The Government of the Union of Myanmar
Ministry of Education

Department of Higher Education (Lower Myanmar)
and
Department of Higher Education (Upper Myanmar)

Universities Research Journal

Vol. 1, No. 2 December, 2008.

Universities Research Journal 2008
Vol. 1, No. 2

Contents

Applications of Eulerian Path and Tour

Sandar Myint[1] , Win Win Mar[2] , Yi Myint[3]

Abstract

A path in a graph is an Eulerian path if every edge of the graph appears as an edge in the path exactly once. A closed Eulerian path is an Eulerian tour. A graph is said to be an Eulerian graph if it has an Eulerian tour. In this paper, we give necessary and sufficient conditions for a graph to be Eulerian graph and present some examples for applications of Eulerian path and tour.

Key words: path, tour, degree

Introduction

Problems of finding paths and tours on graphs are of fundamental importance and found many applications. An Eulerian tour is a tour which traverses each edge exactly once. There are several interesting and useful applications of Eulerian paths and tour in many areas, such as computer science, cryptograph and transportation problem. In this paper we consider a postman problem. A postman problem is that a postman starts from his office, delivers the mail along all the streets and then goes back to the office minimizing the length of his walk. In graph theoretical terms, a postman problem consists of finding a minimum cost tour of a graph traversing all its directed edges and undirected edges at least once. We can solve the postman problem by using an Eulerian tour in the graph.

Preliminaries

This section collects some mathematical preliminaries (i.e., some basic definitions and notations) which are necessary for our future discussions.

Let V be a nonempty finite set and E a finite family of (not necessarily different) unordered pairs {i, j} of (not necessarily different) elements of V. Then we call the pair **G = (V , E) an (undirected) graph,** V be the vertex set of G and E the edge set of G respectively. We use the symbols $v(G)$ and ε (G) to denote the numbers of vertices and edges in graph G. The number of elements in V is called the order of G. If the unordered pair

[1] Lecturer, Department of Mathematics, West Yangon University
[2] Lecturer, Department of Mathematics, West Yangon University
[3] Professor & Head , Dr., Department of Mathematics, West Yangon University

e = {i, j} is an **edge** of G, then i and j are the end points of e. We also say that e joins i and j or e is incident with i and j or i and j are adjacent. Edges of the form {i, i} with $i \in V$ are **loops**. If e_1 and e_2 are two different edges that have the same end points (i.e., $i(e_1) = i(e_2)$), then we call e_1 and e_2 **parallel edges.** A graph is called a **simple graph** if it has no parallel edges or loops.

Let G = (V, E) be a graph. Then **walk** W of length $n \geq 0$ in G is a sequence $v_0e_1v_1 \ldots v_{n-1}e_nv_n$ such that each $v_k \in V$, each $e_k \in E$, and for each k from 1 to n, edge e_k joins vertices v_{k-1} and v_k. In a simple graph, a walk $v_0e_1v_1 \ldots e_kv_k$ is determined by the sequence $v_0v_1 \ldots v_k$ of its vertices; hence a walk in a simple graph can be specified simply by its vertex sequence. The walk W is said to join v_0 and v_n. If the edges in W are all distinct, then W is called a **trail**. If the vertices in W are all distinct, then W is called a **path**. If $v_0 = v_n$ and $n \geq 1$, then W is said to be **closed**. A nontrivial closed trail of a graph G is referred to as a **circuit of G** and a circuit $v_1, v_2, \ldots, v_n, v_1$ ($n \geq 3$) whose n vertices v_i are distinct is called a **cycle**. A **tour** of G is a closed walk that traverses each edge of at least once.

The **degree** of a vertex in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex. The degree of the vertex v is denoted by d(v).

Two vertices u and v of G are said to **connected** if there is a (u, v) path in G. There is a partition of V into nonempty subsets $V_1, V_2, \ldots, V\omega$ such that two vertices u and v are connected if and only if both u and v belong to the same set V_i. The subgraphs $G[V_1], G[V_2], \ldots, G[V\omega]$ are called the **components** of G. If G has exactly one component, G is connected, otherwise G is disconnected. We denote the number of components of G by ω (G).

A **cut edge** of G is an edge e such that $\omega(G-e) > \omega(G)$.

A **directed graph or digraph D=(V, A)** is a set of nodes V along with a multiset of arcs A that connect the nodes in V. An **arc or directed edge** (u, v) is an ordered pair of nodes that are connected in G. If (u, v) is an arc in A, then u is called its tail and v is called its head. The arc is said to go from u to v.

A digraph D is **strongly connected** if whenever u and v are vertices of D, there is a walk from u to v and a walk from v to u, otherwise D is **weakly connected**. With each edge e of G let there be associated a real number a_e, called its **weight (or cost, or length).** Then G, together with these weights on its edges, is called a **weighted graph.**

Some Theorems on Eulerian Graph

An **Euler trail** in a connected graph G is a trail that contains every edge of G.

An **Euler tour (or Euler circuit) is a tour (or circuit)** which traverses each edge exactly once (in other words, a closed Euler trail).

A graph is **Eulerian** if it contains an Euler tour (or Euler circuit).

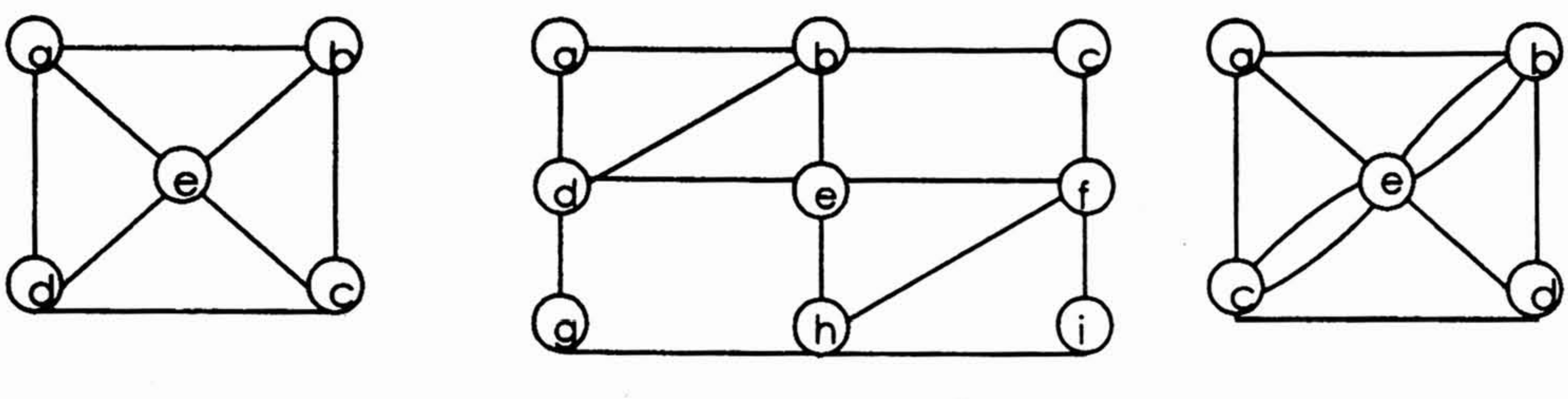
An **Eulerian trail of digraph D** is an open trail of D containing all the arcs and vertices of D.

An **Eulerian circuit of digraph D** is a circuit containing every arc and vertex of D.

A digraph that contains an Eulerian circuit is called an **Eulerian digraph.**

Example 1

Consider the following graphs.

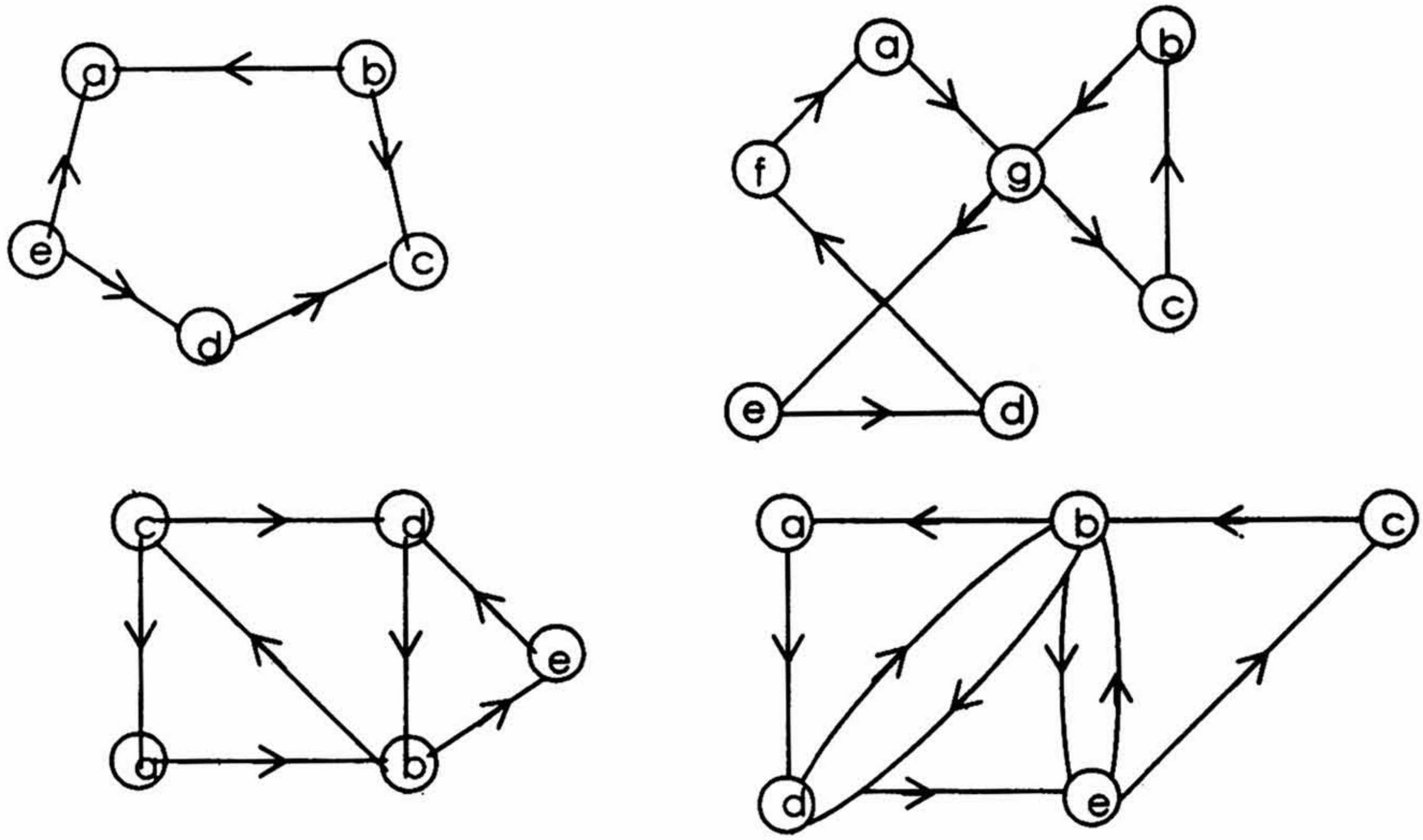


G_1 **G_2** **G_3**

Figure 1 The undirected graphs G_1, G_2 and G_3

The graph G_1 does not have an Euler trail. G_2 has Euler tours a,b,d,e,b,c,f,e,h,f,i,h,g,d,a. G_3 has Euler trails a,b,e,c,a,e,b,a,e,c,d.

Example 2



H_3 H_4

Figure 2 The digraphs H_1, H_2, H_3 and H_4

H_1 does not have an Euler trail. H_3 has Euler trail c,a,b,e,d,b,c,d,b. H_2 and H_4 have Euler tours a,g,c,b,g,e,d,f,a and a,d,b,d,e,b,e,c,b,a.

Theorem 1

Let G be a nontrivial connected graph. Then G is Eulerian if and only if the degree of every vertex of G is even.

Theorem 2

Let G be a nontrivial connected graph. Then G contains an Eulerian trail if and only if G has exactly two odd degree vertices. Furthermore, the trail begins at one of these odd degree vertices and terminates at the other.

Theorem 3

A nontrivial connected graph G is Eulerian if and only if E(G) can be partitioned into subsets E_i , $1 \leq i \leq k$, where each subgraph $\langle E_i \rangle$ is a cycle.

Theorem 4

A nontrivial connected graph G is Eulerian if and only if every edge of G lies on an odd number of cycles.

Applications of Euler Tour

The Postman Problem

A postman starts from his office, delivers the mail along the streets, and then goes back to the office. The problem is to design a tour of minimum length for the problem. This problem was first discovered by a Chinese mathematician, Kwan Mei-Ko, is known as the **Chinese Postman Problem.**

For a mathematical formulation of this problem, we represent the transport network by a graph, inter-city connections (streets in the town) correspond to the edges of the graph, cities (intersections of the streets) correspond to the nodes and we interpret the inter-city distances (the lengths of the streets) as the weight of the corresponding edges. Under this modeling, a postman's tour corresponds to a closed walk containing each of the graph at least once.

In a weighted graph, we define the weight of a tour $v_0e_1v_1...e_nv_n$ to be $\sum_{i=1}^{n} a(e_i)$.

Clearly, the Chinese postman problem is just that of finding a minimum-weigh tour in a weighted connected graph with non-negative weights. We shall refer to a such a tour as an optimal tour.

If G is Eulerian, then any Euler tour of G is an optimal tour because an Euler tour is a tour that traverses each edge exactly once.

The Chinese postman problem is easily solved in this case, since there exist a good algorithm for determining an Euler tour in an Eulerian graph. The algorithm, due to Fleury, constructs an Euler tour by tracing out a trail, subject to the once condition that, at any stage, a cut edge of the untraced subgraph is taken only if there is no alternative.

Fleury's Algorithm

1. Choose an arbitrary vertex v_0 , and set $W_0 = v_0$.
2. Suppose that the trail $W_i = v_0e_1v_1 \ldots e_iv_i$ has been chosen.

   Then choose an edge e_{i+1} from $E\backslash\{e_1, e_2, \ldots, e_i\}$ in such a way that

   (i) e_{i+1} is incident with v_i;

   (ii) unless there is no alternative, e_{i+1} is not a cut edge of

   $G_i = G - \{e_1, e_2, \ldots, e_i\}$
3. Stop when step 2 can no-longer be implemented.

   By its definition, Fleury's algorithm constructs a trail in G.

Theorem 5

If G is Eulerian, then any trial in G constructed by Fleury's algorithm is an Euler tour of G.

An Application of Eulerian Graphs to Coding Theory

Any word with m letters out of which n are distinct can be associated with a weakly connected digraph D with n vertices and m1 arcs such that the word represents a directed Eulerian path if the first letter and the last letter are different and Eulerian tour if the first letter and the last letter are the same. For example, in the word "MYANMAR" we have m= 7 and n=5, and this letter can be associated with the directed path from the vertex M to the vertex R in the digraph of Figure 3 with five vertices and six arcs represented by M…Y…A…N…M…A…R , which is a directed Eulerian path. Similarly, the word "MAXIMUM" can be associated with a directed Eulerian cycle M…A…X…I…M…U…M in the digraph in Figure 4. Notice that even though a word defines a digraph uniquely, it is possible that the same digraph can be associated with several words of equal length.

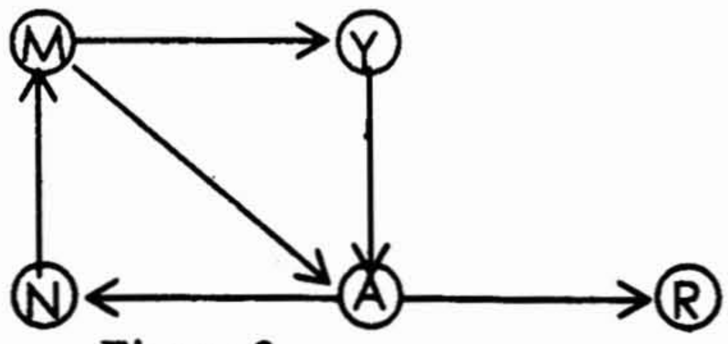


Figure 3

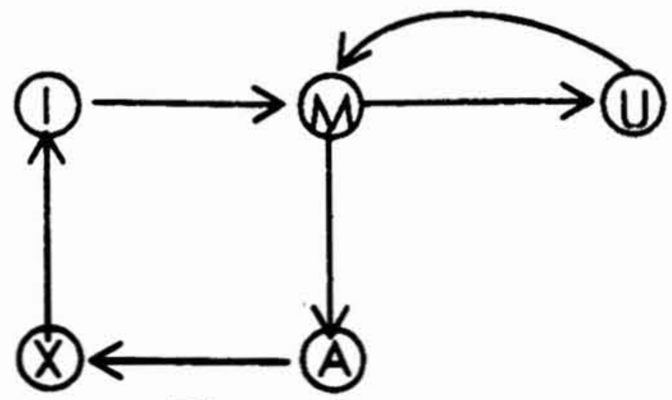


Figure 4

For example, in the word "MYANMAR" the five distinct letters are A, M, N, R and Y with a frequency set {2, 2, 1, 1, 1} , and the 5 × 5 matrix is

M =	A	M	N	R	Y	Row Sum
A	0	0	1	1	0	2
M	1	0	0	0	1	2
N	0	1	0	0	0	1
R	0	0	0	0	0	0
Y	1	0	0	0	0	1
Column Sum:	2	1	1	1	1	

In the word "MAXIMUM" the five distinct letters are A, I, M, U and X with a frequency set {1, 1, 3, 1, 1} and the 5 × 5 matrix is

M =	A	I	M	U	X	Row Sum
A	0	0	0	0	1	1
I	0	0	1	0	0	1
M	1	0	0	1	0	2
U	0	0	1	0	0	1
X	0	1	0	0	0	1
Column Sum:	1	1	2	1	1	

We now make the following easily verifiable assertions:

1. The digraph of any word is weakly connected.
2. If the first letter and the last letter are not the same, the row sum of the first letter equals the column sum of the first letter plus one, the row sum of the last letter equals the column sum of the last letter minus one, and for all other letter the row sum and the column sum are equal.
3. If the first letter and the last letter are the same, then row sum equals the column sum for all letters and the row sum of the starting letter will be one less than its frequency.

All the information in a codeword is contained in the frequency set and the matrix M associated with the word. Suppose that we given (a) a set of n letters $A_1, A_2, \ldots, A_n$ (b) a set of n positive integers $f_1, f_2, \ldots, f_n$, and (c) a $n \times n$ matrix (m_{ij}) of non negative integers. If conditions corresponding to assertions (1)-(3) are satisfied there exists a word in which A_i appears exactly f_i times and A_j appears immediately after A_i exactly m_{ij} times. We thus have the following result.

Theorem 6

Let $M = (m_{ij})$ be a non-negative integer components and let A_i (i =1, 2, …, n) be a set of n distinct letters such that A_i is associated with both row i and column i . Let

r_i = sum of all the elements of row i of M

c_j = sum of all the elements of column j of M.

(a) If $r_j = c_j+1$, $r_k = c_k - 1$, where j and k are distinct and if $r_i = c_i$ in all other cases, there exists a word beginning with A_j and ending in A_k in which the frequency of A_j and r_j, the frequency of A_k is c_k , and the frequency of every other letter is r_i, which is also c_i. Moreover, in the word, the letter A_p appears immediately after A_q exactly m_{pq} times.

(b) If $r_i = c_i$ for i = 1, 2, …, n and if f_i(i = 1, 2, …, n) are non-negative integers such that $r_i = f_i - 1$ and $r_i = f_i$ for every i other than j, there exists a word that begins with A_j and end with A_j in which A_k appears exactly f_k times and A_p appears after A_q exactly m_{pq} times.

Example 3

Suppose that the distinct letter in a word are A, B, C, D and E and the matrix is

$$
\begin{array}{c} \\ A \\ B \\ M = C \\ D \\ E \\ \text{Column Sum:} \end{array}
\begin{array}{c} \begin{array}{ccccc} A & B & C & D & E \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \\ \begin{array}{ccccc} 2 & 2 & 2 & 3 & 1 \end{array} \end{array}
\begin{array}{c} \text{Row Sum} \\ 3 \\ 1 \\ 2 \\ 3 \\ 1 \\ \\ \end{array}
$$

First we construct a digraph D with five vertices A, B, C, D and E as in Figure 5.

We observe

1. The digraph is weakly connected
2. (Row sum for A) = (column sum for A) + 1
3. (Row sum for B) = (column sum for B) – 1
4. Row sum = Column sum for all other letters

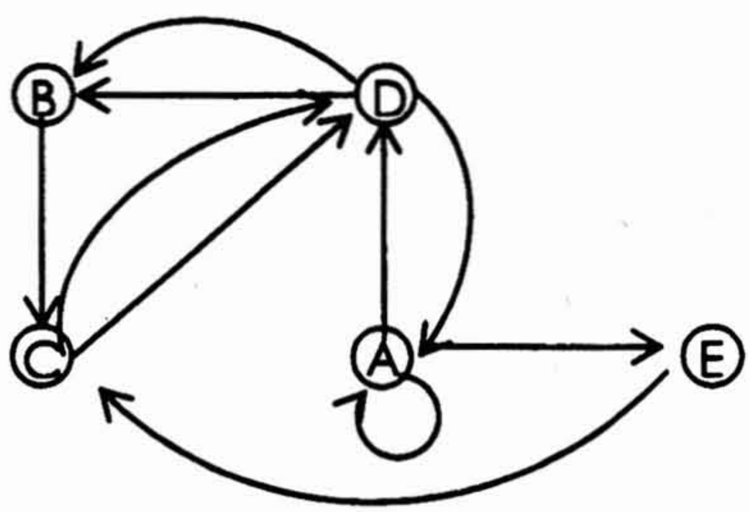


Figure 5

So there is a word that starts with A and ends in B which can be represented by an Eulerian path from A to B in the digraph of Figure 5. One such word is ADBCDAAECDB.

Example 4

If the frequency set is {3, 4, 6, 4, 4} and the matrix is

	A	B	C	D	E	Row Sum
A	1	0	0	1	0	2
B	0	0	1	0	1	2
M = C	0	1	1	1	1	4
D	1	0	1	0	0	2
E	0	1	1	0	0	2
Column Sum:	2	2	4	2	2	

Then we can find a word as follow; first we construct a digraph D with five vertices A, B, C, D and E as in Figure 6.

We observe:

1. The digraph is weakly connected
2. (Row sum for A) = (Frequency for A) – 1
3. Row sum = Column sum for all other letters.

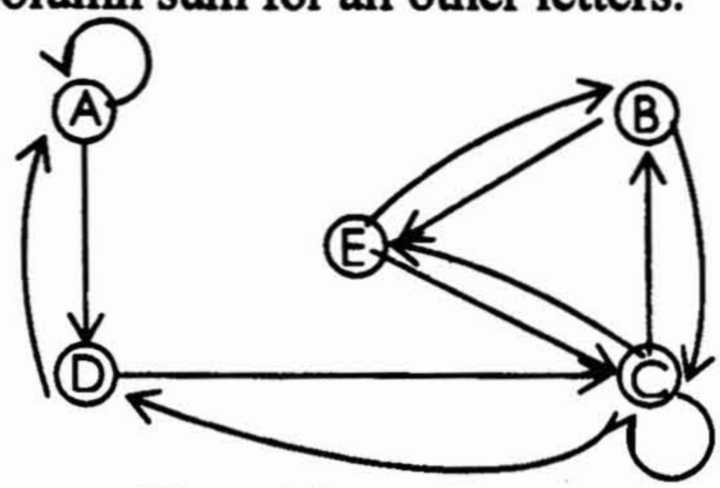


Figure 6

So there is a word that start and end in A which can be represented by an Eulerian tour from A to A in the digraph of Figure 6. One such word is AADCCBECEBCDA.

We conclude this section with a discussion of de Bruijn digraphs, another application of Eulerian digraphs.

de Bruijn Sequence

There are 2^{n-1} binary words of length n-1. We construct a digraph with 2^{n-1} vertices as follows. Let each word of length n-1 be a vertex. From each vertex of the form $v = a_1a_2 \ldots a_{n-1}$ draw two arcs: one to $a_2a_3 \ldots a_{n-1}\,0$ and the other to $a_2a_3 \ldots a_{n-1}\,1$ to represent two n-1 letter words v_0 and v_1, respectively. So the 2^n arcs of the digraph thus constructed represent the set of binary words of length n. This digraph D(2, n) known as the **de Bruijn digraph,** is weakly connected and is Eulerian since the indegree of each vertex equals its outdegree. The digraph D(2, 3) is as shown in Figure 7.

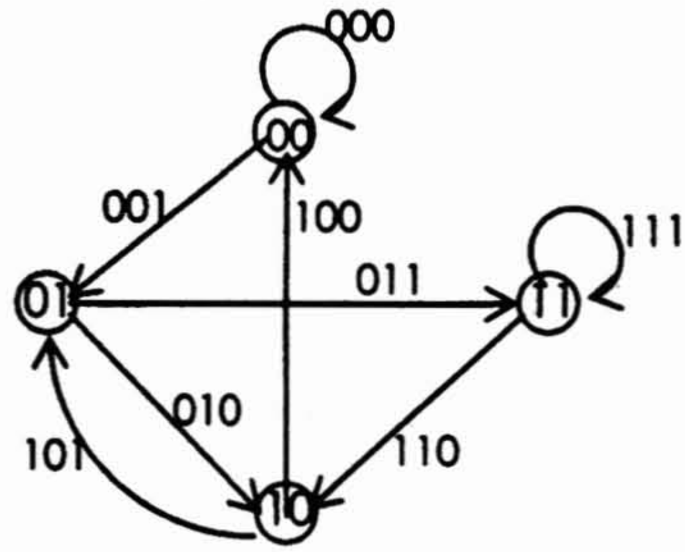


Figure 7

More generally, for an alphabet of p letters, D(p, n) is a de Bruijn digraph with p^{n-1} vertices and p^n arcs such that the indegree and outdegree of each vertex are both p. Thus D(p, n) is Eulerian. Now consider any Eulerian tour in this digraph that will contain all the p^n arcs in a sequence. Construct the sequence of the first letters of all these words. Let us denote this sequence by $a_1a_2 \ldots a_r$ where $r = p^n$. Then the r distinct word of length n are all of the form $a_ia_{i+1} \ldots a_{i+n-1}$, where the addition operation defined on the subscript is modulo r. For example, if p = 2 and n = 3, then a a_9 is a_1. In the digraph of Figure 7 a directed Eulerian tour starting from 00 consists of the following sequence of eight arcs: 000,001,011,111,110,101,010,100. The first letters of these arcs form the word 00011101, so that $a_1 = 0$, $a_2 = 0$, $a_3=0$, $a_4 = 1$, $a_5 = 1$, $a_6 =1$, $a_7 = 0$ and $a_8 = 1$. Any three-letter word is now of the form $a_i\, a_{i+1}\, a_{i+2}$. Thus $a_7a_8a_9 = a_7a_8a_1 = 010$, and so on.

We can formally define a de Bruijn sequence for two positive integers p and n. If S is any alphabet consisting of p letters, then a sequence $a_1a_2 \ldots a_r$ of r ($r = p^n$) letters is called a **de Bruijn Sequence**, denoted by B(p, n), if every word of length n from S can be realized as $a_ia_{i+1} \ldots a_{i+n-1}$ ($i = 1, 2, \ldots, r$), where the addition operation in the subscripts is modulo r.

We summarize above observations that **for every pair of positive integers there exists a de Bruijn sequence.**

Conclusion

In the previous sections we have presented the problems of finding path and tour on graphs. The study of routing problems is an important area of operations research and remaining to be done efficient methods to solve these problems.

References

Bondy, J.A.& Murty, U.S.R., (1976), *Graph theory with applications*. American Elesevier. New York.

Buckley, F. & Harary, F., (1990), *Distance in graphs*. Addison-Wesley, New York.

Harary, F., (2000), *Graph theory*. Narosa Publishing House. New Delhi.

Rosen, K.H. (2003), *Discrete Mathematics and Its application,* Mc. Graw-Hill, Inc., New York

Win. Z. (1987), *Contributions to Routing Problems,* Ph. D. Dissertation, University of Augsburg, Germany.