Journal of Contemporary Applied Mathematics
V. 2, No 2, 2012, December
ISSN 2222-5498

A Theoretical Characterization of the Data Structure 'Multigraph'

Siddhartha Sankar Biswas*, Bashir Alam, M. N. Doja

Abstract. Multigraph is a non-linear data structure, a generalization of the important data structure 'graph'. There are many real life situations, in particular in the network systems, which cannot be modeled into graphs but can be well modeled into multigraphs. Consequently, those situations cannot be dealt with the theory of graphs, but by the theory of multigraphs only. It is quite obvious that since the multigraphs are a generalized model of graphs, the theories and properties of graphs cannot be automatically granted to hold good in the theory of multigraphs, unless studied and verified rigorously in the context of multigraphs. In this paper the authors make some theoretical characterizations of multigraphs. Various useful fundamental operations are defined and their properties are studied. A method on how to store the data structure multigraph in computer memory and how to retrieve it back is proposed. A number of propositions are proved on the theory of multigraphs, and it is observed that these properties and theories reduce to those of the graph theory as special cases.

Key Words and Phrases: Multigraph, sub-multigraph, order, null multigraph, edgeless multigraph, indegree, outdegree, degree, adjacent matrix, incidence matrix, edge subtraction, vertex subtraction

2000 Mathematics Subject Classifications: 68R10, 05C99

1. Introduction

Graph theory [1-4,12] has wide applications in several branches of Engineering, Science, Social Science, Medical Science, etc. to list a few only out of many. Graph is also an important non-linear data structure in Computer Science. Multigraph [1,6-9,11,12] is a generalized concept of graph where multiple edges (or arcs) may exist between vertices. Many real life situations of communication network, transportation network, etc. cannot be modeled into graphs, but can be well modeled into multigraphs because of the scope of dealing with multiple edges (or arcs) connecting a pair of nodes. A huge and rich volume of literature is available in the area of 'Graph Theory', but unfortunately the 'Theory of Multigraphs' has not so far developed upto that extent to meet the present requirements to deal with real life network problems.

*Corresponding author.

Since the notion of multigraphs [1,6-9,11,12] is an extension of the notion of graphs, we cannot take it granted that all the rich theories and properties of graphs will be true in case of multigraphs too, unless studied rigorously in the context of multigraphs. There is a genuine need to make algebraic characterization of multigraphs, to define various fundamental operations on multigraphs and then to study the various properties of multigraphs.

In this paper we have done a theoretical work on these issues of multigraphs. Besides that, considering a multigraph as an important non-linear data structure, we have proposed a method on how to store a multigraph in computer memory and how to retrieve it.

2. Preliminaries

In this section we present basic preliminaries about multigraphs from the existing literature [1,6-9,11,12]. A multigraph or pseudograph is like a graph but it is permitted to have multiple edges (also called ”parallel edges”) that have the same end nodes. Thus two vertices may be connected by more than one edge. Some authors also allow multigraphs to have loops, i.e. an edge that connects a vertex to itself, while others call these pseudographs reserving the term multigraph for the case with no loops. *Throughout in our work here, we will work with multigraphs, not pseudographs (i.e. we will not consider the cases of loops).*

Obviously, a classical graph [1-5, 10, 12] is a special case of multigraph where between a pair of vertices there is no multi-edges, only single edge (or no edge). The following diagram (Figure 1) shows a multigraph consisting of four cities A, B, C and D in a country, where the edges denote bus routes and alternative bus-routes from one city to another, with the distances mentioned in miles against each edge.

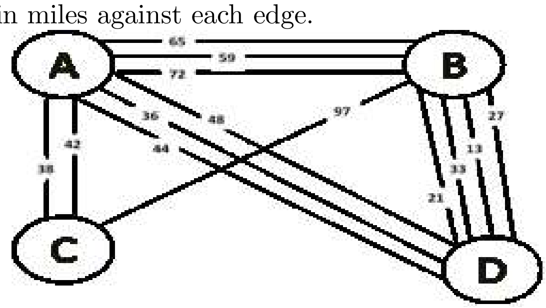

Fig. 1 A multigraph G

2.1. Multigraph

A multigraph G is an ordered pair (V, E) which consists of two sets V and E, where V is the set of vertices (or, nodes), and E is the set of edges (or, arcs).

Here, although multiple edges or arcs might exist between pair of vertices but no loop exists. Multigraphs may be of two types: undirected multigraphs and directed multigraphs. Figure 1 , shown earlier, shows an undirected multigraph and the Figure 2 shows a directed multigraph $G = (V, E)$, where $V = \{A, B, C, D\}$ and $E =$

$\{AB_1, AB_2, BA, AD, AC, CB, BD, DB\}$. In an undirected multigraph the edge (i, j) and the edge (j, i), if exist, are obviously identical unlike in the case of directed multigraph.

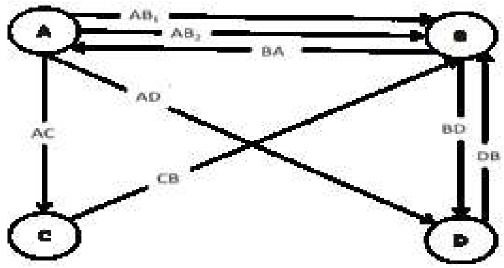

Fig. 2 Multigraph G

As an example, multigraphs might be used to model the possible flight connections offered by an airline. In this case the multigraph would be a directed graph with pairs of directed parallel edges connecting cities to show that it is possible to fly both *to* and *from* these locations.

2.2. Submultigraph

A multigraph $H = (W, F)$ is called a submultigraph of the multigraph G = (V, E) if W $\subseteq$ V and F $\subseteq$ E.
The Figure 3 shows a submultigraph H of the multigraph G of Figure 2.

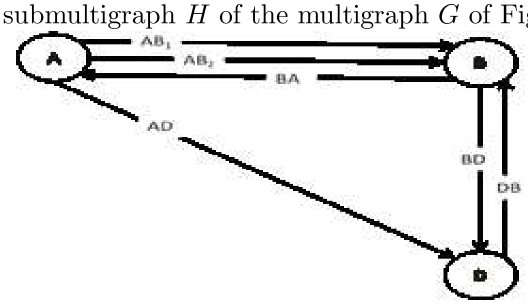

Fig. 3 Submultigraph H

3. Characterization of Multigraphs

It is fact that many of the real life network models are not simple graphs, but multigraphs. A lot of research on technological advancement on Communication Network, Road Transportation System, Airlines Network, etc. are being reported in the journals for which the theory of graph is not an appropriate tool to deal with, but the theory of multigraphs. Unfortunately, there is not much literature available on the theory of multigraphs. In this section we make some useful characterization of the multigraphs.

3.1. Order of a Multigraph

The order of a multigraph $G = (V, E)$ is the cardinality of its vertex set V, and is denoted by $\mathrm{O}(G)$. Thus, order of a multigraph G is a non-negative integer.

3.2. Null Multigraph and Edgeless Multigraph

If $\mathrm{O}(G) = 0$, then the multigraph G is called a 'null multigraph'. Obviously, for a null multigraph G both V and E are null sets. Thus the null multigraph is a trivial submultigraph of every multigraph. However, a multigraph $G = (V, E)$ is called an 'edgeless multigraph' if E is null set. In this sense, a null multigraph is a special case of edgeless multigraphs. Where the null multigraph is an absolutely unique object, but an edgeless multigraph is not. Order of an edgeless multigraph may be 0 or any natural number. Clearly, for an edgeless multigraph it does not matter if we call it a directed or undirected multigraph as it does not have any edge.

3.3. Degree of a Vertex of a non-null Multigraph

In an undirected multigraph, the degree of a vertex is the number of edges adjacent to that vertex. The indegree and outdegree of a vertex can be calculated only in a directed multigraph. The number of arcs incident to a vertex A from the other vertices is called the indegree of the vertex A; and the number of arcs outgoing from a vertex A to the other vertices is called the outdegree of the vertex A. Degree of a vertex in a directed multigraph is the sum total of its indegree and outdegree.

Suppose we want to find out in how many routes a city is connected directly with all other cities. Degree of node for that particular City helps in finding out the number of routes.

3.4. Adjacent edge (arc) set of a vertex

For an undirected multigraph, the collection of all edges of a vertex is called the 'adjacent edge set' of that vertex. For a directed multigraph, the collection of all arcs (incident to or outgoing from) of a vertex is called the 'adjacent arc set' of that vertex.

Example 3.1

Consider the following undirected multigraphs $G_1 = (V_1, E_1)$ as shown below in the Figures 4, where $V_1 = \{A, B, C\}$ and

$$E_1 = \{AB, BC, AC_1, AC_2\}.$$

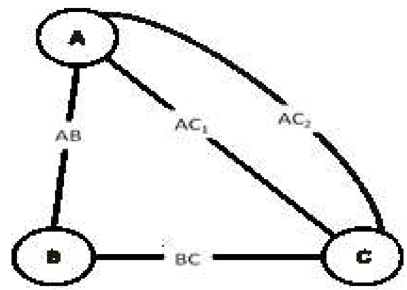

Fig. 4 An undirected multigraph G_1

For the vertex C of this multigraph G_1, the adjacent edge set is $\{AC_1, AC_2, BC\}$.

Example 3.2

Consider the following directed multigraphs $G_2 = (V_2, E_2)$ as shown below in the Figures 5, where $V_2 = \{L, M, N\}$ and

$$E_2 = \{MN_1, LN, LM, ML\}.$$

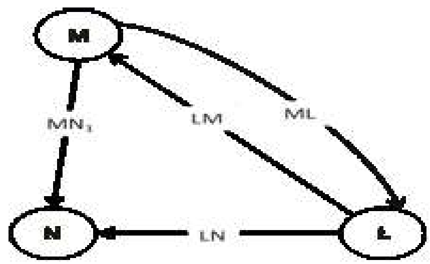

Fig. 5 An undirected multigraph G_2

For the vertex L of this multigraph G_2, the adjacent arc set is $\{LM, ML, LN\}$

There is no justification to say that the results which are true in the theory of graphs will also be true in the theory of multigraphs. We now analyze and prove an important result for multigraphs, analogues to the existing result on graph theory.

Proposition 3.1

Sum of the degrees of the vertices of a multigraph is twice the number of edges (or, arcs) in that multigraph.

Proof :

An edge (or, arc) is identified with two distinct vertices. Thus, when degrees of vertices are counted each edge is counted twice, once for each of the two vertices which are linked with this edge. Therefore, sum of the degrees of the vertices of a multigraph is twice the number of edges in that multigraph. ■

Example 3.3

Let us consider the multigraph G as shown below in Figure 6 below :

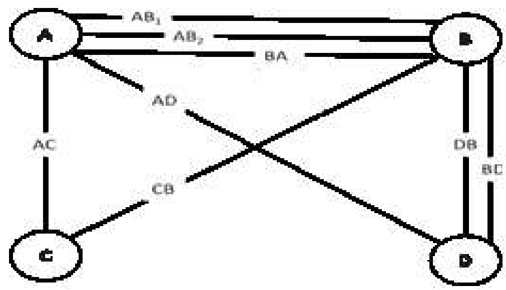

Fig.6 Multigraph G

Here degrees of the vertices A, B, C and $D$are respectively 5, 6, 2, and 3. And the number of edges is 8. Clearly, sum of the degrees of the vertices is twice the number of edges.

Proposition 3.2

Every multigraph has even number of odd vertices.

Proof :

Consider a multigraph G. Suppose that the sum of the degrees of the odd vertices in G is y; and the sum of the degrees of the even vertices in G is x (which is always even).

$\therefore$ Total sum of the degrees of the vertices of the multigraph $G = x + y$

Now, $x + y =$ twice the number of edges (using proposition 3.1)

$=$ even number, z (say)

$\therefore y = z - x =$ even number. ■

Proposition 3.3

In a directed multigraph, the sum of the outdegrees of all the vertices is equal to the number of arcs, which is equal to the sum of all the indegrees of all the vertices.

Proof:

The outdegree of a vertex is the number of arcs adjacent from that vertex. So when we add all the outdegrees, each arc is counted exactly once.

Likewise, when the indegrees are summed up, each arc is counted exactly once. Thus the sum of the outdegrees and the sum of the indegrees are both equal to the total number of arcs in a directed multigraph. ■

Example 3.4

Consider the directed multigraph G as in Figure 2. We see that, indegrees of vertices A, B, C and D are respectively 1, 4, 1 and 2 whose sum is 8. Also we see that outdegrees of vertices A, B, C and $D$are respectively 4, 2, 1, 1 whose sum is 8. And the number of edges is also 8.

Graph is an important non-linear data structure. The multigraphs being the generalized concept of the graphs, are also non-linear data structures. We must have some technique on how to store a multigraph in a computer memory, and how to retrieve a multigraph from the memory. For this, we define adjacency matrix of a multigraph.

3.5 Adjacency Matrix of a non-null Multigraph

The adjacency matrix of a non-null multigraph of order n having the vertex set $V = \{V_1, V_2, V_3, \ldots, V_n\}$ is the
$n \times n$ integer matrix $A = [a_{ij}]$ in which a_{ij} is the number of edges (or, arcs in case of directed multigraph) adjacent from vertex V_i to vertex V_j.

Clearly, $a_{ii} = 0, \forall\ i = 1, 2, \ldots, n$; and also if the multigraph is an undirected multigraph then

$$a_{ij} = a_{ji}, \forall\ i, j = 1, 2, \ldots, n.$$

For any edgeless multigraph, the adjacency matrix is a null square matrix, but for the null multigraph the adjacency matrix is undefined (does not exist).

Example 3.5

Let us consider the directed multigraph G, as shown below (in figure 7) :-

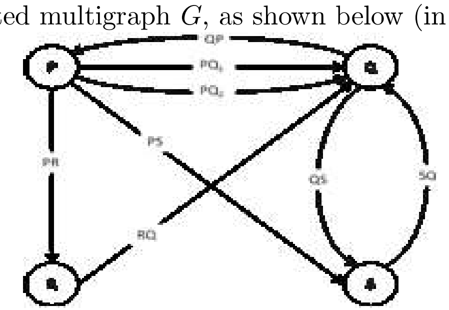

Fig. 7A directed multigraph G

The adjacency matrix of the multigraph G is

$A =$

	P	Q	R	S
P	**0**	**2**	**1**	**1**
Q	**1**	**0**	**0**	**1**
R	**0**	**1**	**0**	**0**
S	**0**	**1**	**0**	**0**

Example 3.6

Consider the undirected multigraph G_1, as in Figure 4, for which the adjacency matrix B is given by

$B =$

	A	B	C
A	**0**	**1**	**2**
B	**1**	**0**	**1**
C	**2**	**1**	**0**

Looking at the adjacency matrix we can immediately find out how many routes are coming in a particular city from another city, or coming out of a city to another city. In the adjacency matrix, a row-head represents the city from which a particular direct route has started, i.e. it is the starting node and a column-head represents the city to which a route would end, i.e. it is the end node.

Thus, by accessing the elements of the adjacency matrix from the memory, it is easy to get the set V and the set E, and hence to retrieve the multigraph itself (an isomorphic multigraph, in fact).

Proposition 3.4 (i)

In the adjacency matrix of an undirected multigraph, the sum of the entries of the row (or column) corresponding to a vertex is its degree, and sum of all the entries of the matrix is twice the sum of the number of edges in the multigraph.

Proof :

Suppose that the vertex set of an undirected multigraph G is $V = \{1, 2, \ldots, n\}$. In its adjacency matrix $A = [a_{ij}]$, the entry a_{ij} is equal to the number of vertices joining vertex i to vertex j. When we add all the entries of the i^{th} row (or i^{th} column) we count the number of edges adjacent to vertex.

$\therefore$ The sum of the i^{th} row (or i^{th} column) is equal to the degree of the vertex i.

When all the entries in the matrix is added, we obtain the sum of the degrees of all vertices, which is twice the sum of the edges. ■

A similar proposition for the directed multigraphs is as below :

Proposition 3.4 (ii)

In the adjacency matrix of a directed multigraph, the sum of the entries in the row corresponding to a vertex is its outdegree; the sum of the entries in the column corresponding to a vertex is its indegree; and sum of all the entries of the matrix is equal to the number of arcs in the multigraph.

Proof: straightforward.

3.5. Incidence Matrix of a non-null Multigraph

Suppose that we want to know whether a given city, say city A, is connected to the city B; and if yes then how the route is, i.e. Is it inward or outward for A? Incidence matrix is helpful in storing such information and data for a multigraph.

Consider a non-null multigraph $G = (V, E)$, where $\#(V) = n$ and $\#(E) = r$. The incidence matrix of the multigraph G is a $n \times r$ matrix $M = [e_{ij}]$ which is defined as below :-

(i) If G is a directed multigraph :

For a given arc, if it originates from a particular vertex we mark the corresponding element of the incidence matrix as 1. If the arc terminates at a particular vertex we mark it as -1 in the matrix. Otherwise, we mark 0 (zero) in the matrix.

(ii) If G is an undirected multigraph :

If an edge exists between any two vertices, we mark the corresponding element of the incidence matrix as 1, else we mark it 0.

Example 3.7

Consider the directed multigraph $G = (V, E)$ of Figure 2. Clearly, the incidence matrix (E) of this multigraph G is :

$E =$

	AB1	AB2	BA	BD	DB	AD	AC	CB
A	**1**	**1**	**-1**	**0**	**0**	**1**	**1**	**0**
B	**-1**	**-1**	**1**	**1**	**-1**	**0**	**0**	**-1**
C	**0**	**0**	**0**	**0**	**0**	**0**	**-1**	**1**
D	**0**	**0**	**0**	**-1**	**1**	**-1**	**0**	**0**

Example 3.8
Consider the undirected multigraph $G_1 = (V_1, E_1)$ as shown in the Figure 4. Clearly, the incidence matrix of G_1 is :

$E =$

	AB	AC1	AC2	BC
A	**1**	**1**	**1**	**0**
B	**1**	**0**	**0**	**1**
C	**0**	**1**	**1**	**1**

To make further characterizations of multigraphs, we like to introduce various fundamental operations on multigraphs which will be useful for fruitful applications of multigraphs in various application fields, in particular in network problems of any system where the notion of these operations will be of genuine needs.

4. Operations on Multigraphs

In this section we define various fundamental operations on multigraphs. First of all we introduce the operations : union and intersection on multigraphs.

4.1. Union of multigraphs

Union of two multigraphs $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$ is the multigraph $G_U = (V_U, E_U)$, denoted by $G_U = G_A \bigcup G_B$, where $V_U = V_A \bigcup V_B$, and $E_U = E_A \bigcup E_B$.

We present below two examples, one showing the union of two undirected multigraphs and the other showing the union of two directed multigraphs.

Example 4.1
Consider the two undirected multigraphs $G_1 = (V_1, E_1)$ as shown in earlier Figure 4 and $G_3 = (V_3, E_3)$ as shown in Figure 8 below, where $V_3 = \{A, C, D, E\}$ and $E_3 = \{AC_1, AC_3, CD_1, CD_2, AD, AE\}$.

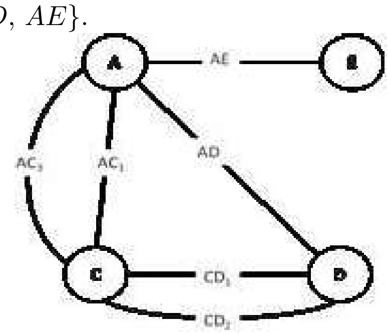

Fig.8 Multigraph G_3

Then $G_1 \bigcup G_3$ will be the undirected multigraph $G = (V, E)$, where $V = \{A, B, C, D, E\}$ and $E = \{AB, BC, AC_1, AC_2, AC_3, CD_1, CD_2, AD, AE\}$ as shown in the Figure 9 below.

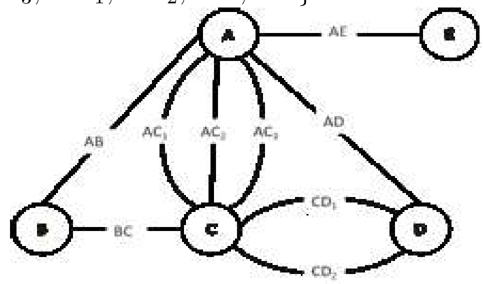

Fig. 9 Multigraph $G = G_1 \bigcup G_3$

Example 4.2

Consider the two directed multigraphs $G_2 = (V_2, E_2)$ as shown in Figure 5 and $G_4 = (V_4, E_4)$ as shown in Figure 10 where $V_4 = \{M, N, P, Q\}$ and $E_4 = \{MQ, MP, MN_1, MN_2, NM, NP\}$.

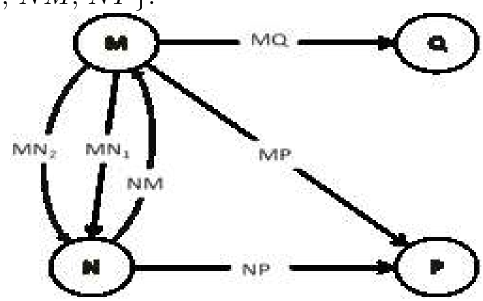

Fig. 10 Multigraph G_4

Then $G_2 \bigcup G_4$ will be the directed multigraph $G = (V, E)$, where $V = \{L, M, N, P, Q\}$ and $E = \{MN_1, LN, LM, ML, MQ, MP, MN_2, NM, NP\}$ as shown in following diagram (Figure 11).

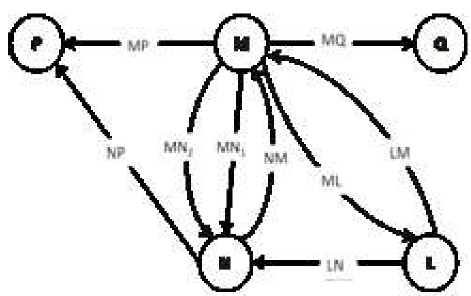

Fig. 11 Multigraph $G = G_2 \bigcup G_4$

4.2. Intersection of two multigraphs

Intersection of two multigraphs $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$ is the multigraph $G_\alpha = (V_\alpha, E_\alpha)$, denoted by
$G_\alpha = G_A \bigcap G_B$, where $V_\alpha = V_A \bigcap V_B$ and $E_\alpha = E_A \bigcap E_B$.

We present below two examples, one showing the intersection of two undirected multigraphs and the other showing the intersection of two directed multigraphs.

Example 4.3
Consider the two undirected multigraphs $G_1 = (V_1, E_1)$ and $G_3 = (V_3, E_3)$, as shown repectively in the Figure 4 and Figure 8 earlier.

Then $G_1 \bigcap G_3$ will be the undirected multigraph $G = (V, E)$, where $V = \{A, C\}$ and $E = \{AC_1\}$ as shown in the Figure 12 below

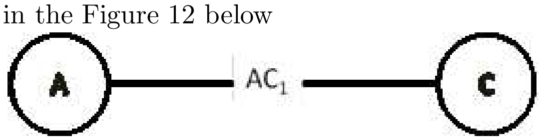

Fig. 12 Multigraph $G = G_1 \bigcap G_3$

Example 4.4
Consider the two directed multigraphs $G_2 = (V_2, E_2)$ and $G_4 = (V_4, E_4)$, as shown respectively in the Figure 5 and Figure 10 earlier.

Then $G_2 \bigcap G_4$ will be the directed multigraph $G = (V, E)$, where $V = \{M, N\}$ and $E = \{MN_1\}$ as shown in the Figure 13 below

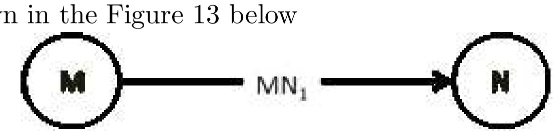

Fig. 13 Multigraph $G = G_2 \bigcap G_4$

It is obvious that both union and intersection operations in multigraphs are commutative as well as associative. i.e. the following results are true in the theory of multigraphs.

Proposition 4.1
If G_1, G_2, G_3 and G are undirected (directed) multigraphs, then

1. $G_1 \bigcup G_2 = G_2 \bigcup G_1$
2. $G_1 \bigcup (G_2 \bigcup G_3) = (G_1 \bigcup G_2) \bigcup G_3$
3. $G_1 \bigcap G_2 = G_2 \bigcap G_1$
4. $G_1 \bigcap (G_2 \bigcap G_3) = (G_1 \bigcap G_2) \bigcap G_3$

5. $G \bigcup G = G$
6. $G \bigcap G = G$
7. $G \bigcup \Phi = G$
8. $G \bigcap \Phi = \Phi$
9. $G \bigcup \emptyset = G$
10. $G \bigcap \emptyset = \emptyset$

where Φ is the null multigraph, and $\emptyset$ is any edgeless submultigraph of G.
The following proposition is straightforward.
Proposition 4.2
For any undirected (directed) multigraphs G_1, G_2 and G_3, the following distributive properties hold true :

1. .$G_1 \bigcup (G_2 \bigcap G_3) = (G_1 \bigcup G_2) \bigcap (G_1 \bigcup G_3)$
2. $G_1 \bigcap (G_2 \bigcup G_3) = (G_1 \bigcap G_2) \bigcup (G_1 \bigcap G_3)$

4.3. Ring sum of Multigraphs

Ring sum of two multigraphs $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$ is the multigraph $G = (V, E)$ denoted by $G = G_A \oplus G_B$, where $V = V_A \bigcup V_B$ and $E = E_A \oplus E_B$ = the set of edges those are either in E_A or in E_B, but not in both.

We present below two examples, one showing the ring sum of two undirected multigraphs and the other showing the ring sum of two directed multigraphs.
Example 4.5
Consider the two undirected multigraphs $G_1 = (V_1, E_1)$ and $G_3 = (V_3, E_3)$, as shown in Figure 4 and Figure 8 earlier.

Then $G_1 \oplus G_3$ will be the undirected multigraph $G = (V, E)$, where $V = \{A, B, C, D, E\}$ and $E = \{AB, BC, AC_2, AC_3, CD_1, CD_2, AD, AE\}$ as shown in the Figure 14.

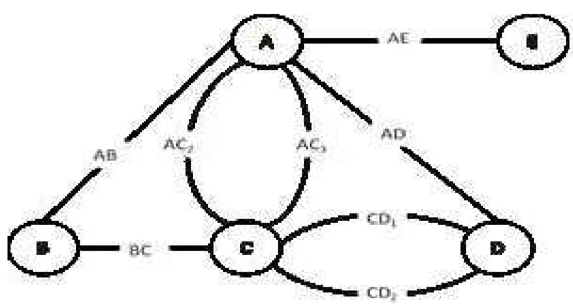

Fig. 14 Multigraph $G = G_1 \oplus G_3$

Example 4.6
Consider the two directed multigraphs $G_2 = (V_2, E_2)$ and $G_4 = (V_4, E_4)$, as shown in Figure 5 and Figure 10 earlier.

Then $G_2 \oplus G_4$ will be the directed multigraph $G = (V, E)$, where $V = \{L, M, N, P, Q\}$ and $E = \{LN, LM, ML, MQ, MP, MN_2, NM, NP\}$ as shown in Figure 15.

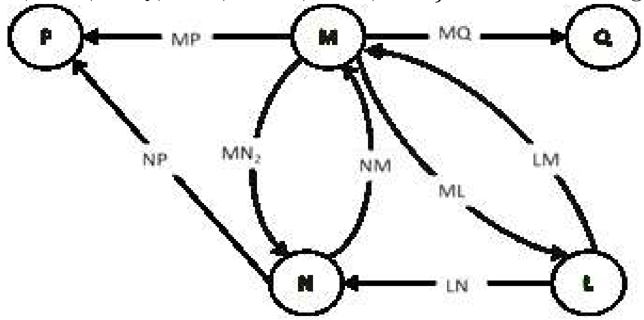

Fig. 15 Multigraph $G = G_2 \oplus G_4$

The following results hold good in multigraphs for the operation of ring sum.

Proposition 4.3
If G_1, G_2, G_3 and G are undirected (directed) multigraphs and Φ is the null multigraph, then

1. $G_1 \oplus G_2 = G_2 \oplus G_1$
2. $G \oplus G =$ The edgeless multigraph (V, Φ) where V is the edge-set of G.
3. $G \oplus \Phi = G$

4.4. Insertion Operation in Multigraphs

Insertion is one of the most useful operations in network problems. Consider a situation in which a new vertex has to be added in a multigraph network model.

The new vertex is to be incorporated with connectivities (inward or outward or both ways) with some of the existing vertices of the multigraph or maybe with no connectivity with any vertices (i.e. in an isolated way). Thus insertion of a node happens along with its adjacent edge set (or, adjacent arc set), not independently.

If v_i is a new vertex, then $G + v_i$ denotes a multigraph G_A obtained by inserting (or adding) v_i to multigraph G,
defined by $G_A = (V_A, E_A)$ where $V_A = V \bigcup \{v_i\}$ and $E_A = E \bigcup E_{v_i}$, where E_{v_i} is the adjacent edge set (or, adjacent arc set) of v_i.

Example 4.7
Consider the undirected multigraph $G_3 = (V_3, E_3)$ as shown earlier in Figure 8.

Suppose that a new vertex K is to be inserted to the multigraph G_3, along with its adjacent edge set $E_K = \{AK, EK_1, EK_2, KE, KD, DK\}$, which contains all the new edges by which K has links can be connected to the other vertices of the multigraph G_3.

Clearly, the resultant multigraph G shown in following Figure 16, will be given by $G = G_3 + K\,(V,\,E)$, where
$V = V_3 \bigcup \{K\}$ and $E = E_3 \bigcup E_K = \{AC_1,\,AC_3,\,CD_1,\,CD_2,\,AD,\,AE,\,AK,\,EK_1,\,EK_2,\,KD\}$.

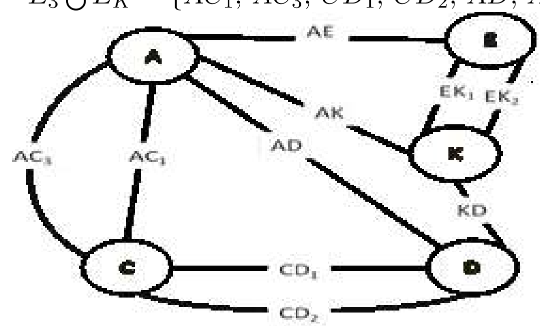

Fig. 16 Multigraph $G = G_3 + K$

Example 4.8
Consider the directed multigraph $G_4 = (V_4,\,E_4)$ as shown earlier in Figure 10

Suppose that we want to insert a new vertex K to the multigraph G_4, along with its adjacent arc set $E_K = \{KM,\,QK,\,KQ,\,PK\}$, which contains all the new arcs by which K can be connected to the other vertices of multigraph G_4.

Clearly, the resultant multigraph G, shown below in Figure 17, will be given by $G = G_4 + K = (V,\,E)$, where
$V = V_4 \bigcup \{K\}$ and $E = E_4 \bigcup E_K = \{AC_1,\,AC_3,\,CD_1,\,CD_2,\,AD,\,AE,\,AK,\,EK_1,\,EK_2,\,KD\}$.

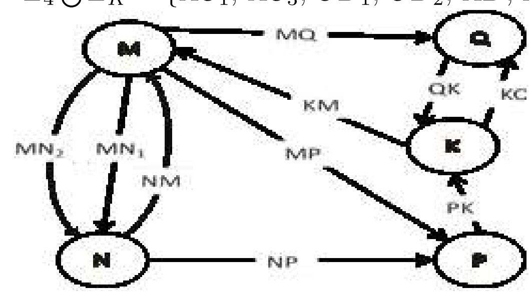

Fig. 17 Multigraph $G = G_4 + K$

4.5. Deletion Operation in Multigraphs

Deletion is an important operation in network problems. Sometimes it happens that some nodes or some links are temporarily or permanently damaged in a network. Then for efficient and best alternative solution of the problems pertaining to that network, we need to consider the rest part only, instead of the whole part of the network.

There are two kinds of deletion operations in multigraphs, which are :
(a) Deletion of a vertex in a multigraph, and
(b) Deletion of an edge (arc) in a multigraph.

4.5.1. Deletion of an edge (or arc) in a multigraph

If v_i is a vertex in a multigraph G, then $G - v_i$ denotes a submultigraph obtained by deleting (or removing) v_i from G. Thus, deletion of a vertex always implies the deletion of all the edges (arcs) which are linked (incident or outgoing) with that vertex.

The notation **" - "** is called 'vertex subtraction' operation.
Thus, $G - v_i = (V - \{v_i\}\,,\, E - E_{v_i})$, where v_i is the vertex to be deleted and E_{v_i} is the set of all the edges (or arcs) adjacent to the vertex v_i.

We present below two examples, one showing deletion of a vertex in an undirected multigraph and the other showing deletion of a vertex in a directed multigraph.

Example 4.9
Consider the undirected multigraph $G_3 = (V_3,\, E_3)$ as shown earlier in Figure 8.

Let us delete the vertex A from the multigraph G_3 and find out the resultant multigraph G (Figure 18).
Clearly, $G = G_3 - A = (V, E)$, where $V = \{C, D, E\}$ and $E = E_3 - E_A = \{AC_1, AC_3, CD_1, CD_2, AD, AE\} - \{AC_1, AC_3, AD, AE\} = \{CD_1, CD_2\}$.

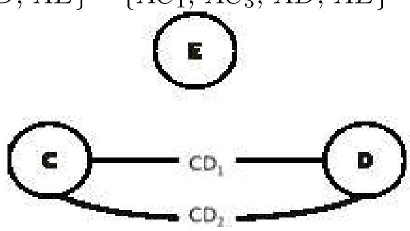

Fig. 18 Multigraph $G = G_3 - A$

Example 4.10
Consider the directed multigraph $G_4 = (V_4,\, E_4)$ as shown earlier in Figure 10 earlier.

Now let us delete the vertex M from the multigraph G_4 and find out the resultant multigraph G (Figure 19).
Clearly, $G = G_4 - M = (V, E)$, where $V = \{N, P, Q\}$ and $E = E_4 - E_M = \{MQ, MP, MN_1, MN_2, NM, NP\} - \{MQ, MP, MN_1, MN_2, NM\} = \{NP\}$.

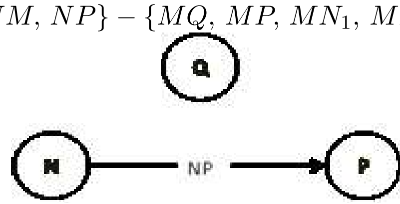

Fig. 19 Multigraph $G = G_4 - M$

4.5.2. Deletion of an edge (or arc) in a multigraph

If e_i is an edge (or arc) in a multigraph G, then $G \sim e_i$ denotes the submultigraph obtained by deleting (or removing) e_i from G. Deletion of an edge (or arc) does not affect the vertices which it connects.

The notation " ˜ " is called 'edge subtraction' (or, 'arc subtraction') operation. Thus, $G \sim e_i = (V, E - \{e_i\})$.

We present below two examples, one showing deletion of an edge in an undirected multigraph and the other showing deletion of an arc in a directed multigraph.

Example 4.11

Consider the undirected multigraph $G_3 = (V_3, E_3)$ as shown earlier in Figure 8.

Now let us delete the edge AE from the multigraph G_3 and find out the resultant multigraph G (as shown in
Figure 20). Clearly, $G = G_3 \sim AE = (V, E)$, where $V = V_3$ and $E = E_3 - \{AE\} = \{AC_1, AC_3, CD_1, CD_2, AD\}$.

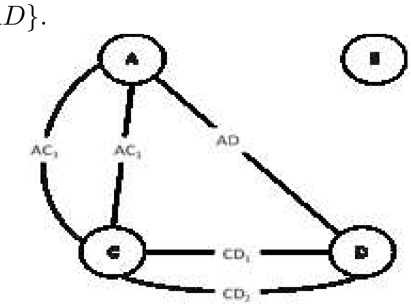

Fig. 20 Multigraph $G = G_3 \sim AE$

Example 4.12

Consider the directed multigraph $G_4 = (V_4, E_4)$ as shown earlier in Figure 10, where $V_4 = \{M, N, P, Q\}$ and

$$E_4 = \{MQ, MP, MN_1, MN_2, NM, NP\}.$$

Now let us delete the edge MQ from the multigraph G_4 and find out the resultant multigraph G (Figure 21).
Clearly, $G = G_4 \sim MQ = (V, E)$, where $V = V_4$ and $E = E_4 - \{MQ\} = \{MP, MN_1, MN_2, NM, NP\}$.

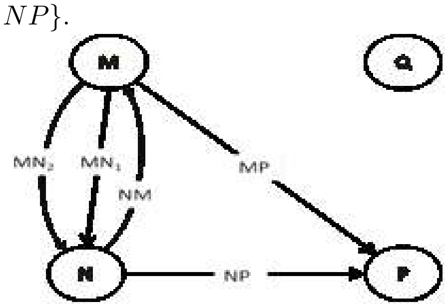

Fig. 21 Multigraph $G = G_4 \sim MQ$

Proposition 4.4

If $G = (V, E)$ is a multigraph, then $G \sim e_j = G \oplus (V, \{e_j\})$, where e_j is an edge of G.

Proof :
Suppose that $G \oplus (V, \{e_j\}) = (V_R, E_R)$.
Clearly, $V_R = V \bigcup V = V$, and
E_R =Set of edges that are either in E or in $\{e_j\}$, but not in both. $= \{E \bigcup e_j\} - \{E \bigcap e_j\} = E - \{e_j\}$

$$\therefore\ G \oplus (V, \{e_j\}) = G \sim e_j. \blacksquare$$

Proposition 4.5
Consider the multigraph $G = (V, E)$ where E is a non null set of cardinality n. Let $F = \{f_1, f_2, f_3, \ldots, f_r\}$ be a non-null subset of E (i.e., $r \leq n$).

Then, $(G \sim f_i) \sim f_j = (G \sim f_j) \sim f_i \ \forall\, i, j = 1, 2, 3. \ldots, r$ where $i \neq j$.
Proof : $(G \sim f_i) \sim f_j = (V, (E - \{f_i\})) \sim f_j$

$$= (V, (E - \{f_i\} - \{f_j\})$$

$$= (V, (E - \{f_i, f_j\}))$$

$$= (V, (E - \{f_j, f_i\}))$$

= (G ˜ fj) ˜ fi ■
The above result shows that in the expression $(G \sim f_i) \sim f_j$, the order of the appearances of f_i and f_j is not significant.
Definition 4.5
Consider the multigraph $G = (V, E)$ where E is a non null set of cardinality n. Let $F = \{f_1, f_2, f_3, \ldots, f_r\}$ be a non-null subset of E (i.e., $r \leq n$). Then, for the set F of edges, the edge-subtraction $G \sim F$ is defined by

$$G \sim F = ((\ldots\ldots((G \sim f_1) \sim f_2) \sim \ldots\ldots) \sim f_r)\,,$$

where the order of appearances of $f_1, f_2, \ldots, f_r$ is not significant.

The following results are now straightforward.
Proposition 4.6
If $G = (V, K)$ is a multigraph and F, K are two disjoint subsets of E, then

1. $G \sim E = (V, \Phi)$, an edgeless multigraph.
2. $G \sim \Phi = G$
3. $(G \sim F) \sim K = (G \sim K) \sim F$, i.e. the order of appearances of F and K is not significant here.

5. Conclusion

The objective philosophy behind this work is that since the notion of graphs is a particular instance of the notion of multigraphs, the existing rich literature on the theory of graphs cannot be granted to be automatically valid in the theory of multigraphs, unless studied rigorously in the context of multigraphs. In this paper, we have made a theoretical study on various important properties of multigraphs by making a number of useful characterizations of multigraphs. Some fundamental operations on multigraphs are defined and explained with examples. A number of useful propositions are proved, which reduce to those of graph theory as special cases just. Since multigraph is a non-linear data structure, a method is proposed on how to store multigraphs in computer memory and how to retrieve them back. The complete work of this paper is of theoretical nature and of fundamental requirements for any good application/exercise of multigraphs in the real life problems of different branches of Engineering, Information Technology, Science, Social Science, Medical Science, etc. to list a few only.

References

[1] Balakrishnan, V. K., Graph Theory, McGraw-Hill; 1997.

[2] Bollobas, Bela., Modern Graph Theory, Springer; 2002.

[3] Diestel, Reinhard., Graph Theory, Springer 2000.

[4] Harary, Frank., Graph Theory, Addison Wesley Publishing Company, 1995.

[5] Jenson P, Barnes J., Network Flow Programming, John Wiley and Sons, New York. 1980.

[6] J. Ivanco, Decompositions of multigraphs into parts with the same size, Discussiones Mathematicae Graph Theory 30(2)(2010), 335-347.

[7] J. Ivanco, M. Meszka, Z. Skupien, Decompositions of multigraphs into parts with two edges, Discussiones Mathematicae Graph Theory 22(1)(2002),113-121.

[8] K. Brys, M. Kouider, Z. Lonc, M. Maheo, Decomposition of multigraphs, Discussiones Mathematicae Graph Theory 18(2)(1998), 225-232.

[9] Mariusz Meszka and Zdzislaw Skupien, Decomposition of a Complete Multigraph into Almost Arbitrary Paths, Discussiones Mathematicae Graph Theory 32(2)(2012) 357–372.

[10] Okada, S. and T. Soper., A Shortest Path Problem on a Network with Fuzzy Arc Lengths, Fuzzy Sets and Systems 109 (2000), 129–140.

[11] Zdzislaw Skupien, On Distance edge Coloring of a Cyclic Multigraph, Discussiones Mathematicae Graph Theory 19(1999) 251–252.

[12] http://en.wikipedia.org/wiki/Multigraph and http://en.wikipedia.org/wiki/Graph_(mathematics).

Siddhartha Sankar Biswas
Department of Computer Engineering, Faculty of Engineering & Technology, Jamia Millia Islamia, New Delhi – 110025, India
E-mail: ssbiswas1984@gmail.com

Bashir Alam
Department of Computer Engineering, Faculty of Engineering & Technology, Jamia Millia Islamia, New Delhi – 110025, India

M. N. Doja
Department of Computer Engineering, Faculty of Engineering & Technology, Jamia Millia Islamia, New Delhi – 110025, India