Augmenting Matching Algorithm on the Bipartite Graphs

Wai Oo Paing, Khin Phyo Thant
University of Computer Studies, Yangon
waioopaing@gmail.com

Abstract

Discrete or Decision Mathematics is a useful subject for making the undesirable or impossible decisions especially for the assignment problems. These problems can be expressed in the graphs because graphs are used to solve problems in many fields and applications. Some applications are scheduling, finding the shortest paths and matching the elements of two different sets which are called the assignment problems. Although there are many kinds of graphs, the bipartite graphs provide the most useful way to represent the matching problems. This paper represents how the matching problems are implemented in the bipartite graphs and (0, 1) matrices. It also shows how the augmenting matching algorithm is applied in these problems to get the complete matching. It also shows that the original or the user chosen matching is already maximal for some cases. In this paper, the personnel assignment problem is used as the typical matching problem.

1. Introduction

Decision mathematics has become popular in recent decades because of its applications to computer science. Among many kinds of Decision mathematics problems, matching the elements of two different sets is a common task which we often perform by inspection without being aware of it. But inspection can miss the chance of the maximal or complete matching when the numbers of vertices and edges increase. Matching can also become difficult when the elements of two different sets have undesirable or impossible characteristics. For example, in the personnel assignment problem, the qualifications of the applicants and status of the jobs are such characteristics. To solve these difficulties, the bipartite graphs, (0, 1) matrices and the augmenting matching algorithm can be used.

That is because graph theory is a very fascinating branch of Mathematics in which we study objects called graphs. It can also play an important role in computer science because it provides an easy and systematic way to model many problems. The problems expressed in terms of graphs can be solved using the standard graph algorithms.

In this system, the personnel assignment problem is implemented as the typical matching problem for the certain agent company. So, the system can match the suitable applicants for the jobs in the five fields: accounting, engineering, information technology, management and general. For this system, the user can input the qualifications of the applicants and the requirements of the jobs for these five fields. Then, the system will check who is suitable for which job and change this relation into bipartite graph and (0, 1) matrix. Finally, Augmenting Matching Algorithm is applied on this (0, 1) matrix to get the complete or maximal pairs of matching.

2. Related Work

A variety of approaches has been proposed to solve the graph matching problems. An incomplete list includes Edmonds' Blossom Shrinking for maximum matching [7], Cycle Shrinking Lemma [6], Hopcroft-Karp algorithm [2] and Hall Marriage Theorem [3]. The above literatures strictly focus on reducing the running time for solving the graph matching problem. So, they always try on the better algorithms. In this way, they usually use the network flow algorithms which are complex for the users.

However, Augmenting Matching Algorithm of Alan Tucker [1] is an algorithm for a (0, 1) matrix M with a given set I of independent 1's of the bipartite graph. Because of its newness, this may seem more difficult than other familiar matching algorithms. But, with a little practice, the matrix method is easier to use. It is also sure to get the maximal or complete pairs of matching.

3. Theory Background

3.1. Basic Notations and Terminology

In very simple terms, a graph G is a collection of vertices and edges. Vertices are nodes which represent for the elements of the sets and edges are lines which represent for the links of those elements of the sets.

Bipartite Graphs are the natural mode
for the matching problems. We let X and Y be the two sets to be matched and edges (x, y) represents pairs of elements that may be matched together. For example, Set X contains the elements: a,b,c,d and set Y contains the elements: 1,2,3,4. For this case, the bipartite graph will be as in Figure 1.

A bipartite graph G=(X, Y, E) shown in Figure 1 is an undirected graph with two specified vertices X and Y and with all edges of the form (x, y), $x \in X, y \in Y; that$ is edges only connect vertices from X to Y and not Y to X.

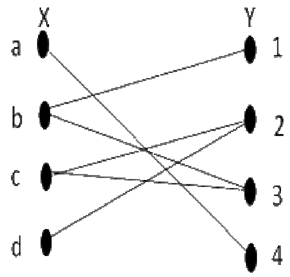

Figure 1. Bipartite Graph

A matching in a bipartite graph is a set of independent edges (with no common endpoints). The darkened edges in Figure 2 show a matching of (b,1), (c,3), (d,2) .

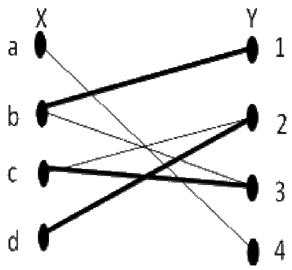

Figure 2. A Matching in Bipartite Graph

The darkened edges in Figure 3 show a matching of (a,4), (b,1), (c,3), (d,2) . It is an X-matching involving all vertices in the set X. This means that all the elements of the set X is more favorable to get the pair of elements of other set Y although all the elements of the other set Y cannot get the suitable pairs.

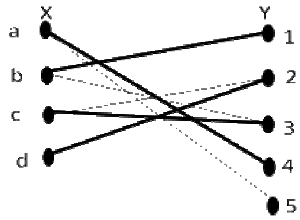

Figure 3.An X-Matching in Bipartite Graph

The darkened edges in Figure 4 show a matching of (a,4), (d,1) . Although there are four elements of each set in this graph, the only two pairs of matching can be obtained because of the relationships between the elements of the two sets. So, it is a maximal matching of the largest possible size.

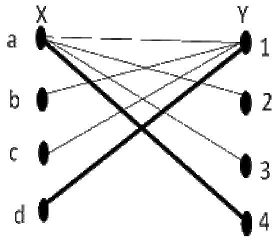

Figure 4.A Maximal Matching in Bipartite Graph

The darkened edges in Figure 5 show a matching of (a,4), (b,1), (c,3), (d,2) . The four elements of each set in this graph can get their pairs because of the relationships between the elements of the two sets. So, it is a complete matching covered all of the vertices of the two sets.

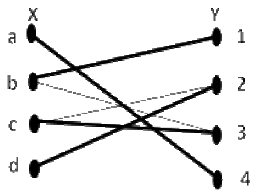

Figure 5.A Complete Matching in Bipartite Graph

A (0, 1) matrix shown in Figure 6 is an equivalent model for matching problems of the bipartite graph shown in Figure 1. We make a row for each element in X, a column for each element in Y, and place a 1 in the entry of row x and column y whenever x can be paired with y; other entries have a 0. Such a matrix is merely the adjacency matrix for the associated bipartite graph. A matching in a (0, 1) matrix is a set of independent 1-entries (no two of which are in the same line, i.e., in the same row or column). The circled set of independent 1's in Figure 6 corresponds to the matching in Figure 2.

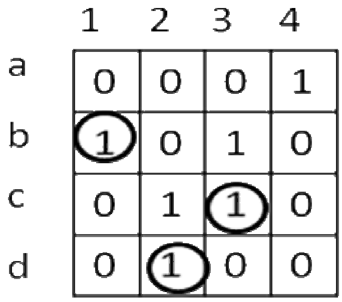

Figure 6. A (0, 1) Matrix

3.2. Augmenting Matching Algorithm

Augmenting Matching Algorithm is an algorithm for a (0, 1) matrix M with a given set I of independent 1's of the bipartite graph. Because of its newness, this algorithm may seem more difficult than other familiar matching algorithms.

However, with a little practice, the matrix method is easier to use. It is also sure to get the maximal or complete pairs of matching.

The major steps of this algorithm are as follow:

Step-1: Circle the 1's in I and label with an * every row without a circled 1.

Step-2: Scan every newly labeled row i; label with an i each unlabeled column with an un-circled l in row i.

Step-3: Scan every newly labeled column j; label with a j the row with a circled 1 in column j. If there is no circled 1, go to Step 5.

Step-4: If new rows were labeled in Step 3, go to Step 2. Otherwise the given set of circled 1's is maximal and the unlabeled rows together with the labeled columns constitute a minimal 1's cover.

Step-5: A "breakthrough" has occurred.

- In the column j_0 just scanned (with no circled 1),
- place a circle around the 1 in row i_1 designated by the label for column j_0
- in row i_1 remove the circle around the 1 in column j_1 designated by the label for row i_1
- Now in column j_1 place a circle around the 1 in row i_2 designated by j_1's label
- In row i_2 remove the circle around the 1 in column j_2; and so forth.

Continue this process until a row with an * label is reached.

3.3. Some Applications of Bipartite Graphs

- Personnel Assignment Problem
- Optimal Assignment Problem
- Marriage Problem
- Classroom assignment
- Scheduling
- Opponents selection for sport competitions

4. System Implementation

4.1. Implementation of the System

As the typical graph matching problem is the personnel assignment problem, the qualifications of the applicants and status of the jobs can make conflicts to decide who is suitable for the certain job. So, the system will be implemented as follow in Figure 7.

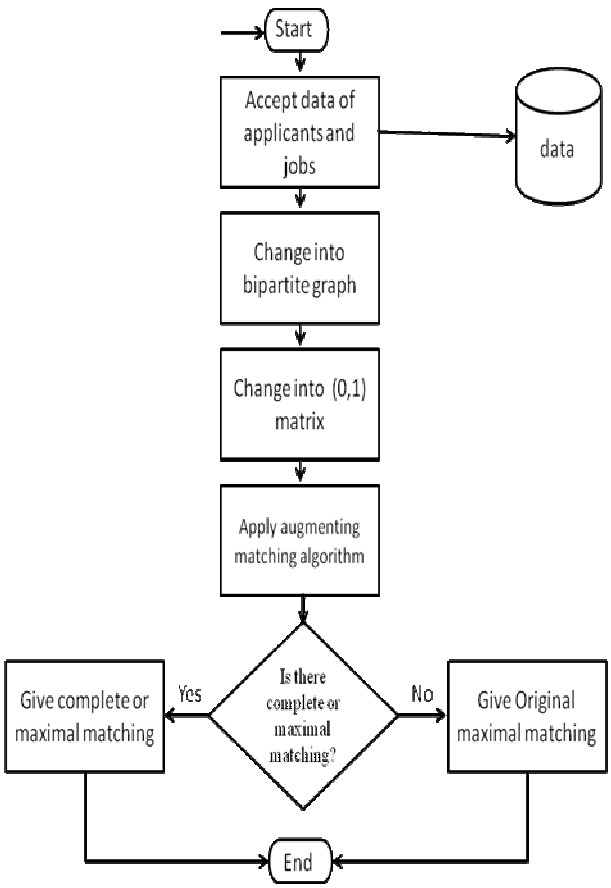

Figure 7. System Flow Chart

As the system is implemented as in Figure 7, at first, the system will accept the data of the applicants and jobs. To accurate in deciding which applicant is suitable for which job, the input data will also be checked like in Figure 8.

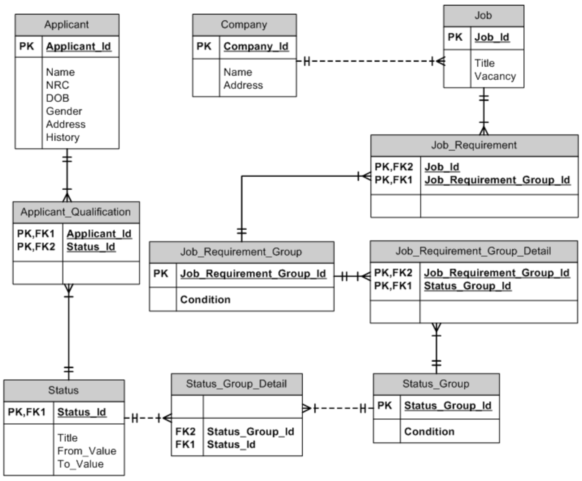

Figure 8. Entity Relationship Diagram

4.2. Sample Bipartite Graph Matching Problem

Although the bipartite graphs can be used to model many kinds of applications, the personnel assignment problem is chosen as typical bipartite graph matching problem in this paper. To understand this system easily, the sample personnel problem with the small numbers of applicants and the jobs will be implemented.

Suppose that there are three applicants: ‘a’, ‘b’, ‘c’ and three jobs: ‘1’, ‘2’, ‘3’ in the sample personnel assignment problem. After the checking for which applicant is suitable for which job is done as in Figure 8, suppose that the system will give as follow:

- The applicant ‘a’ can get the jobs ‘1’ and ‘2’.
- For the applicant ‘b’, the jobs ‘2’ and ‘3’ are suitable.
- But the applicant ‘c’ can only choose for the job ‘1’.

As the system will implement as in Figure 7, it will accept the data of the applicants and jobs using this relationship. And change it into the bipartite graph shown in Figure 9 for this relationship.

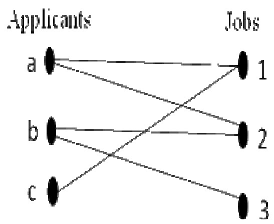

Figure 9. Bipartite Graph for Sample Problem

And change it into (0, 1) matrix as in Figure 10 for this relationship.

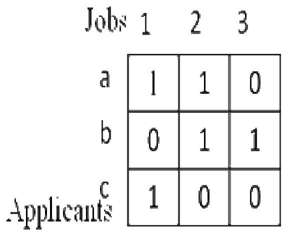

Figure 10. (0, 1) Matrix for Sample Problem.

The above figures, Figure 9 and Figure 10, show which applicant can get which job.

At that time, suppose that the user matches them like in the darken edges in Figure 11. So, the applicant ‘c’ can miss to get the suitable job though he can get it.

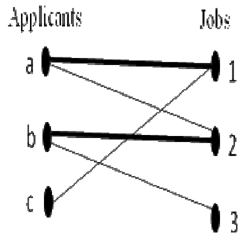

Figure 11. Bipartite Graph for User Chosen Matching

The circled ‘1’ of (0, 1) matrix in Figure 12 also shows the user chosen matching.

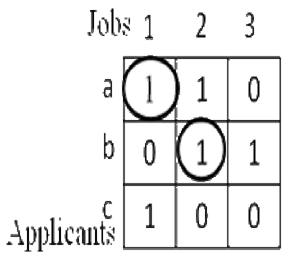

Figure 12. (0, 1) Matrix for User Chosen Matching

Then, apply the steps of augmenting matching algorithm as follow and Figure 13:

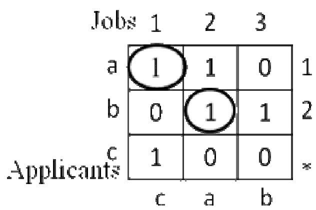

Figure 13. (0, 1) Matrix after applying Augmenting Matching Algorithm

Step-1: c ⟵ *

Step-2: 1 ⟵ c

Step-3: a ⟵ 1
Step-4: Step-2: 2 ⟵ a
Step-3: b ⟵ 2
Step-2: 3 ⟵ b
Step-5: A breakthrough.

After tracing the above path, the pairs of matching will be as shown in Figure 14. Now, all the applicants: a, b and c can get their suitable jobs by improving (augmenting) the user-chosen matching although the user can only match the two applicants to get the suitable jobs at first.

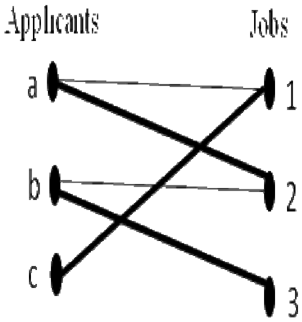

Figure 14. Bipartite Graph with augmented matching

In this way, the system will give the complete matching of the pairs for the applicants and the jobs in the typical assignment problem.

4.3. Another Sample Bipartite Graph Matching Problem

Sometimes, the complete matching cannot be found in some personnel assignment problem. For example, there are three applicants: 'a', 'b', 'c' and three jobs: '1', '2', '3' in the sample personnel assignment problem. After the checking for which applicant is suitable for which job is done as in Figure 8, suppose that the system will give as follow:

- The applicant 'a' can get the jobs '1', '2' and '3'.
- For the applicant 'b', the job '1' is only suitable.
- But the applicant 'c' can only choose for the job '1'.

As the system will implement as in Figure 7, it will accept the data of the applicants and jobs using this relationship.

And change it into the bipartite graph shown in Figure 15 for this relationship.

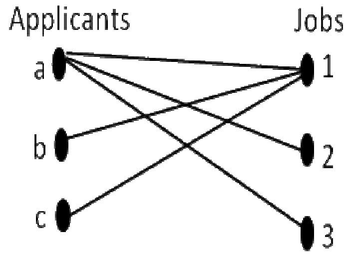

Figure 15. Bipartite Graph for Another Sample Problem

And change it into (0, 1) matrix as in Figure 11 for this relationship.

Applicants \ Jobs	1	2	3
a	1	1	1
b	1	0	0
c	1	0	0

Figure 16. (0, 1) Matrix for Another Sample Problem

The above figures, Figure 15 and Figure 16, show which applicant can get which job.

At that time, suppose that the user matches them like in the darken edges in Figure 17. So, the applicant 'b' can miss to get the suitable job though he can get it.

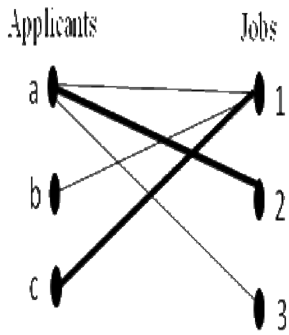

Figure 17. Bipartite Graph for User Chosen Matching

The circled '1' of (0, 1) matrix in Figure 18 also shows the user chosen matching.

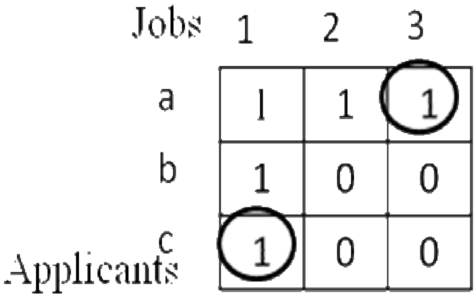

Figure 18. (0, 1) Matrix for User Chosen Matching

Then, apply the steps of augmenting matching algorithm as follow and Figure 19:

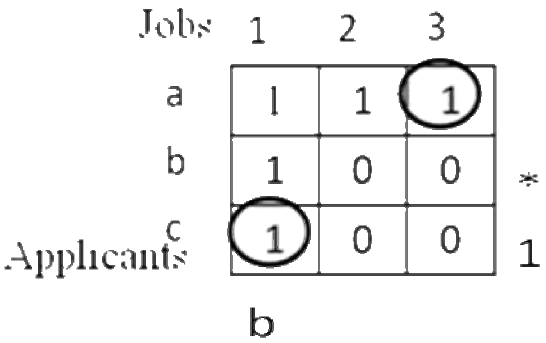

Figure 19. (0, 1) Matrix after applying Augmenting Matching Algorithm

Step-1: b ⟵ *
Step-2: 1 ⟵ b
Step-3: c ⟵ 1
Step-4: Otherwise, the given set of circled 1's is maximal.
Step-5: A breakthrough.

After tracing the above path, the pairs of matching will be as shown in Figure 20 like the Figure 15 of the user chosen matching.

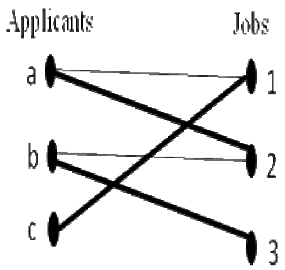

Figure 20. Bipartite Graph with augmented matching

In this way, the system can show that the user chosen matching (the original matching) is already maximal though it is not complete matching.

5. Conclusion

Although there are many kinds of matching problems, the personnel assignment problem is chosen as the typical problem in this system. That is because nowadays no one can stand in his life without job and assigning the suitable applicant to the certain job plays the important role of today's world.

As this system aims to the agent companies, it can accept the data for the applicants and jobs in the five fields: accounting, information technology, engineering, management and general, can check who is suitable for which job. Finally, it can give the suitable pairs of the matchings of the applicants and jobs in any cases. Basing on the matrix, it can accept and match many numbers of the applicants and jobs although it takes a little longer when the matrix is large. In this way, the user can know who is suitable for which job and how to engage which applicant for which job to avoid the vacancy in the company.

It is implemented using the C#.NET (ASP.NET), MSSQL Server 2005, MS.NET Framework V3.5.0 and Internet Information Service (IIS) – Web Server.

6. References

[1] Alan Tucker, *Applied Combinatorics*, John Wiley & Sons, State University of New York of Stony Brook, 1980.

[2] David Jackson ID# 258894, *"A review of: An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs written by John E. Hopcroft and Richard M. Karp"*

[3] Ewa Romanowicz, Adam Grabowski, "*Formalized Mathematics*", Volume 12, Number **3**, University of Bialystok, 2004.

[4] Nick Thorpe, "*Introduction to Decision Mathematics*", MEI Conference, (05/07/07).

[5] Stan Dolan, "*Decision Mathematics 2*", Cambridge University Press.

[6]Santosh Vempala, "*Combinatorial Optimization: Matching Algorithms*", September 9, 14, 16.

[7] Uri Zwick, *Lecture notes for "Analysis of Algorithms: Maximum matching in general graphs"*, spring 2006.