = d^-(v)$ *for each vertex* $v \in V(D)$ *and the underlying graph has at most one non-trivial component.*

Theorem 1.5. *A digraph* $\mathbf{D(V,E)}$ *is Eulerian if and only if* $\mathbf{D(V,E)}$ *has at most one nontrivial component and* $d^+(v_x) = d^-(v_x)$ *for each vertex* $v \in V(D)$*.*

Proof. **Necessity :** Suppose $\mathbf{D(V,E)}$ is Eulerian. All edges are on a Eulerian cycle. Therefore, all edges are in one component. Other components have no edges. Thus, they are isolated vertices. For any vertex v_x in the nontrivial component, the number of edges leaving v_x is equal to the number of edges entering v_x. Thus, $d^+(v_x) = d^-(v_x)$

Sufficiency : We will prove it by induction on the number m of edges.

If $m = 0$, the Eulerian cycle is empty. It holds.

Suppose that the statement holds for any graph with at most m edges. In another words, if a graph $\mathbf{D(V,E)}$ with at most m edges has at most one nontrivial component and its vertices all have even degrees, then $\mathbf{D(V,E)}$ is Eulerian.

Now we consider a graph with $m + 1$ edges, which has at most one nontrivial component H and $d^+(v) = d^-(v) \geq 1$ for all $v \in V(H)$. By Lecture 3: Lemma 4.2, it contains a cycle C. Deleting all edges on C from $\mathbf{D(V,E)}$, H might be breaking into several components, say $H_1, H_2, \ldots, H_r$. It is clear that $d^+(v) = d^-(v)$ still holds for every vertex v.

Each component H_i has at most m edges. By inductive hypothesis, There is an Eulerian circuit C_i for each component H_i. Since $\mathbf{D(V,E)}$ has only one non-trivial component, the cycle C must intersect with every component H_i. Pick one vertex $v_i \in V(C) \bigcap V(H_i)$. The vertices $v_1, v_2, \ldots, v_r$ break the cycle C into r paths, say $v_1P_1v_2, v_2P_2v_3, \ldots, v_rP_rv_1$. Arrange

Eulerian circuit C_i so that the starting vertex and end vertex is v_i. Now we construct an Eulerian circuit as follows

$$C_1 P_1 C_2 P_2 \dots C_r P_r v_1$$

It contains all edges of $\mathbf{D}(\mathbf{V}, \mathbf{E})$.

□

Theorem 1.6. *Prove that a graph* $\mathbf{G}(\mathbf{V}, \mathbf{E})$ *has an Eulerian orientation if, and only if* $\mathbf{G}(\mathbf{V}, \mathbf{E})$ *is Eulerian*

Proof. **Contrapositive :** It suffices to ignore trivial and disconnected graphs. Thus, suppose that $\mathbf{G}(\mathbf{V}, \mathbf{E})$ is a connected graph that is not Eulerian. Then $\mathbf{G}(\mathbf{V}, \mathbf{E})$ has a vertex $v \in V(G)$ with $deg(v)$ an odd integer.

Let $\mathbf{D}(\mathbf{V}, \mathbf{E})$ be any digraph with $V(D) = V(G)$ obtained by providing an orientation to the edges of $\mathbf{G}(\mathbf{V}, \mathbf{E})$. In short, let $\mathbf{D}(\mathbf{V}, \mathbf{E})$ be any orientation of $\mathbf{G}(\mathbf{V}, \mathbf{E})$. Then there are a total of $deg(v)$ arcs in $E(D)$ with v as either initial element of the pair or terminal element of the pair.

Since there must be an odd number of these arcs, it follows that $d^+(v) \neq d^-(v)$. From Theorem 1.5, the digraph $\mathbf{D}(\mathbf{V}, \mathbf{E})$ cannot be Eulerian. Since $\mathbf{D}(\mathbf{V}, \mathbf{E})$ was an arbitrary orientation of $\mathbf{G}(\mathbf{V}, \mathbf{E})$, $\mathbf{G}(\mathbf{V}, \mathbf{E})$ cannot have an Eulerian orientation.

Direct Proof : Suppose that $\mathbf{G}(\mathbf{V}, \mathbf{E})$ is a Eulerian graph. Then $\mathbf{G}(\mathbf{V}, \mathbf{E})$ has an Eulerian circuit, say

$$C : u = v_0, \dots, v_k = u$$

.

Let $\mathbf{D}(\mathbf{V}, \mathbf{E})$ be the digraph with vertex set $V(D) = V(G)$ and whose arc set is $E(D) = \{(v_{i-1}, v_i) : i = 1, \dots, k\}$ with the v_i's from the Eulerian circuit $C(G)$ above.

Then $\mathbf{D}(\mathbf{V}, \mathbf{E})$ is an orientation for $\mathbf{G}(\mathbf{V}, \mathbf{E})$ since each edge of $\mathbf{G}(\mathbf{V}, \mathbf{E})$ appears exactly once in C, and $\mathbf{D}(\mathbf{V}, \mathbf{E})$ is Eulerian due to C actually being an Eulerian circuit for $\mathbf{D}(\mathbf{V}, \mathbf{E})$ as well for $\mathbf{G}(\mathbf{V}, \mathbf{E})$.

□

1.4 Application : deBruijn Cycle

8
De Bruijn sequences

The following problem has a practical origin: the so-called *rotating drum problem*. Consider a rotating drum as in Fig. 8.1.

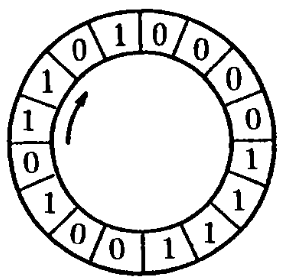


Figure 8.1

Each of the segments is of one of two types, denoted by 0 and 1. We require that any four consecutive segments uniquely determine the position of the drum. This means that the 16 possible quadruples of consecutive 0's and 1's on the drum should be the binary representations of the integers 0 to 15. Can this be done and, if yes, in how many different ways? The first question is easy to answer. Both questions were treated by N. G. de Bruijn (1946) and for this reason the graphs described below and the corresponding circular sequences of 0's and 1's are often called *De Bruijn graphs* and *De Bruijn sequences*, respectively.

We consider a digraph (later to be called G_4) by taking all 3-tuples of 0's and 1's (i.e. 3-bit binary words) as vertices and joining the vertex $x_1x_2x_3$ by a directed edge (arc) to x_2x_30 and x_2x_31. The arc $(x_1x_2x_3, x_2x_3x_4)$ is numbered e_j, where $x_1x_2x_3x_4$ is the binary representation of the integer j. The graph has a loop at 000 and at 111. As we saw before, the graph has an Eulerian circuit because every vertex has in-degree 2 and out-degree 2. Such a closed path produces the required 16-bit sequence for the drum. Such a (circular) sequence is called a De Bruijn sequence. For example the path $000 \to 000 \to 001 \to 011 \to 111 \to 111 \to 110 \to 100 \to 001 \to 010 \to 101 \to 011 \to 110 \to 101 \to 010 \to 100 \to 000$ corresponds to 0000111100101101 (to be read circularly). We call such a path a *complete cycle*.

We define the graph G_n to be the directed graph on $(n-1)$-tuples of 0's and 1's in a similar way as above. (So G_n has 2^n edges.)

The graph G_4 is given in Fig. 8.2. In this chapter, we shall call a digraph with in-degree 2 and out-degree 2 for every vertex, a '2-in 2-out graph'. For such a graph G we define the 'doubled' graph G^* as follows:

(i) to each edge of G there corresponds a vertex of G^*;

(ii) if a and b are vertices of G^*, then there is an edge from a to b if and only if the edge of G corresponding to a has as terminal end (head) the initial end (tail) of the edge of G corresponding to b.

Clearly $G_n^* = G_{n+1}$.

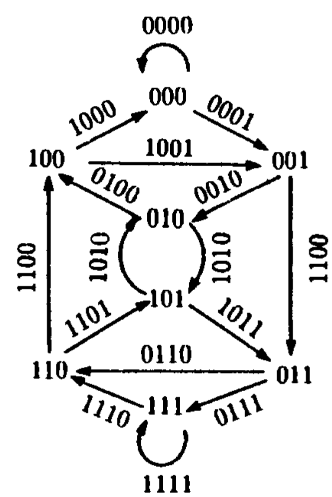


Figure 8.2

Theorem 8.1. *Let G be a 2-in 2-out graph on m vertices with M complete cycles. Then G^* has $2^{m-1}M$ complete cycles.*

Proof: The proof is by induction on m.

(a) If $m = 1$ then G has one vertex p and two loops from p to p. Then $G^* = G_2$ which has one complete cycle.

(b) We may assume that G is connected. If G has m vertices and there is a loop at *every* vertex, then, besides these loops, G is a circuit $p_1 \to p_2 \to \cdots \to p_m \to p_1$. Let A_i be the loop $p_i \to p_i$ and B_i the arc $p_i \to p_{i+1}$. We shall always denote the corresponding vertices in G^* by lower case letters. The situation in G^* is as in Fig. 8.3.

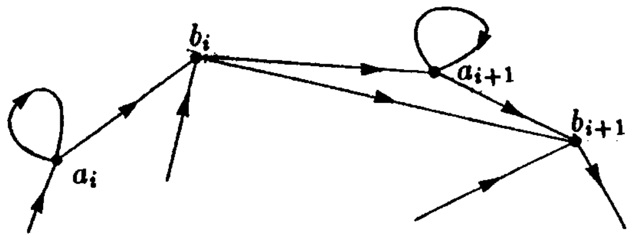


Figure 8.3

Clearly a cycle in G^* has two ways of going from b_i to b_{i+1}. So G^* has 2^{m-1} complete cycles, whereas G has only one.

(c) We now assume that G has a vertex x that does not have a loop on it. The situation is as in Fig. 8.4, where P, Q, R, S are different edges of G (although some of the vertices a, b, c, d may coincide).

From G we form a new 2-in 2-out graph with one vertex less by deleting the vertex x. This can be done in two ways: G_1 is obtained by the identification $P = R$, $Q = S$, and G_2 is obtained by $P = S$, $Q = R$. By the induction hypothesis, the theorem applies to G_1 and to G_2.

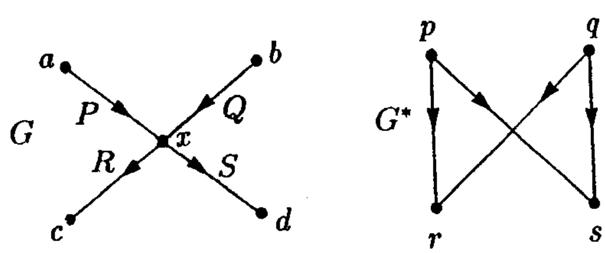


Figure 8.4

There are three different types of complete cycle in G^*, depending on whether the two paths leaving r and returning to p, respectively q, both go to p, both to q, or one to p and one to q. We treat one case; the other two are similar and left to the reader. In Fig. 8.5 we show the situation where path 1 goes from r to p, path 2 from s to q, path 3 from s to p, and path 4 from r to q.

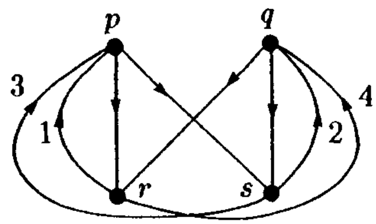


Figure 8.5

These yield the following four complete cycles in G^*:

$$\begin{array}{cccccccc} 1, & pr, & 4, & qs, & 3, & ps, & 2, & qr \\ 1, & ps, & 2, & qr, & 4, & qs, & 3, & pr \\ 1, & ps, & 3, & pr, & 4, & qs, & 2, & qr \\ 1, & ps, & 2, & qs, & 3, & pr, & 4, & qr \end{array}$$

In G_1^* and G_2^* the situation reduces to Fig. 8.6.

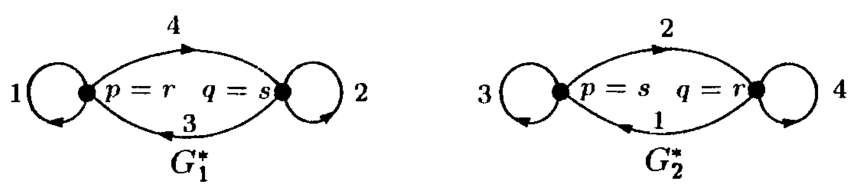


Figure 8.6

In each of G_1^* and G_2^* *one* complete cycle using the paths $1, 2, 3, 4$ is possible. In the remaining two cases, we also find two complete cycles in G_1^* and G_2^* corresponding to four complete cycles in G^*. Therefore the number of complete cycles in G^* is twice the sum of the numbers for G_1^* and G_2^*. On the other hand, the number of complete cycles in G is clearly equal to the sum of the corresponding numbers for G_1 and G_2. The theorem then follows from the induction hypothesis. $\square$

We are now able to answer the question how many complete cycles there are in a De Bruijn graph.

Theorem 8.2. *G_n has exactly $2^{2^{n-1}-n}$ complete cycles.*

PROOF: The theorem is true for $n = 1$. Since $G_n^* = G_{n+1}$, the result follows by induction from Theorem 8.1. □

For a second proof, see Chapter 36.

Problem 8A. Let α be a primitive element in $\mathbb{F}_{2^n}$. For $1 \le i \le m := 2^n - 1$, let

$$\alpha^i = \sum_{j=0}^{n-1} c_{ij}\alpha^j.$$

Show that the sequence

$$0, c_{10}, c_{20}, \ldots, c_{m0}$$

is a De Bruijn sequence.

Problem 8B. Find a circular ternary sequence (with symbols $0, 1, 2$) of length 27 so that each possible ternary ordered triple occurs as three (circularly) consecutive positions of the sequence. First sketch a certain directed graph on 9 vertices so that Eulerian circuits in the graph correspond to such sequences.

Problem 8C. We wish to construct a circular sequence $a_0, \ldots, a_7$ (indices mod 8) in such a way that a sliding window a_i, a_{i+1}, a_{i+3} ($i = 0, 1, \ldots 7$) will contain every possible three-tuple once. Show (not just by trial and error) that this is impossible.

Problem 8D. Let $m := 2^n - 1$. An algorithm to construct a De Bruijn sequence $a_0, a_1, \ldots, a_m$ works as follows. Start with $a_0 = a_1 = \cdots = a_{n-1} = 0$. For $k > n$, we define a_k to be the maximal value in $\{0, 1\}$ such that the sequence $(a_{k-n+1}, \ldots, a_{k-1}, a_k)$ has not occurred in $(a_0, \ldots, a_{k-1})$ as a (consecutive) subsequence. The resulting sequence is known as a *Ford sequence*. Prove that this algorithm indeed produces a De Bruijn sequence.

Notes.

Although the graphs of this chapter are commonly called De Bruijn graphs, Theorem 8.1 was proved in 1894 by C. Flye Sainte-Marie. This went unnoticed for a long time. We refer to De Bruijn (1975).

2 Hamiltonian Graphs

Check ICOSIAN Game

Definition 2.1. *A cycle in a graph* $\mathbf{G(V,E)}$ *that contains every* $v \in V(G)$ *is called a* **Hamiltonian cycle,** $\mathbf{H_n}$ *of* $\mathbf{G(V,E)}$ *, where* $n = |V(G)|$.

Thus, a Hamiltonian cycle of $\mathbf{G(V,E)}$ is a spanning cycle of $\mathbf{G(V,E)}$.

Definition 2.2. *A* **Hamiltonian graph,** $\mathbf{G(V,E)}$ *is a graph that contains a* **Hamiltonian cycle**.

Definition 2.3. *A path is a graph* $\mathbf{G(V,E)}$ *that contains every vertex of* $\mathbf{G(V,E)}$ *is called a* **Hamiltonian path** *in* $\mathbf{G(V,E)}$

Note : If a graph contains a **Hamiltonian cycle**, then it contains a hamiltonian path **BUT** if a graph contains a Hamiltonian path, it need not contain a **Hamiltonian cycle** e.g $\mathbf{P_n}$ (Path of order $|P_n|$)

Features :

- A Hamiltonan graph of order $|H_n|$ consists of a cycle C_n of length $|C_n|$, with some additional edges joining non-consecutive vertices of C_n.
- A Hamiltonian cycle, H_n in a graph $\mathbf{G(V,E)}$ or order $|V(G)| \geq 3$ is a connected 2-regular subgraph (all vertices having same degree) of order $|H_n|$, every proper subgraph of H_n is a path or a (disjoint) union of paths.

Results :

- H_n contains no cycle of order less than H_n as a subgraph, and certainly H_n contains no subgraph, if it has $deg(v) = 2, v \in V(G)$, then both edges of $\mathbf{G(V,E)}$ incident with v must lie in H_n.

Theorem 2.1. *The Peterson Graph is Non-Hamiltonan.*

Proof.
- Peterson Graph is a 3-regular graph of order 10.
- It can be considered as being constructed of two 5-cycles: $C_5' = u_1u_2u_3u_4u_5$ and $C_5'' = v_1v_2v_3v_4v_5$ and joining edges u_1v_1, u_2v_2, u_3v_3, u_4v_4 and u_5v_5 or $E = \{\bigcup e_i = u_iv_i | 1 \leq i \leq 5\}$

Suppose, that Peterson Graph is Hamiltonian. Then, paterson Graph contains a Hamiltonian cycle, H_{10}. which contains 10 edges. Therefore, two of the 3 edges incident with each vertex of Peterson graph necessarily belongs to H_{10}. Clearly, then H_{10} contains all five, some or none of the edges $e_i = u_iv_i; 1 \leq i \leq 5$; so at least 5 edges of H_{10} belongs to either C_5' or C_5''. Therefore, either C_5' contains at least 3 edges of H_{10} or C_5'' contains at least 3 edges of H_{10}.

Without, loss of generality, let us assume that C_5' contains at least 3 edges of H_{10}. Observe, that all 5 edges of C_5' cannot belong to H_{10}, since so sycle contains a smaller cycle as a subgraph.

Say, H_{10} contains exactly 4 edges of C_5' : $u_1u_2, u_2u_3, u_4u_5, u_5u_1$. However, the cycle H_{10} must contain the edges u_4v_4, u_3v_3 as well as v_1v_3, v_1v_4.

But, this implies that H_{10} contains an 8-cycle, which is "contradiction".

Case remains that H_{10} contains exactly 3 edges of C_5'. There are two possibilities :

1. the 3 edges of C_5' on H_{10} are consecutive on C_5' and
2. these 3 edges are not consecutive on C_5'

Figure 3d : Impossible as u_1v_1 is the only edge incident with u_1 that could lie on H_{10}

Figure 3e : H_{10} would have to contain the smaller cycle $u_4v_4v_1v_3u_3v_4$

Therefore, as claimed, the Peterson Graph **is NOT** Hamiltonian.

$\square$

Why determining whether a graph contains a Hamiltonian cycle is **difficult** ?

While there is a simple characterization of Eulerian graph, i.e. a nontrivial connected graph $\mathbf{G(V,E)}$ is Eulerian iff every vertex of $\mathbf{G(V,E)}$ has even degree, there is no such characterization of Hamiltonian graphs.

When there is no characterization of graphs possessing a certain property, one looks for sufficient conditions for a graph to have such a property.

2.1 Closure of a Graph

The Closure $Clsr(G(V,E))$ of a graph, $\mathbf{G(V,E)}$ of order $|V(G)|$ is the graph obtained from $\mathbf{G(V,E)}$ by recursively joining pairs of non-adjacent vertices whose degree sum is at least $|V(G)|$ (in the resulting graph at each stage) until no such pair remains.

Theorem 2.2. *A graph* $\mathbf{G(V,E)}$ *is Hamiltonian iff its closure is Hamiltonian*

Corollary 2.3. *If* $\mathbf{G(V,E)}$ *is a graph of order at least 3 such that* $Clsr(G(V,E))$ *is complete, then* $\mathbf{G(V,E)}$ *is Hamiltonian.*

2.2 Weighted Hamiltonian Graphs

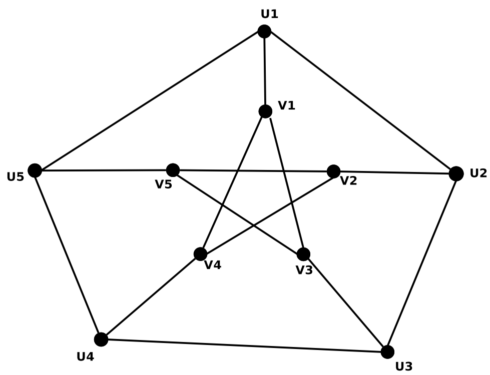
There exists two practical problems realted to Weighted Hamiltonian graphs

Chinese Postman Problem : A postman starts from a post office to deliver mail. After visiting a number of streets, he comes back.

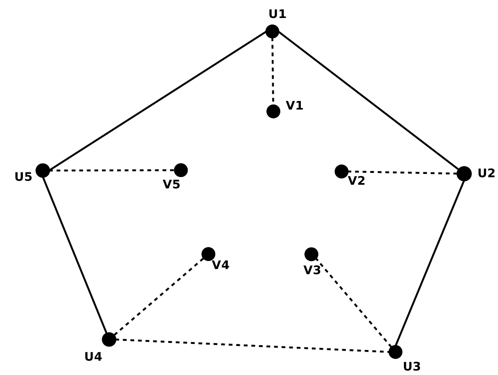
What route should he take so that he visits every street at least once and the total distance covered is least.

Travelling Salesman Problem : A travelling Salesman visits a number of cities and comes back to his head office. What route should he travel so that he comes back after visiting each of the cities exactly once and covers the minimumdistance/covers the distance with minimum expense.

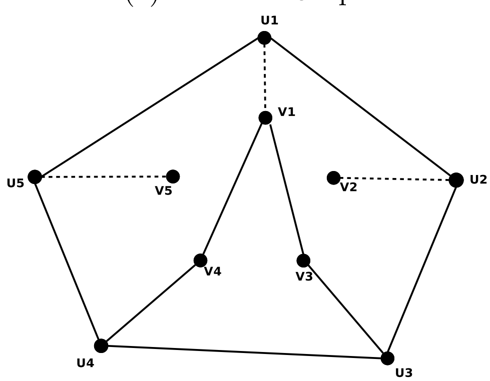
Note : Relation between Hamiltonian and Eulerian graphs – Line Graphs



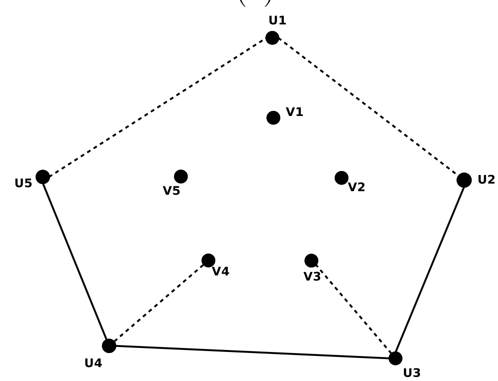
(a) Peterson Graph



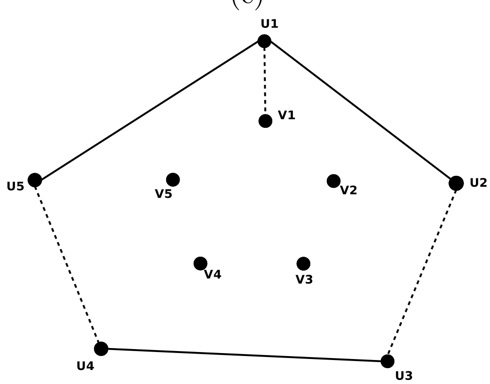
(b)



(c)



(d)



(e)

Figure 3: Peterson Graph – Is it Hamiltonan ??

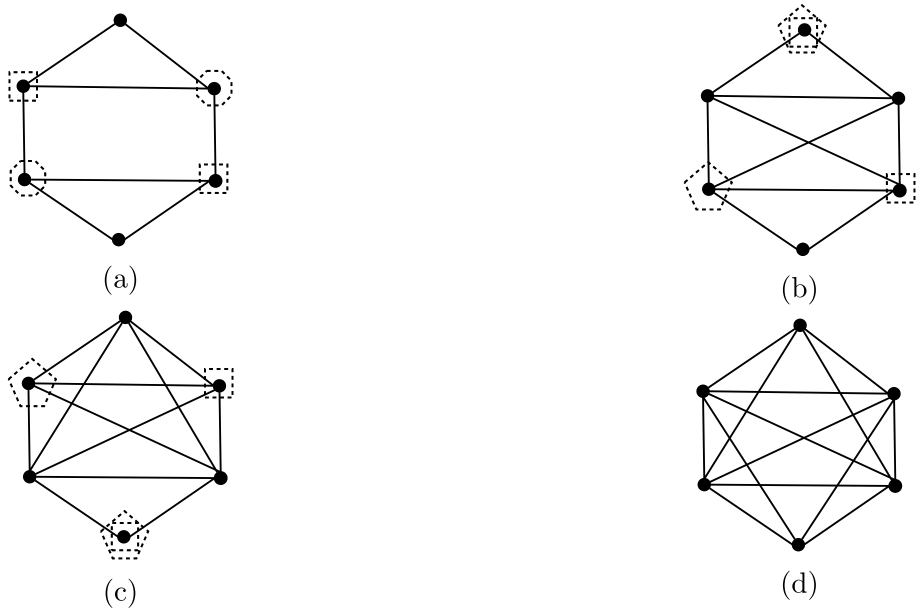


Figure 4: Example 1: Closure of Graph

Figure 5: Example 2: Closure of Graph