Applications of Bipartite Graph in diverse fields including cloud computing

[1]Arunkumar B R, [2]Komala R

Prof. and Head, Dept. of MCA, BMSIT, Bengaluru-560064 and Ph.D. Research supervisor, VTU RRC, Belagavi
Asst. Professor, Dept. of MCA, Sir MVIT, Bengaluru and Ph.D. Research Scholar,VTU RRC, Belagavi

ABSTRACT: *Graph theory finds its enormous applications in various diverse fields. Its applications are evolving as it is perfect natural model and able to solve the problems in a unique way.Several disciplines even though speak about graph theory that is only in wider context. This paper pinpoints the applications of Bipartite graph in diverse field with a more points stressed on cloud computing.*
KEY WORDS: *Graph theory, Bipartite graph cloud computing, perfect matching applications*

I. INTRODUCTION

Graph theory has emerged as most approachable for all most problems in any field. In recent years, graph theory has emerged as one of the most sociable and fruitful methods for analyzing chemical reaction networks (CRNs). Graph theory can deal with models for which other techniques fail, for example, models where there is incomplete information about the parameters or that are of high dimension. Models with such issues are common in CRN theory [16]. Graph theory can find its applications in all most all disciplines of science, engineering, technology and including medical fields.

Both in the view point of theory and practical bipartite graphs are perhaps the most basic of entities in graph theory. Until now, several graph theory structures including Bipartite graph have been considered only as a special class in some wider context in the discipline such as chemistry and computer science [1]. This effort deals exclusively with bipartite graphs and its applications in cloud computing.

It appeared recently that the classical random graph model used to represent real-world complex networks does not capture their main properties. It is showed in paper [17] that any complex network can be modeled using bipartite graph with some specifications. It also implies that you have always have got alternate graph theory structure to replace some structure which is not suitable that is also from graph theory. The next subsection brings the bipartite graph with its definition, uniqueness and characteristics.

1.1 Bipartite Graph

In the mathematical field of graph theory, a **bipartite graph** (or **bigraph**) is a graph whose vertices can be divided into two disjoint sets U and V (that is, U and V are each independent sets such that every edge connects a vertex in U to one in V. Vertex set U and V are often denoted as partite sets. Equivalently, a bipartite graph is a graph that does not contain any odd-length cycles [1], [2].

The two sets U and V may be thought of as a coloring of the graph with two colors: if one colors all nodes in U blue, and all nodes in V green, each edge has endpoints of differing colors, as is required in the graph coloring problem.[3][4] In contrast, such a coloring is impossible in the case of a non-bipartite graph, such as a triangle after one node is colored blue and another green, the third vertex of the triangle is connected to vertices of both colors, preventing it from being assigned either color.

One often writes G=(U,V,E)to denote a bipartite graph whose partition has the parts U and V, with Edenoting the edges of the graph. If a bipartite graph is not connected, it may have more than one bipartition; in this case, the (U,V,E) notation is helpful in specifying one particular bipartition that may be of importance in an application. If $|U| = |V|$, that is, if the two subsets have equal cardinality, then Gis called a balanced bipartite graph.[3]. If all vertices on the same side of the bipartition have the same degree, then Gis called bi-regular. The given item to be searched in cloud can be modeled as a bipartite cloud, further, perfect matching algorithms, theorems and lemmas can obviously mathematically modeled and analyzed.

A bipartite graph G = (U; V;E) is specified by two disjoint sets U and V of vertices, and a set E of edges between them. A perfect matching is a subset of the edge set E such that every vertex has exactly one edge incident on it. Since we are interested in perfect matching's in the graph G, we shall assume that $|U| = |V| = n$. Let U = {u1; u2; _ _ _ ; un} and V = {v1; v2; _ _ _ ; v$_n$}. The algorithm has no error if G does not have a

---

perfect matching (no instance), and errors with probability at most ½ if G does have a perfect matching (yes instance).
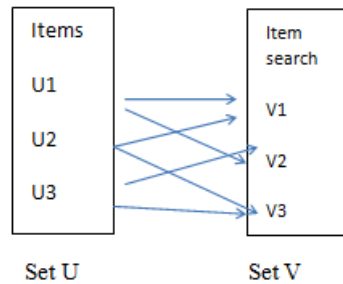

Figure 3: Bipartite representation.

The two disjoint sets U and V where Set U contains data to be searched and Set V data is stored. The match done with these two disjoint set to match with exact data searched in Set V.

1.2 Perfect matching counting in Bipartite Cloud
In mathematical graph theory, a matching or independent edge set in a graph is a set of edges without common vertices. It may also be an entire graph consisting of edges without common vertices.

1.2.1 Properties
In any graph without isolated vertices, the sum of the matching number and the edge covering number equals the number of vertices. If there is a perfect matching, then both the matching number and the edge cover number are |V| / 2.
If A and B are two maximal matchings, then $|A| \leq 2|B|$ and $|B| \leq 2|A|$. To see this, observe that each edge in B \ A can be adjacent to at most two edges in A \ B because A is a matching; moreover each edge in A \ B is adjacent to an edge in B \ A by maximality of B, hence

$$\left| A \backslash B \right| \leq \left| B \backslash A \right|$$

Further, we get that

$$\left| A \right| = \left| A \cap B \right| + A \backslash B \leq 2 \left| B \cap A \right| + 2 \left| B \backslash A \right| = 2 \left| B \right|$$

In particular, this shows that any maximal matching is a 2-approximation of a maximum matching and also a 2-approximation of a minimum maximal matching. This inequality is tight: for example, if G is a path with 3 edges and 4 nodes, the size of a minimum maximal matching is 1 and the size of a maximum matching is 2.

1.3 Maximal matchings
A maximal matching can be found with a simple greedy algorithm. A maximum matching is also a maximal matching, and hence it is possible to find a largest maximal matching in polynomial time. However, no polynomial-time algorithm is known for finding a **minimum maximal matching**, that is, a maximal matching that contains the smallest possible number of edges.
Note that a maximal matching with k edges is an edge dominating set with k edges. Conversely, if we are given a minimum edge dominating set with k edges, we can construct a maximal matching with k edges in polynomial time. Therefore the problem of finding a minimum maximal matching is essentially equal to the problem of finding a minimum edge dominating set.[9] Both of these two optimization problems are known to be NP-hard the decision versions of these problems are classical examples of NP-Complete problems.[10]Both problems can be approximated within factor 2 in polynomial time: simply find an arbitrary maximal matching M.[11]

1.3.1Counting problems
The number of matchings in a graph is known as the Hosoya index of the graph. It is #P-complete to compute this quantity. It remains #P-complete in the special case of counting the number of perfect matchings in a given dipartite graph , because computing the permanent of an arbitrary 0–1 matrix (another #P-complete problem) is the same as computing the number of perfect matchings in the bipartite graph having the given matrix as its biadjacency matrix. However, there exists a fully polynomial time randomized approximation scheme for counting the number of bipartite matchings.[12].A remarkable theorem of Kasteleyn states that the number of perfect matchings in a planar graph can be computed exactly in polynomial time via the FKT algorithm.

The number of perfect matching in a complete graph K_n (with n even) is given by the double factorial $(n − 1)!!$.[13]The numbers of matchings in complete graphs, without constraining the matchings to be perfect, are given by the telephone numbers.[14]

Applications

A **Kekulé structure** of an aromatic compound consists of a perfect matching of its carbon skeleton, showing the locations of double bonds in the chemical structure. These structures are named after Friedrich August Kekule von Stradonitz, who showed that benzene (in graph theoretical terms, a 6-vertex cycle) can be given such a structure.[19]

The Hosoya index is the number of non-empty matchings plus one; it is used in computational chemistry and mathematical chemistry investigations for organic compounds.

1.4 Some important characterization of bipartite graphs are as follows [4][5]:
- A graph is bipartite if and only if it does not contain an odd cycle.
- A graph is bipartite if and only if it is 2-colorable, (i.e. its chromatic number is less than or equal to 2).
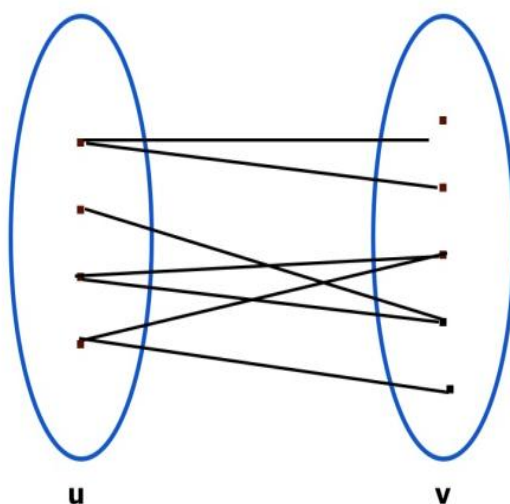- The spectrum of a graph is symmetric if and only if it's a bipartite graph.

Figure 2: Bipartite Graph

1.5 Some types of Bipartite Graph and example

A complete bipartite graph is a graph G whose vertex set V can be partitioned into two non emptysetsV_1 and V_2 in such a way that every vertex in V_1 is adjacent to every vertex in, no vertex in V_1 is adjacent to a vertex in V_1, and no vertex in V_2 is adjacent to a vertex in V_2. If V_1 has r vertices and V_2 has s vertices then the complete bipartite graph is written as $K_{r, s}$The complete bipartite graph $K_{1,3}$ is called the claw graph.
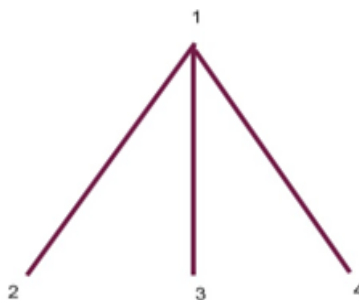

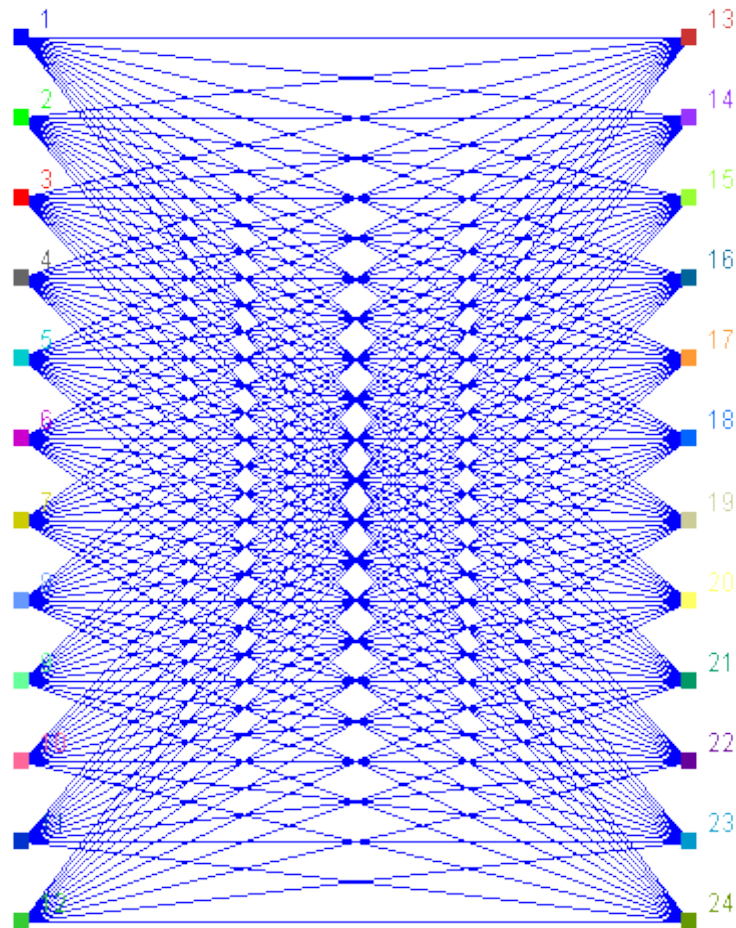**Figure 3.** [15] The claw graph $K_{1,3}$

Figure 4. [15] The complete bipartite graph $K_{12,12}$

1. Applications of Bipartite Graph

Bipartite graphs have a wonderful property that their vertices can be divided into two parts such that no two vertices which are in same part are joined by an edge[1].This property can lead to several applications of bipartite graph. The first systematic investigation on Bi-graph was started by Konig (1914). Bi-graph are natural mathematical models which can represent a practical context where two different types of objects interact such as jobs and workers, telephone exchanges and cities.

For instance, a graph of football players and clubs, with an edge between a player and a club if the player has played for that club, is a natural example of an affiliation network, a type of bipartite graph used in social network analysis.

Bipartite graphs are extensively used in modern coding theory, especially to code words received from the channel. Factor graphs and Tanner graphs are examples of this. A Tanner graph is a bipartite graph in which the vertices on one side of the bipartition represent digits of a codeword, and the vertices on the other side represent combinations of digits that are expected to sum to zero in a codeword without errors. A factor graph is a closely related belief network used for probabilistic decoding of LDPC and turbo codes.

Bipartite matching   can be used to solve a problem to determine which couples are compatible for marriage (man/woman), example: (Rama, Sita),(Shiva,Parvati).   Problem is to maximize the number matches,neither polygamy nor polyandry are allowed. Each vertex can be assigned only one vertex of the other group. This problem can be solved by solve this problem by creating a flow network out of a bipartite graph.
Bipartite graph can be used in the medical field in the detection of lung cancer, throat cancer etc
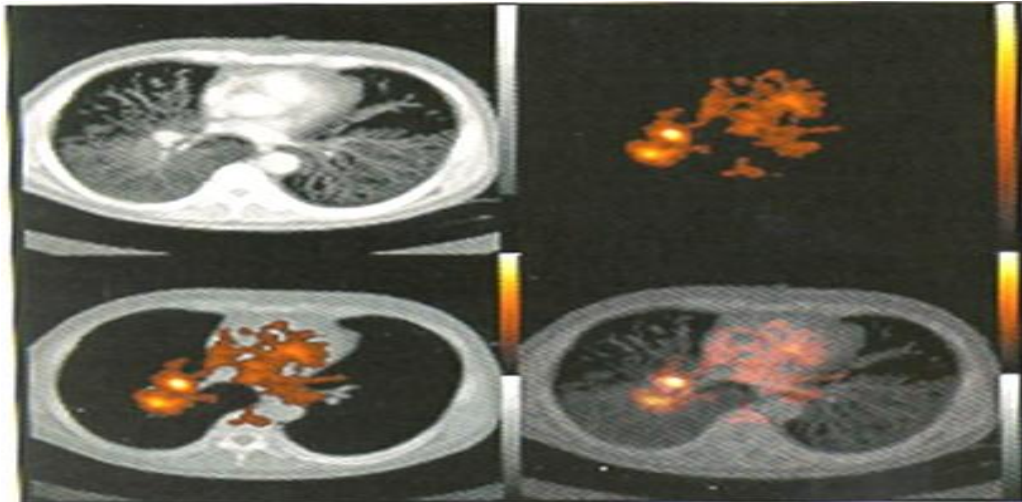
Figure5. Shows a PET image of the thorax [**Sources: Levitin and many other CSE, Marmara Univ. May/2009**]

Bipartite graph can mathematical model the common situations as well as serious problems in the several areas of wireless networks including cognitive radio networks, big data and cloud computing. Therefore, one can understand that it is desire of the researcher to apply the graph theory structures such as bi-graph in their works. Theoretical bi-graphs appears to be very simple but with all the complexities of the general principles of graph theory embodiment in it. This work has done an extensive literature analysis and found that bi-graphs are not much discussed and applied much in the specific context of cloud computing.

Cloud computing is an Internet-based computing [2]. It relies on sharing computing resources which are delivered as services on the Internet. Web service is one of the most important types of services that can be used in cloud computing. But many of them may be similar in some functional or nonfunctional properties, making how to recommend a suitable web service a problem facing many developers. Researchers have taken the QoS attributes into consideration. However, their research is on the premise that all the recommended web services are compatible, i.e., the recommended web services can be composed with existing web services. It may not always be true. In the paper [2],the compatibility of web services is taken into consideration, and presented a Bipartite Graph based Service Recommendation (BIGSIR) method to address the service compatibility problem.

The work in [2], BIGSIR uses the historical usage data of web services to recommend web services to developers. Different from existing web service recommendation approaches, BIGSIR adopts a bipartite graph to visual the web services and the relationship between them. Based on the graph model, an effective recommendation algorithm is introduced to recommend the suitable web services. The approach in [2] is evaluated on a dataset constructed from Experiment, a search engine that contains about 1,851 web services and 2,000 workflows. Experimental results demonstrate that apart from some isolated web services or workflows, BIGSIR can obtain promising results. This work not only provides a new dataset, but also highlights a new perspective for service recommendation, i.e. services as a bipartite network.

The work in [6], discusses on the implementation of—Map Reduce software which is a framework for processing data-intensive applications with a parallel manner in cloud computing systems. There are also an increasing number of Map Reduce jobs that require deadline guarantees. The existing deadline-concerning scheduling schemes do not consider the two problems in the Map Reduce computing environment: slot performance heterogeneity and job time variation. In this paper, the utilize the Bipartite Graph modeling to propose a new Map Reduce Scheduler called the BGMRS. The BGMRS can obtain the optimal solution of the deadline-constrained scheduling problem by transforming the problem into a well-known graph problem: minimum weighted bipartite matching.

1.6 Bipartite Graphs and Perfect Matchings
    In bipartite graph, the nodes are divided into two categories and each edge connects a node in one category to a node in the other category. For example: Administrators of a college dormitory are assigning rooms to students, each room for single student. Students and rooms are categories.

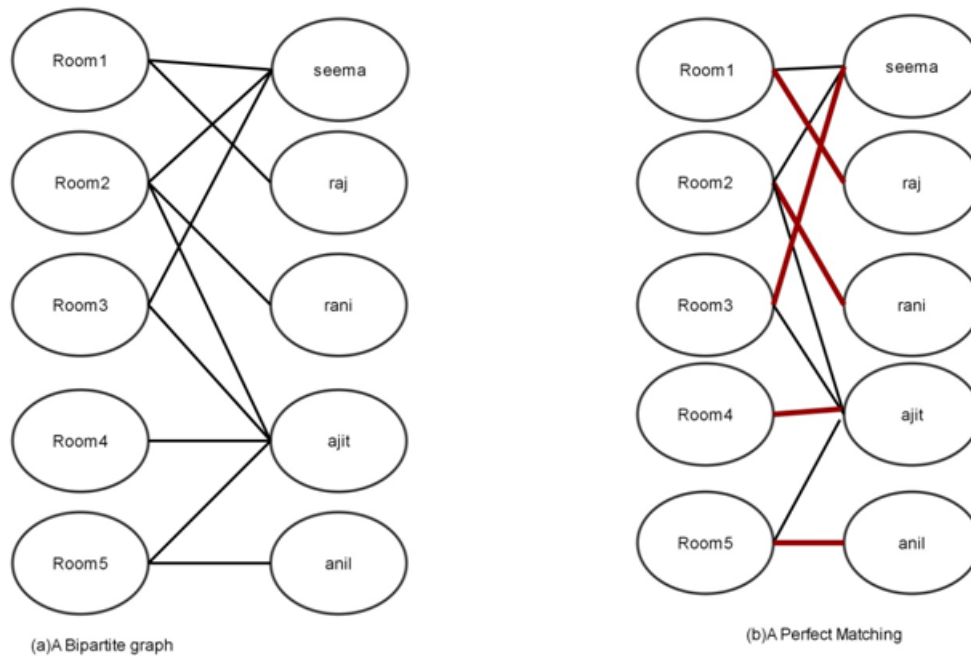(a)A Bipartite graph   (b)A Perfect Matching

Figure 6: A Bipartite Graph and A Perfect Matching

When there are an equal number of nodes on each side of a bipartite graph, a **perfect matching** is an assignment of nodes on the left to nodes on the right, in such a way that
I)   each node is connected by an edge to the node it is assigned to, and
II)   no two nodes on the left are assigned to the same node on the right

A set of nodes are called **constricted set** when their edges constrict the formation of a perfect matching. In Fig 7 Seema, Raj and Rani form a constricted set.
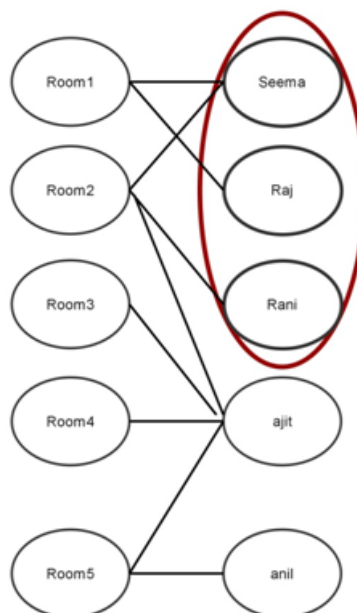

Figure 7: Constricted set

II. CONCLUSION

Bipartite graph finds a very large number of applications in science, engineering and technology and medical fields.

The bipartite graphs and perfect matching algorithms can solve many problems in cloud computing and cognitive radio networks. The research work in future focuses on modeling the cloud computing problems and cognitive radio network problems. Applications of bipartite graph in computer science especially in the above mentioned areas are rarely discussed in the literature.

REFERENCES

[1]. Armen S. Asratian, LuleåTekniskaUniversitet, Sweden,TristanM.J.Denley university of Mississippi , RolandHaggkvist, UmeåUniversitet, Sweden,"Bipartite Graphs and their Applications, June 2008 , ISBN: 9780521065122

[2]. Diestel, Reinard (2005). Graph Theory Grad.Text in Math .Springer.ISBN 978-3-642-14278-9.

[3]. Bo JIANG, Xiao-xiao ZHANG, Wei-feng PAN, Bo HU,"BIGSIR: A Bipartite Graph Based Service Recommendation Method ",2013 IEEE Ninth World Congress on Service,978-0-7695-5024-4/13 $26.00 © 2013 IEEE,pp-363-369, IEE Computer Society.

[4]. Chartrand, Gary; Zhang, Ping (2008), Chromatic Graph theory , Discrete Mathematics And Its Applications **53**, CRC Press, p. 223, ISBN 9781584888000.

[5]. Biggs, Norman (1994), Algebraic Graph Theory , Cambridge Mathematical Library (2nd ed.), Cambridge University Press, p. 53,ISBN 9780521458979.

[6]. Chien-Hung Chen,,Jenn-Wei Lin, and Sy-Yen Kuo,"Deadline-Constrained MapReduceSchedulingBased on Graph Modelling",2014 IEEE International Conference on Cloud Computing,978-1-4799-5063-8/14 $31.00 © 2014 IEEE,DOI 10.1109/CLOUD.2014.63,pp.416-423.

[7]. http://www.mathcove.net/peterson/lessons

[8]. http://citeseerax.ist.psu.edu/viewdoc/download?doi=10.1.1.1.186.3904&rep=rep1&type=pdf

[9]. C. BERGE, Two theorems in graph theory, Proc. Nat. Acad. Sci. U.S.A., 43 (1957), pp. 842-844.

[10]. J. EDMONDS, Paths, trees and.flowers, Canad. J. Math., 17 (1965), pp. 449-467.

[11]. L. R. FORD AND D. R. FULKERSON, Flows in Networks, Princeton University Press, Princeton, N.J.,1965.

[12]. M. HALL, Distinct representatives of subsets, Bull. Amer. Math. Soc., 54 (1948), pp. 922-926.

[13]. H. W. KUHN, The Hungarian methodjbr the assignment problem, Naval Res. Logist. Quart.,2 (1955), pp. 83-97.

[14]. http://www.cs.helsinki.fi/u/floreen/ncm/ncmPart3.pdf

[15]. [http://www.mathcove.net/Petersen]

[16]. M. Domijan,Mathematics Institute, University of Warwick,Coventry CV4 7AL, United Kingdom,"What are... some graphs of chemical reaction networks?,available at:- http://www.math.uzh.ch/?file&key1=9116.

[17]. Jean-Loup Guillaume and MatthieuLatapy,"Bipartite Graphs as Models of Complex Networks"http://denif.ens-lyon.fr/data/algorithmiquereseaux telecoms enslyon /2007/cours/biparti.pdf

[18]. Milan Bradonji´c, ,LoukasLazos," Graph-based Criteria for Spectrum-Aware Clustering in Cognitive Radio Networks,