Available online at http://scik.org
J. Math. Comput. Sci. 6 (2016), No. 2, 272-280
ISSN: 1927-5307

A SIMPLE ALGORITHM TO FIND MAXIMUM MATCHING FOR COMPLETE GRAPH AND COMPLETE BIPARTITE GRAPH: USING INCIDENCE MATRIX APPROACH

ISHWAR BAIDARI*, VIJAYKUMAR GURANI

Dept. of Computer Science, KU Dharwad-580003, India

Abstract: A simple graph *G(V,E)* has vertex set *V* and edge set *E* then matching means a subset *S* of the edge set *E* such that no two edges of *S* are adjacent in *E*. If *S* is matching, the two end points of each edge of *S* are said to be matched under *S*, and each vertex incident with an edge of *S* is said to be covered by *S*. The matching *S* with maximum number of edges is called maximum matching. In this paper we present a polynomial time algorithm to find maximum matching for complete graph and complete bipartite graph using incidence matrix approach.

Keywords: Maximum matching; Edge Independent Set; Incidence Matrix.

2010 AMS Subject Classification: 05C85.

1. Introduction

Since computers can process graph effectively in form of matrix rather than picture format it has been standard practice to specify the graph in matrix form either adjacency matrix or incidence matrix. In this work, graph is represented in the form of incidence matrix. Consider undirected graph G, G has vertex set V and edge set E then edge independent set is a subset S of the edge set E such that the edges in set S does not have common vertex in graph G. The edge independent set of G is also called as matching of G. A matching is called as maximum when the subset S covers maximum vertices of G and the cardinality of maximum matching is denoted by α'(G). A matching is called as perfect matching if subset S covers all the vertices of G [1]. A covering of a graph G is a subset of K of V such that every edge of G has at least one end in K. A covering K* is minimum covering if G has no covering K with |K|<|K*|. The number of vertices in minimum covering is called as covering number of G denoted by β(G). Given vertex v the number of edges incident on the vertex v in graph G is called Degree of the graph denoted by $d_G(v)$. The maximum degree of the graph G is denoted by Δ(G) and minimum degree denoted by δ(G) [1].

The graph G can be represented as incidence matrix which is denoted by A(G)[4]. Let G be a graph with n vertices, m edges and no self loops. The incidence matrix representation of a graph G consists of $n \times m$ matrix $A(G) = [a_{ij}]$ where n rows are n vertices and m columns are m edges

The matrix is

$$a_{ij} = \begin{cases} 1 & if\ j^{th}\ edge\ m_j\ is\ incident\ on\ the\ i^{th}\ vertex; \\ 0 & Otherwise \end{cases}$$

*Corresponding author

Received September 23, 2015

Example:

Consider the graph G given below

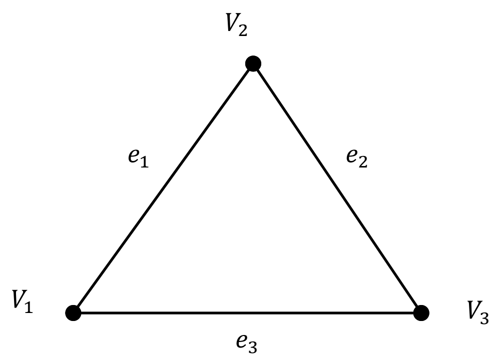

Figure 1: Simple graph G

The incidence matrix A (G) is

$$A(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \end{array} \begin{array}{c} \begin{array}{ccc} e_1 & e_2 & e_3 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \end{array}$$

The matrix is $n \times m = 3 \times 3$ i.e. $n = 3\ vertices, m = 3\ edges$

From the incidence matrix it can be observed that if there exists an edge between 2 vertices then the edge column contains exactly two 1's. If there is no edge then the edge column contains all 0's without 1's .

2. Problem Statement and preliminaries:

Definition: A Complete graph is simple graph G with n vertices that has edge to every other vertex (n-1) denoted by K_n. Let M be a matching in graph G. An alternating path or cycle in G is alternate edges from S and E \ M. An M-alternating path might or might not start or end with edges of M. If neither start nor end of path is covered by M then such path is called as M-augmenting path. The following theorem can be found in [1]

Theorem 1[1]**:** (BERGE'S)

A matching M in graph G is a maximum matching if and only if G contains no augmenting path.

Proof: Let M be a matching of G and G contains M-augmenting path P. Then M′:=M Δ E(P) is matching in G and |M′|:=|M| + 1

Thus M is not a maximum matching.

Conversely, suppose that M is not a maximum matching, and let M* be a maximum matching in G, so that |M*|>|M|. Set H := G[MΔM*]. Each vertex of H has a degree one or two in H, for it can be incident with at most one edge of M and one edge M*. Consequently, each component of H is either an even cycle with edges alternately in M and M*, or else a path with edges alternately in M and M*.

Because |M*|>|M|, the sub graph H contains more edges of M* than of M, and therefore some path-component P of H must start and end with edges of M*. The path P is thus an M-augmenting path in G.

Definition: A graph is bipartite if its vertex set can be partitioned into two subsets X and Y such that every edge has one vertex in set X and other vertex in set Y, such partition (X,Y) is bipartition which is denoted by G[X,Y]. A complete bipartite graph is bipartite graph G[X,Y] where every vertex in set X has edge connected to every vertex of set Y denoted by $K_{n,m}$.

Theorem 2 [1]**:** (HALL'S)

A bipartite graph G := G[X,Y] has a matching which covers every vertex in X if and only if |N(S)|≥|S| for all S⊆X

Proof: Let G:=G[X,Y] be a bipartite graph which has a matching M covering every vertex in X. Consider a subset S of X. The vertices in S are matched under M with distinct vertices in N(S). Therefore |N(S)|≥|S|.

Conversely, let G:=G[X,Y] be a bipartite graph which has no matching covering every vertex in X. Let M* be a maximum matching in G and u a vertex by M*-alternating path. Because M* is a maximum matching than u is only vertex in Z not covered by M*. Set R := X∩Z and B:=Y∩Z. Set Z is vertices reachable from u by M*-alternating paths.

Clearly the vertices of R\{u} are matched under M* with the vertices of B. Therefore |B|=|R|-1 and N(R)⊇B. In fact N(R) = B because every vertex in N(R) is connected to u by an M*-alternating path. Thus |N(R)|=|B|=|R|-1 and Hall's condition fails for the set S := R.

Corollary: A bipartite graph G[X,Y] has a perfect matching if and only if |X|=|Y| and |N(S)| ≥|S| for all S $\subseteq X$

Proposition: Let M be a matching and K a covering such that |M|=|K| then M is a maximum matching and K is minimum covering.

i.e. α'(G)=β(G)

In the following we consider complete graph and complete bipartite graph G with n vertices and m edges (i.e. $n \times m$) to extract the set of edges from graph G such that it constitutes for maximum matching. Since matrix representation of graph is one way to represent the graph G and set operations like union and intersection can be applied easily to the graph matrix. Taking the incidence matrix A(G) as the input to extract the maximum matching, the algorithmic logic of set operations on incidence matrix will lead to the output of maximum matching α'(G).

Graph $G = matrix\ A(G)$ with n vertices and m edges, represents $n\ (rows) \times\ m\ (columns)$.

3. Algorithm to find maximum matching

In this section we discuss about algorithm to find edge independent set (EIS) taking incidence matrix $a[n][m]$ as input and applying the basic set operations to extract the α'(G). We consider set variable EIS to store vertex pairs to represent the existence of edge between the two vertices. Initially EIS is set to NULL and at the end of procedure execution EIS consists of edge set which constitute for maximum matching.

$|V(G)|$ is vertex cardinality of graph $G, |E(G)|$ is the edge cardinality in graph G. Since any single edge other than self loop is held by 2 vertices, algorithm uses two variables $vertex1, vertex2$ to represent edge during algorithm execution.

Algorithm:

1. Input graph G as Incidence matrix $A(G)$
2. $A(G) = a[n][m]$
3. $n = number\ of\ vertices, m = number\ edges$
4. $EIS \leftarrow \{\emptyset\}$, flag=0;
5. $if(|V(G)| <= 1)$
6. $return$;
7. $for\ j = 1\ to\ m\{$
8. $for\ i = 1\ to\ n \quad \{$
9. $if(a[i][j] == 1)$
10. $if(flag == 0)$
11. $Vertex1 = i, flag = 1;$
12. $else$
13. $Vertex2 = i$; break;
14. $\}$
15. $if(\ (EIS \ \cap vertex1) == \ \emptyset\ and\ (EIS \ \cap vertex2) == \emptyset\ ) \qquad then$
16. $EIS \leftarrow EIS \ \cup \{vertex1, vertex2\}$
17. $flag = 0;$
18. }
19. $output\ \ EIS$

4. Working of algorithm

The procedure edge independent set can be explained as follows:

In a given procedure, line 1-3 takes the graph G input in the form of incidence matrix. In line 4 the EIS is initialized to null set. Line 5-6 checks for given graph G if there exists at least one edge. If the graph G contains more than one vertex the procedure continues further, else terminates the procedure. Line 7 reads the incidence matrix horizontally from the left to right. Line 8 reads the incidence matrix vertically from top to down. Line 9-13 checks the existence of edge between the vertices if there exists an edge then two vertices of graph are assigned to variable $vertex1$ and $vertex2$. When $vertex2$ is assigned i value the inner for loop breaks. This means that vertex pair corresponding to edge is read from the matrix and further reading of matrix has no significance.

We consider flag to distinguish between first vertex and second vertex from incidence matrix. Line 15-17 checks non-existence of edge $(vertex1, vertex2)$ in EIS. If EIS already contains edge (vertex pair) which has one of the vertexes as common then such edge is discarded from adding in to set EIS. If EIS has no common vertex from the vertex pair $(vertex1, vertex2)$ then that edge is added to the set EIS. The flag is reset every time when next column is read from incidence matrix. After all edges are read from the incidence matrix the edge set in variable EIS contains edges which constitutes to edge independent set.

5. Performance analysis of Algorithm;

Referring to above given algorithm, line 1-6 is executed only once and its time complexity is O(1). Line 7-8 has nested *for* loop where outer loop is executed m times and inner loop n times therefore complexity is O(n*m). Line 9-13 is covered by inner *for* loop and gets executed once for each iteration with time complexity O(1). Line 15-17is covered by outer *for* loop and gets executed with time complexity O(n). Line 19 prints the results of variable EIS with complexity O(1). Thus for proposed algorithm the total complexity is $O(n \times m)$; to generalize it's going to be $O(n^2)$

6. Algorithm applied for class of graph:

In this section we apply the proposed algorithm on complete graph and complete bipartite graph without parallel edges and loop-less graph. The result of algorithm execution is stored in set variable EIS.

Theorem 3: For complete graph K_n maximum matching cardinality $|\alpha'(G)| = \left\lfloor \frac{n}{2} \right\rfloor$.

First Proof: Let G is complete graph with n vertices denoted by K_n, then edge cardinality of K_n is

$$|E(K_n)| = (n-1) + (n-2) + (n-3) \ldots (n-n) = n(n-1)/2$$

In complete graph, n^{th} vertex has edges to (n-1) vertices. Cardinality of edges incident at n^{th} vertex is (n-1). For n^{th} and $(n-1)^{th}$ vertex the cardinality of edges is (n-1)+(n-2). Similarly for n^{th}, $(n-1)^{th}$, $(n-2)^{th}$ vertex it is (n-1)+(n-2)+(n-3) and so on. Therefore the edge cardinality of complete graph K_n is $|E(K_n)| = \sum_{i=1}^{n}(n-i)$

For simple loop-less graph to have single edge at least 2 vertices are required. In complete graph every vertex is adjacent to each other and if single edge is selected from K_n to matching set S then $|E(K_n)| = \sum_{i=1}^{n-2}(n-i)$

Every edge need 2 vertices and therefore for n vertices the maximum edge independent set is n/2 if n is even and $\left\lfloor \frac{n}{2} \right\rfloor$ if n is odd. To generalize maximum matching for n vertices is $|\alpha'(G)| = \left\lfloor \frac{n}{2} \right\rfloor$

Second proof: consider $|\alpha'(G)| = \left\lfloor \frac{n}{2} \right\rfloor$ → Eq-1, from [2] we have

$|\alpha(G)| + |\beta(G)| = n$ → Eq-2;

$|\alpha'(G)| + |\beta'(G)| = n$ → Eq-3;

$therefore\ |\alpha'(G)| = n - |\beta'(G)|;$ →Eq-4

From Eq-1 $|\alpha'(G)| = \frac{n}{2};\ \ 2|\alpha'(G)| = n;$

$2(n - |\beta'(G)|) = n\ \ (from\ Eq - 4);$

$2n - 2|\beta'(G)| = n;\ \ 2n - n = 2|\beta'(G)|$

$n = 2|\beta'(G)|$ →Eq-5

Add Eq-5 in Eq-3

$|\alpha'(G)| + |\beta'(G)| = 2|\beta'(G)|;\quad |\alpha'(G)| = 2|\beta'(G)| - |\beta'(G)|;$

$|\alpha'(G)| = |\beta'(G)|$ → Eq-6

Adding Eq-6 in Eq-3

$|\alpha'(G)| + |\beta'(G)| = n;$

$|\alpha'(G)| + |\alpha'(G)| = n;$

$2|\alpha'(G)| = n;$

Therefore $|\alpha'(G)| = n/2$ for even vertices and $\left\lfloor\frac{n}{2}\right\rfloor$ for odd vertices

Complete graph K_4

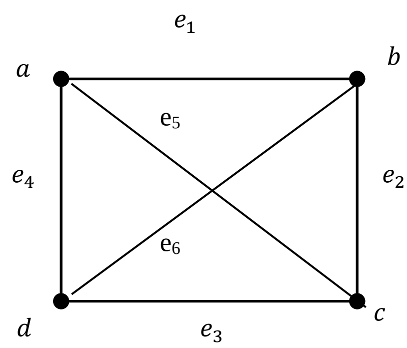

Figure 2: complete graph

Incidence matrix $A(K_4) =$

E / V	e_1	e_2	e_3	e_4	e_5	e_6
a	1	0	0	1	1	0
b	1	1	0	0	0	1
c	0	1	1	0	1	0
d	0	0	1	1	0	1

V(G) vertex set and E(G) is edge set of complete graph [1] Maximum matching generated from proposed algorithm for given complete graph K_4 EIS = { e_1, e_3}; $|\alpha'(G)|=2$

Theorem 4: For complete bipartite graph $K_{n,m}$ if n<m then $|\alpha'(G)|= n$

First Proof: Consider a complete bipartite graph G[X,Y] where |X|<|Y| and every vertex of X is having edge to every vertex of Y. By induction method consider

Basis step if |X|=1 then $|\alpha'(G)|= 1$ and if |X|=2 then $|\alpha'(G)|= 2$

Inductive hypothesis: if |X|=n then $|\alpha'(G)|= n$

Then for |X|=n+1 we have $|X|=|\alpha'(G)|+1$ and |X|=|X|+1 (by inductive hypothesis)

Therefore for $K_{n,m}$ if n<m then $|\alpha'(G)|= n$

Second proof: From Theorem 3 we know that any edge needs two vertices. In bipartite graph there exits an edge such that one vertex is in set X and other vertex in set Y. Therefore for $K_{n,m}$ with n<m it is possible to have n vertices as part of matching set but for m-n vertices present in set Y we don't have vertices in set X to include in matching set. Therefore $|\alpha'(G)|= n$.

Lemma 1: Given a complete graph the number of 1's in vertex row $v \in V$ of incidence matrix = $d_G(v)$

Proof: By the property of incidence matrix each row represents the vertices of graph and column represents the edges incident on these vertices if the edge column contains 1 to corresponding vertex. And hence the number of 1's in row of vertex v$\in V$gives the total edges incident on that vertex. Therefore the degree of vertex v= $d_G(v)$, is the number of 1's in that row. In same of complete graph all vertex has same degree i.e. every row of incidence matrix has same number of 1's.

Lemma 2: Given a complete graph sum of entries in each column of incidence matrix are equal.

Proof: referring to lemma 1 proof it's known that every column has same number of 1s and therefore result of adding all 1's are same in every column of the incidence matrix.

Lemma 3: If G is complete graph then its induced sub graph H is also complete i.e. clique

Proof: For given complete G with n vertices denoted by K_n has edges to all the n-1 vertices. Induced subgraph H of G with m vertices denoted by H_m has edges to m-1 vertices because by the concept of induced subgraph if m vertices are present in H then all the edges incident on vertices of H must be present therefore the lemma holds.

Corollary: If H is induced sub graph of complete graph G then $|\alpha'(H)| \leq |\alpha'(G)|$

Complete bipartite graph $K_{2,3}$

G[X, Y]

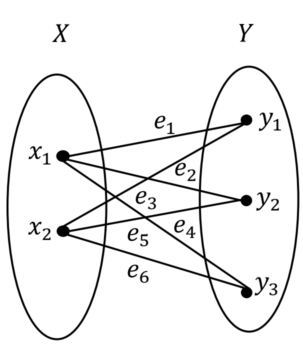

Figure 3: complete bipartite graph

Incidence matrix $A(K_{2,3}) =$

E / V	e_1	e_2	e_3	e_4	e_5	e_6
x_1	1	1	1	0	0	0
y_1	1	0	0	1	0	0
y_2	0	1	0	0	1	0
y_3	0	0	1	0	0	1
x_2	0	0	0	1	1	1

For given bipartite graph |X|=2 and |Y|=3, Maximum matching for given complete bipartite graph $K_{2,3}$ EIS = { e_1, e_5}; $|\alpha''(G)| = 2$.

Lemma 5: Given complete bipartite graph G as incidence matrix then Δ(G) = row with maximum number of 1's and δ(G)= row with minimum number of 1's

Proof: lemma 5 proof is similar to lemma1; here degree is not same for all vertices, the number of 1's in matrix row gives the edges incident on particular vertex of that row. In case of bipartite graph the minimum number of 1's in row represents the minimum degree of graph and maximum number of 1's in row represents the maximum degree of graph.

Lemma 6: If G is complete bipartite graph $K_{n,m}$ where n=m then α'(G) is perfect matching

Proof: By theorem4 it is known that for complete bipartite graph $K_{n,m}$ when n<m then $|\alpha'(G)|=n$. If n=m then every vertex in set X is matched with every vertex in set Y therefore the matching set has all vertex of graph being covered therefore α'(G) is perfect matching.

7. Theorem 5 (correctness of algorithm)

Let G be complete graph or complete bipartite graph and A(G) is incidence matrix for the graph. When proposed algorithm is run on G then algorithm reads the incidence matrix column wise and extracts the edges in the form of vertex pairs to set variable EIS. When the procedure is terminated the set variable EIS contains edges which is maximum edge independent set

Proof: Algorithm uses set variable EIS which is initialized to NULL and flag variable to help in assigning the vertex pair to temporary variables vertex1 and vertex2. Algorithm selects vertex from graph and adds the pair to EIS if either of vertex is not present in EIS. Algorithm terminates once the entire matrix column is being read. After termination of algorithm set EIS contains maximum edge independent set.

Assume that variable EIS does not contain maximum edge independent set then from theorem1 we should have augmenting path in graph G, if G is complete graph then proposed algorithm selects vertex pair such that either of vertex is not part of set EIS. Since in complete graph every vertex has edge of every other vertex the start vertex and end vertex should be part of set EIS or the neighbor vertex should be part of EIS and therefore there is no augmenting path. Set EIS has maximum edge independent set. From theorem 2 and 4 it is proved that EIS has maximum edge independent set for complete bipartite graph G

8. Conclusion

From the proposed algorithm we have proved that with simple set operations on incidence matrix it is possible to find maximum edge independent set in polynomial time for complete graph and complete bipartite graph without constructing the graphical representation of graph. Computing task on graph in matrix form is simple. It would be interesting to extend this work to any graph. It is also interesting to explore the graph property by considering the graph in matrix form.

Conflict of Interests

The authors declare that there is no conflict of interests.

Acknowledgments

Authors would like to thank referees for the valuable comments.

REFERENCES

[1] J.A Bondy and U.S.R. Murty, Graduate text in Mathematics- Graph theory, Springer, 2008, 414-449.

[2] R. Balakrishnan, K. Ranganathan, A textbook of Graph Theory, 2nd edition, Springer, 2012, 99-102.

[3] Thomas H cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein, Introduction to Algorithms, 3rd edition, PHI, 2009.

[4] Narsingh Deo, Graph theory with applications to Engineering and computer science, PHI, 1999, 137-138.

[5] A. Yayimli, Graph Theory and Applications, 2007.