JSc EUSL (2021), vol. 12, no. 1, p 13-21

Print- ISSN: 1391-586X, E-ISSN: 2602-9030

RESEARCH ARTICLE

Properties of Adjacency Matrix of a Graph and It's Construction

Paramadevan, P*., and Sotheeswaran, S.

Department of Mathematics, Eastern University, Sri Lanka

ABSTRACT: This research paper studies about the adjacency matrix of a graph, as it is a fundamental matrix associated with any graph. This study about the properties of adjacency matrix and associated theorems is inevitable since the graph is stored in a computer in terms of its adjacency matrix.

Keywords: Graph, Adjacency Matrix.

DOI: http://doi.org/10.4038/jsc.v12i1.29

1. INTRODUCTION

In the field of discrete mathematics, graph theory simply means the study of Graphs [1]. Conceptually, a graph is formed by vertices and edges connecting its vertices. Formally, a graph G is a pair of sets (V, E), where V is the set of vertices and E is the set of edges, formed by pair of vertices. E is a multi-set, in other words, it's elements can occur more than once so that every element has a multiplicity.

Graph theory is applied academically in various occasions in the field of Physics, Chemistry etc., and mainly in the field of engineering it is applied in road maps, when constructing schemes and drawings, constellations, etc. In scheduling problems also graph theory plays a major role. This research paper studies the adjacency matrix associated with a graph. Since the graph is stored in a computer as its adjacency matrix, the study of adjacency matrix is inevitable when dealing with graphs in a computer.

*Corresponding author: *paramadevanp@esn.ac.lk*

https://orchid.org/0000-0002-6367-2483

This paper is organized in such a way that, Section II is to illustrate some examples to understand about the adjacency matrix of different types of graphs, such as ordinary graph, directed graph and weighted directed graph, and list some properties of the adjacency matrix discussed in [2]. Section III provides some important theorems in which the notion of adjacency matrix is involved. Finally, a computer programme is given in Section IV, to construct the adjacency matrix of different types of graphs discussed in Section II.

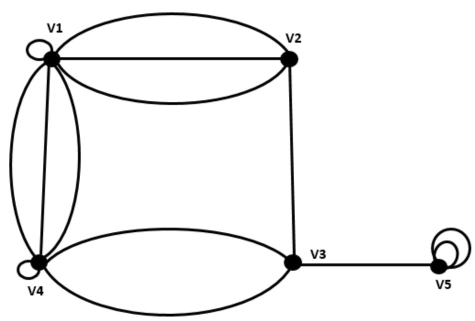
2. THEORY AND METHODS

2.1 Adjacency Matrix

Definition: The adjacency matrix of a graph $G = (V, E)$ is an $n \times n$ matrix $A_G = (a_{ij})$, where $V = \{v_1, v_2, \cdots, v_n\}$ is the vertex set, E is the edge set of G and a_{ij} is the number of edges between the vertices v_i and v_j. In the adjacency matrix of a directed graph, a_{ij} equals the number of arcs from the vertex v_i to v_j.

Example 1: Graph G_1

The adjacency matrix A_{G_1} of the graph G_1 is given in Figure 1.

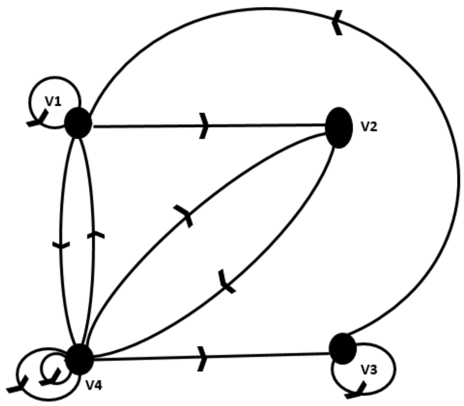


$$\begin{pmatrix} 1 & 3 & 0 & 3 & 0 \\ 3 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 2 & 1 \\ 3 & 0 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 2 \end{pmatrix}$$

Figure 1: An undirected graph G_1 and it's adjacency matrix A_{G_1}.

Example 2: Graph G_2

The adjacency matrix A_{G_2} of the directed graph G_2 is given in Figure 2.

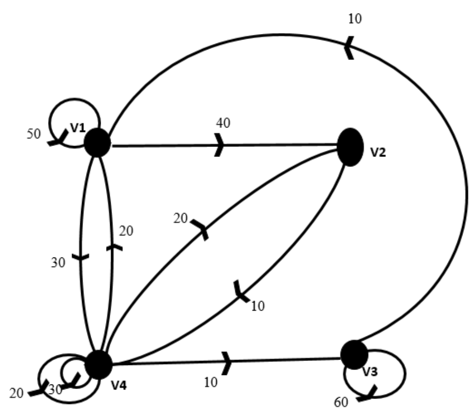


$$\begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 2 \end{pmatrix}$$

Figure 2: A directed graph G_2 and it's adjacency matrix A_{G_2}.

Example 3: Graph G_3

The adjacency matrix A_{G_3} of the weighted directed graph G_3 is given in Figure 3.



$$\begin{pmatrix} 50 & 40 & 0 & 30 \\ 0 & 0 & 0 & 10 \\ 10 & 0 & 60 & 0 \\ 20 & 20 & 10 & 50 \end{pmatrix}$$

Figure 3: A weighted directed graph G_3 and it's adjacency matrix A_{G_3}.

2.2 Properties of Adjacency matrix

1. It is symmetric in non-directed graph and not always symmetric in directed graph.

2. In the absence of loops, entries in the principal diagonal are all zero.

3. Number of non-zero diagonal entries in the main diagonal is equal to the number of loops in the graph.

4. It can be partitioned as $\left(\begin{array}{c|c} A_{G_1} & O_1 \\ \hline O_2 & A_{G_2} \end{array}\right)$, when it has components G_1 and G_2, where A_{G_1} and A_{G_2} are the adjacency matrices of G_1 and G_2, respectively.

5. Interchange of two rows (in this case corresponding columns should be interchanged) is equivalent to relabeling of corresponding vertices.

6. Two graphs G_1 and G_2, with no parallel edges, are isomorphic if and only if $A_{G_2} = PA_{G_1}P^{-1}$, where A_{G_1} and A_{G_2} are adjacency matrices of G_1 and G_2, respectively, and P is the matrix obtained as in (5).

7. The $(i,j)^{th}$ entry of A_G^m is equal to the number of walks of length m from v_i to v_j, and the least non-zero m satisfying the non-zero $(i,j)^{th}$ entry is equal to the distance from v_i to v_j.

8. Degree of v_i is equal to the sum of the off diagonal entries plus twice the diagonal entries.

9. For any given square, symmetric and binary matrix A of order n, there exists a graph G with n vertices without parallel edges whose adjacency matrix is A.

3. RESULTS

3.1 Theorems

Theorem 1: Let A_G be the adjacency matrix of a graph $G = (V, E)$, then the number of different paths of length l between the i^{th} and j^{th} vertices of G is equal to the $(i,j)^{th}$ entry of A_G^l.

Proof: We shall use mathematical induction on l to prove this theorem. Result is trivial for $l = 0$ and $l = 1$.

Let $l = 2$.

Since the adjacency matrix is symmetric, it's square also a symmetric matrix. Note that the $(i,j)^{th}$ entry of A_G^2 is equal to the number of places in which both i^{th} and j^{th} rows of A_G have ones, hence which is equal to the number of vertices those are adjacent to both i^{th} and j^{th} vertices. Thus, the $(i,j)^{th}$ entry of A_G^2 is equal to the number of different paths of length 2 between i^{th} and j^{th} vertices.

While the $(i,i)^{th}$ entry of A_G^2 is equal to the number of $1'$s in the i^{th} row, and hence equal to the degree of the i^{th} vertex. Hence, the result is true for $l=2$.

Now assume the result is true for p.

Note that the $(i,j)^{th}$ entry, $(A_G^{p+1})_{ij}$ of A_G^{p+1} is equal to $\sum_{r=1}^{n}(A_G^p)_{ir}\ (A_G)_{rj}$.

Every path with length $(p+1)$ from the vertex v_i to v_j consists of paths from v_i to v_t of length p together an edge v_tv_j. Since there are $(A_G^p)_{ir}$ number of such paths of length p and a_{rj} such edges for each vertex v_r, the total number of paths of length $(p+1)$ from the vertex v_i to v_j is equal to $\sum_{r=1}^{n}(A_G^p)_{ir}\ (A_G)_{rj}$.

Theorem 2: Let A_G be the adjacency matrix of a graph $G=(V,E)$, where $V=\{v_1,v_2,\cdots,v_n\}$ and let $B(=b_{ij})=A_G+A_G^2+A_G^3+\cdots+A_G^{n-1}$. Then, A_G is connected if and only if $b_{ij}\neq 0$, when $i\neq j$.

Proof: When we apply the Theorem 1 for the matrix B, b_{ij} = number of different paths of length 1 from the vertex i to j + number of different paths of length 2 from the vertex i to j $+\cdots+$ number of different paths of length $(n-1)$ from the vertex i to j. That is,

b_{ij} = number of different paths of length less than n from the vertex i to j.

Since G is connected, then there is a path from any vertex i to any other the vertex j. Since G has only n vertices, length of these paths less than n, hence $b_{ij}\neq 0$, when $j\neq j$.

Conversely, if $b_{ij}\neq 0$, then there exist at least one path from v_i to v_j, thus G is connected.

Theorem 3: Let A_G be the adjacency matrix of a connected graph $G=(V,E)$. The distance between two vertices v_i and v_j is equal to l if and only if l is the smallest integer such that $(A_G^l)_{ij}\neq 0$.

Proof: Since $d(v_i,v_j)=l$, there is no path from v_i to v_j of length less than l.

Then, $(A_G)_{ij}=0,(A_G^2)_{ij}=0,\cdots,(A_G^{l-1})_{ij}=0$, and $(A_G^l)_{ij}\neq 0$.

That is, l is the smallest integer such that $(A_G^l)_{ij}\neq 0$.

Conversely, suppose that l is the smallest integer such that $(A_G^l)_{ij}\neq 0$. Therefore, there is no paths from v_i to v_j of length less than l. Thus, the length of the shortest path (distance) between v_i and v_j is l.

Before moving to Theorem 4, let's see three associated definitions.

Definition 1: The all-vertex incidence matrix of a non-empty loop-less graph $G = (V, E)$ is an $m \times n$ matrix $M = (m_{ij})$, where n is the number of vertices and m is the number of edges in G, and

$$m_{ij} = \begin{cases} 1, & \text{if } v_i \text{ is an end vertex of } e_j; \\ 0, & \text{otherwise.} \end{cases}$$

Definition 2: Another form of an incidence matrix

The matrix $F = (f_{ij})$ of the graph $G = (V, E)$ with $V = \{v_1, v_2, \cdots, v_n\}$ and $E = \{e_1, e_2, \cdots, e_m\}$ is the $n \times m$ matrix associated with the chosen orientation of the edges of G in which for each $e = (v_i, v_j)$, one of v_i or v_j is taken as positive end and the other as negative end, and is defined by

$$f_{ij} = \begin{cases} 1, & \text{if } v_i \text{ is the positive end of } e_j; \\ -1, & \text{if } v_i \text{ is the negative end of } e_j; \\ 0, & \text{if } v_i \text{ is not incident with } e_j. \end{cases}$$

Definition 3: Let G be a simple graph and let d_i be the degree of the vertex v_i in G. The degree matrix $H = (h_{ij})$ of G is defined by

$$h_{ij} = \begin{cases} d_i, & \text{for } i = j; \\ 0, & \text{otherwise.} \end{cases}$$

Theorem 4: Let F be the modified incidence matrix, A be the adjacency matrix and H be the degree matrix of a graph G. Then $FF^T = H - A$.

Proof: The $(i, j)^{th}$ entry of FF^T, $[FF^T]_{ij} = \sum_{r=1}^{m} [F]_{ir}[F^T]_{rj} = \sum_{r=1}^{m} [F]_{ir}[F]_{jr}$.

We have $[F]_{ir}$ and $[F]_{jr}$ are non-zero if and only if the edge $e_r = (v_i, v_j)$ exists in G.

Then, for $i \neq j$, $\sum_{r=1}^{m} [F]_{ir}[F]_{jr} = \begin{cases} -1, & \text{if } e_r = (v_i, v_j); \\ 0, & \text{if } e_r \text{ is not an edge in G.} \end{cases}$

If $i = j$, $[F]_{ir}[F]_{jr} = 1$ whenever $[F]_{ik} = \pm 1$, and this occurs d_i times corresponding to the number of edges incident on v_i. Thus, $\sum_{r=1}^{m} [F]_{ir}[F]_{jr} = d_i$ for $i = j$.

Therefore, $[FF^T]_{ij} = \begin{cases} -1 \text{ or } 0, & \text{according to whether, for } i \neq j,\ v_iv_j \text{ is an edge or not;} \\ d_i, & \text{for } i = j. \end{cases}$

Further, $[H - A]_{ij} = [H]_{ij} - [A]_{ij}$

$$= \begin{cases} d_i - 0, & \text{for } i = j; \\ 0 - (1 \text{ or } 0), & \text{according to, for } i \neq j,\ (v_i, v_j) \text{ is an edge or not.} \end{cases}$$

$$= \begin{cases} d_i, & \text{for } i = j; \\ -1 \text{ or } 0, & \text{according to, for } i \neq j,\ (v_i, v_j) \text{ is an edge or not.} \end{cases}$$

Hence, $FF^T = H - A$.

3.2. Computer programs to construct the adjacency matrix of a graph:

3.2.1 MATLAB code for constructing the adjacency matrix of a **undirected graph**

```
vertex=[1,2,3,4,5];
edges=[1,1;1,2;1,2;1,4;1,4;1,4;2,3;3,4;3,4;
       4,4;3,5;5,5;5,5];
disp("The edges are given below:");
disp(edges);
edgessize=size(edges);
noofvertex=size(vertex);
e=edges;
adj=zeros(noofvertex(1,2),noofvertex(1,2));
for j=1:noofvertex(1,2)
    k=find(e==j);
    edgesize=size(e);
    ksize=size(k);
    for i=1:ksize(1,1)
        if k(i)/(edgesize(1,1)+1)<1
            adj(e(k(i),1),e(k(i),2))++;
            if e(k(i),1)~=e(k(i),2)
                adj(e(k(i),2),e(k(i),1))++;
            end
        else
            k1= mod(k(i),edgesize(1,1));
            if k1==0
                k1=edgesize(1,1);
            end
            adj(e(k1,1),e(k1,2))++;
            if e(k1,1)~=e(k1,2)
               adj(e(k1,2),e(k1,1))++;
            end
        end
    end
end
disp("This is the adjacency matrix for the
             given edges:");
disp(adj/2);
```

3.2.2 MATLAB code for constructing the adjacency matrix of a **directed graph**

```
vertex=[1,2,3,4];
edges=[1,1;1,2;2,4;4,2;1,4;4,1;4,3;3,3;3,1;4,4;4,4];
disp("The edges are given below:");
disp(edges);
edgessize=size(edges);
noofvertex=size(vertex);
e=edges;
adj=zeros(noofvertex(1,2),noofvertex(1,2));
for j=1:noofvertex(1,2)
    k=find(e==j);
    edgesize=size(e);
    ksize=size(k);
    for i=1:ksize(1,1)
        if k(i)/(edgesize(1,1)+1)<1
            adj(e(k(i),1),e(k(i),2))++;
        else
            k1= mod(k(i),edgesize(1,1));
            if k1==0
                k1=edgesize(1,1);
            end
            adj(e(k1,1),e(k1,2))++;
        end
    end
end
disp("This is the adjacency matrix for the given edges:");
disp(adj/2);
```

3.3.3 MATLAB code for constructing the adjacency matrix of a **weighted directed graph**

```
vertex=[1,2,3,4];
edges=[1,1,50;1,2,40;2,4,10;4,2,20;1,4,20;4,1,30;4,3,10;
       3,3,60;3,1,10;4,4,30;4,4,20];
disp("The weighted edges are given below:");
disp(edges);
edgessize=size(edges);
noofvertex=size(vertex);
e=edges;
adj=zeros(noofvertex(1,2),noofvertex(1,2));
for j=1:noofvertex(1,2)
    k=find(e==j);
    edgesize=size(e);
    ksize=size(k);
    for i=1:ksize(1,1)
        if k(i)/(edgesize(1,1)+1)<1
            adj(e(k(i),1),e(k(i),2))=adj(e(k(i),1),e(k(i),2))
                                      +e(k(i),3);
        else
            k1= mod(k(i),edgesize(1,1));
            if k1==0
                k1=edgesize(1,1);
            end
            adj(e(k1,1),e(k1,2))=adj(e(k1,1),e(k1,2))+e(k1,3);
        end
    end
end
disp("The following weighted adjacency matrix is for the
                                        given edges:");
disp(adj/2);
```

CONCLUSION

In Section II of this research paper, it is discussed about adjacency matrix of different types of graphs, namely undirected graph, directed graph, and weighted directed graph and its properties. In Section III, three important theorems related to the adjacency matrix are discussed with their proofs. In Section IV, MATLAB computer programs to develop adjacency matrix of the above three types of graphs are presented. Since MATLAB is a commonly using tool, this work would be useful for the researchers who are working on adjacency matrix of graphs.

REFERENCE

1. DIESTEL, R.(2005). Graph Theory. 5^{th} edition, Springer-Verlag Berlin Heidelberg

2. Chiranjib, M and Gyan, M (2014). Role of Adjacency Matrix in Graph Theory, IOSR Journal Computer Engineering, 16(2) 58-63.