International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:14, No:9, 2020

Matching on Bipartite Graphs with Applications to School Course Registration Systems

Zhihan Li

***Abstract*—**Nowadays, most universities use the course enrollment system considering students' registration orders. However, the students' preference level to certain courses is also one important factor to consider. In this research, the possibility of applying a preference-first system has been discussed and analyzed compared to the order-first system. A bipartite graph is applied to resemble the relationship between students and courses they tend to register. With the graph set up, we apply Ford-Fulkerson (F.F.) Algorithm to maximize parings between two sets of nodes, in our case, students and courses. Two models are proposed in this paper: the one considered students' order first, and the one considered students' preference first. By comparing and contrasting the two models, we highlight the usability of models which potentially leads to better designs for school course registration systems.

***Keywords*—**Bipartite graph, Ford-Fulkerson Algorithm, graph theory, maximum matching.

I. Introduction

NOWADAYS, the network is a mathematical structure in graph theory that has been widely applied to practical problems to represent relations between objects. For instance, the main data structure of the GPS navigation system is related to graph theory. This structure considers locations as separate nodes and the distance as the edges and finds the shortest route between two locations [1]. As an example of graph structures, the bipartite graph can be utilized to construe the connections between two sets of vertices. As an instance, it was used to analyze the relations between the job applicants' group and job positions group, maximizing matchings between the two parties. It can also present the connection between users and content they may take an interest in the online social network [2]. Similarly, this model can also apply in dating websites (man and woman) and the housing market (the buyers and houses resources), ensuring the most effective distribution of resources.

In this paper, we will use the bipartite graph, to discuss University of Washington (UW) students' registration preference towards 400-level economic courses. Two types of models will be presented: 1. considering the number of students as weight, 2. considering the level of preference as weight. We also discuss the situation that if students are able to choose all of the courses. By making those graphs, we can find how many students are able to choose the courses they preferred and the optimized situation that can ensure everyone's registration tendency.

Zhihan Li is an undergraduate student of University of Washington, Seattle, WA 98195 USA (phone: 206-218-2526; e-mail: zl298@uw.edu).

II. Preliminaries and Background

To further explain the models, we need to provide some essential background and terms. Since our research is based on graph theory, we will include content about bipartite graph, weighted and directed graph, and matchings in the following subsections.

A. Bipartite Graph

A bipartite graph G is a graph whose vertices can be divided into two disjoint sets. We set the two disjoint sets as U and V. Every edge only connects one element in U and one in V. The connection between two elements in the U group or V group does not exist [3]. Let us see an example in our real life, the labors and job resources. The labors want to get the jobs they prefer, and the employers need to hire enough employees. These two sets can form a bipartite graph since, first, they are independent and, second, the connection only exists between elements from different groups: apparently, it is impossible for labor, in set U, to find a job from other labors in the set U—only employers in set V can provide the positions for him. The same argument goes for the employers' side. Therefore, there are only links between U and V.

An example of bipartite graph G with two disjoint vertex sets U and V is shown in Fig. 1.

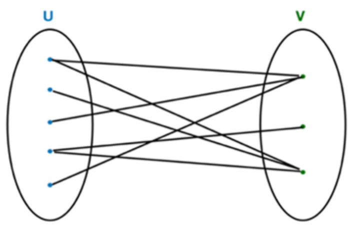

Fig. 1 A bipartitibe graph

B. Weighted Graphs and Directed Graphs

A graph G = E, V, W is a directed graph if the edges set E consists of ordered pair, which means all the edges are directed from one vertex to another [4].

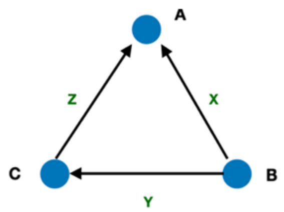

Fig. 2 A directed graph

International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:14, No:9, 2020

Fig. 2 shows a directed graph. Edge X here is directed from B to A; Edge Y is directed from B to C, and edge Z is directed from C to A. Since all the edges are directed, this graph can be called a directed graph.

If an edge $e \in E$ in the graph is assigned with a weight $w(e) \in W$, a numerical value, then it is a weighted graph. [5]

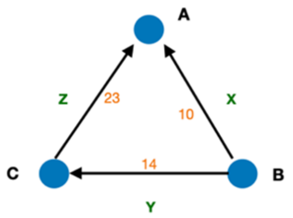

Fig. 3 A weighted graph

As Fig. 3 shows, this is a directed graph since the edges have directions suggested by the ordered pair of vertices. The “23”, “10”, “14” are the weights of the edges Z, Y, and X. Thus, the graph is also weighted. We call this graph a weighted directed graph. In our numerical tests in Section IV, all our graphs are weighted directed networks.

C. Matchings

In the given graph G, a matching, M is a set of edges that do not share any vertex in a graph [6]. Here, we remind the readers with three common types of matchings:

Maximal matching: if we add any edges that are not in the matching M and then it appears two edges sharing the same vertex, this matching is called a maximal matching [6].

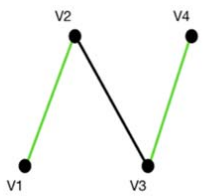

Fig. 4 A maximal matching

We consider the black edge (V2, V3) as M. Adding any green edges to the set will force two edges to meet at the same vertex. Thus, M = (V2, V3) is a maximal matching and it only contains one black edge.

Notice that the maximal matching is not unique. In Fig. 4, we can observe that the edge set contains the green edges and the set of black edge both can be considered as maximal matchings.

Maximum matching: A matching that contains the largest possible number of edges.

Here, we suppose that green edges formed set M’, then M’ is a maximum matching of G. If we set the black edge as the set M, it is only a maximal matching (|M| = 1) but since |M| = 1 < 2 =|M’|, it is not a maximum matching.

As mentioned before, we can find various sets of maximal matchings of a graph. However, there exists a maximum matching for any given graph, which is unique.

Perfect matching: a matching is perfect if every vertex in the graph is incident on a member of M. In Fig. 6, the green edges form a set of perfect matching as it connects all four vertices in the graph.

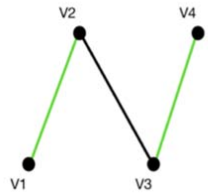

Fig. 5 A maximum matching

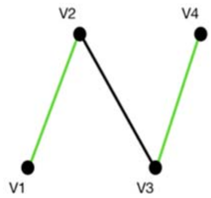

Fig. 6 A perfect matching

As a counterexample, the following set of green edges in Fig. 7 is not considered as a perfect matching since vertex 5 has no incident edge. In fact, it is impossible to find a perfect matching for this graph, since vertex 5 is an isolated vertex. This suggests that not all graphs can have a perfect matching. According to Hall’s Marching theorem, a perfect matching exists if and only if every subset S ⊆ L while G = (U, V) and U = L ∩ R.

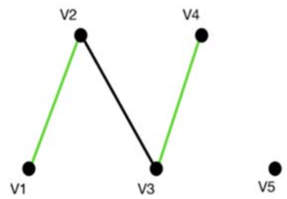

Fig. 7 A directed graph

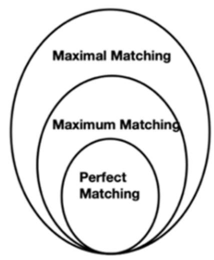

Fig. 8 Relationship

In this research, we use maximum matching to maximize the pairings between two vertex sets of the bipartite graph.

III. Algorithm

We use F.F. Algorithm in this research. It is a greedy algorithm that computes the maximum flow in a flow network.

Let G = (V, E) be the graph and c(u, v) be the capacity and f(u, v) be the flow. We also define $c_f(u,v) = c(u,v) - f(u,v)$ [7]. To find the maximum flow from source s and sink t, the pseudocode of the algorithm:

Inputs Given a Network G = (V, E) with flow capacity *c*, a source node *s*, and a sink node t

Output Compute a flow f from s to t of maximum value

1. F (u, v) ← 0 for all edges (u, v)
2. While there is a path p from s to t in residual network $G_f = (V, E_f)$, with $E_f = \{(u,v) \in V \times V : c_f(u,v) > 0\}$ such that c_f(u, v) > 0 for all edges (u, v) ∈ p:
   a. Find c_f(p) = min {c_f(u, v): (u, v) ∈ p}
   b. For each edge (u, v) ∈ p
      I. F (u, v) ← f (u, v) + c_f(p) (send flow along the path)

F (v, u) ← f (v, u) - c_f(p) (The flow might be "returned later"

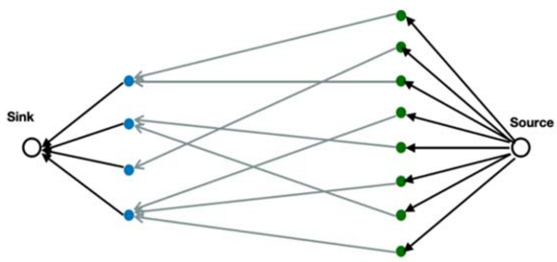

Fig. 9 How F.F. Algorithm works

In Fig. 9, we add a sink, a source and the black directed edges to the bipartite graph [blue and green vertices with the grey edges in between] and set directions to the grey edges, so that it becomes a directed network and the F.F. Algorithm can be applied.

IV. Data

The data are collected through a Google survey. The sample population is UW students major in Economics or ACMS. The survey has been put on several social media websites such as WeChat, Facebook, Reddit, etc.

Students are required to submit their preferences to four different 400-level Econ courses (major only): Econ 424-Computational Finance and Financial Econometrics; Econ 422-Investment, Capital, And Finance; Econ 485-Game Theory with Applications to Economics; Econ 483-Econometric Applications. The preference is shown by number. 4 means the most and 1 means the least. Students are allowed to choose the same preference for different courses. Furthermore, their decisions are limited by several other conditions: 1) Students have to take 400-level courses other than the four listed above. 2) Students can take up to 18 credits in one quarter. 3) They have to satisfy other major requirements and workloads. Therefore, students' preference towards courses is only one of the indicators among all other requirements that students have to consider when scheduling for classes. As a prototype of the model, we mainly consider students' preferences when optimizing the pairings between students and classes, but this can be further extended to a fuller model that eventually takes other factors into account. Thus, the survey also asks for students' choices in their formal plans, which are the 400-level courses they will truly take in the future quarter. There is no constraint for the number of courses students can choose, but each course can only be chosen by one time and we assume the capacity is 20 for each course.

In summary, the survey includes students' names, students' preferences toward each course, and students' formal choice on these four courses.

In this research, we use bipartite graphs to stimulate the relationship between students and 400-level courses. The students and 400-level Econ courses are considered as the two disjoint sets. We assign the weights to the edges between these two sets. Just imagine the weights as water, starting from students and then flowing to the courses. We construct two types of system: 1. the number of registered students is considered as the weights; 2. the level of students' preference is considered as the weights.

In Figs. 10-14, 57 is seen as the source and 58 is the sink. The nodes 1-52 represent the student indices. The 53-56 means four 400-level courses: 53 represents Econ 422. 54 is Econ 424. 55 is Econ 483; 56 is Econ 485.

The first type of system (use number of students as weights) is shown in Fig. 10.

The weight of the edges between the source and students is set as 4, which is the maximum number of courses students are able to choose. The weight of the edges from students to different courses is 1 since students are only able to choose one time on each course.

The matchings between nodes 1-52 and nodes 53-56 denote students' choices, which are the Econ courses in each student's formal plan, collected in our data. However, there will be a capacity constraint for each course, which is 20. So, students cannot always get what they want. In the current design, student 1 has more priority than student 13. In summary, the principle of this system is first come, first served.

What this system evaluates is the registration system that students use to register courses in order. This is the basically what UW is using: based on the class standing, students have different dates of registration.

The second system (uses preference level as weights) is shown in Fig. 11.

The edge weight to the source as the maximum preference level toward courses that each student is able to choose, which is 16, (the maximum number of courses, 4 times the maximum preference 4). The edge weight to the sink 58, in this case, is the whole preference level of registered students 80 since 20 is the maximum capacity and 4 is the maximum level of preference each student can achieve.

The matchings between the 1-52 and 56-53 denote the students' preference toward their choices, which is the Econ courses in each student's formal plan and preference levels, collected in our data. However, different from the first system, students having a higher preference level are more likely to get this course.

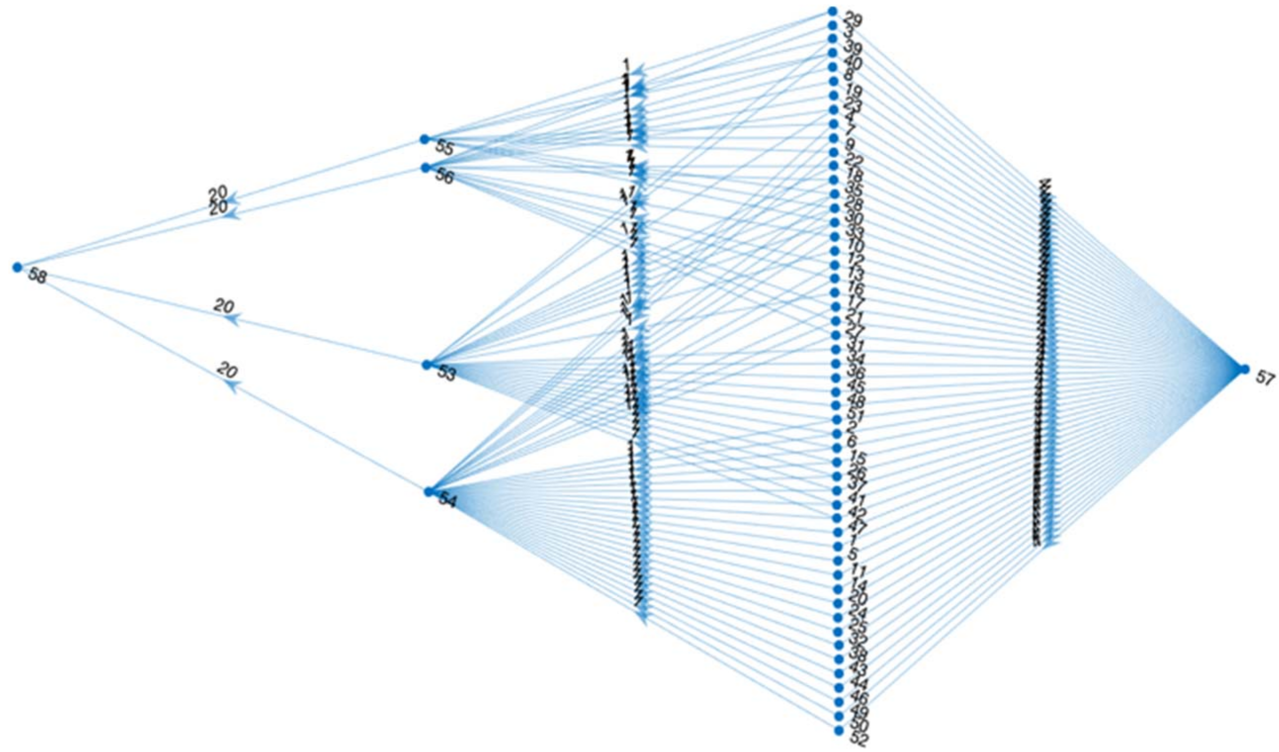

Fig. 10 Number of students as weights

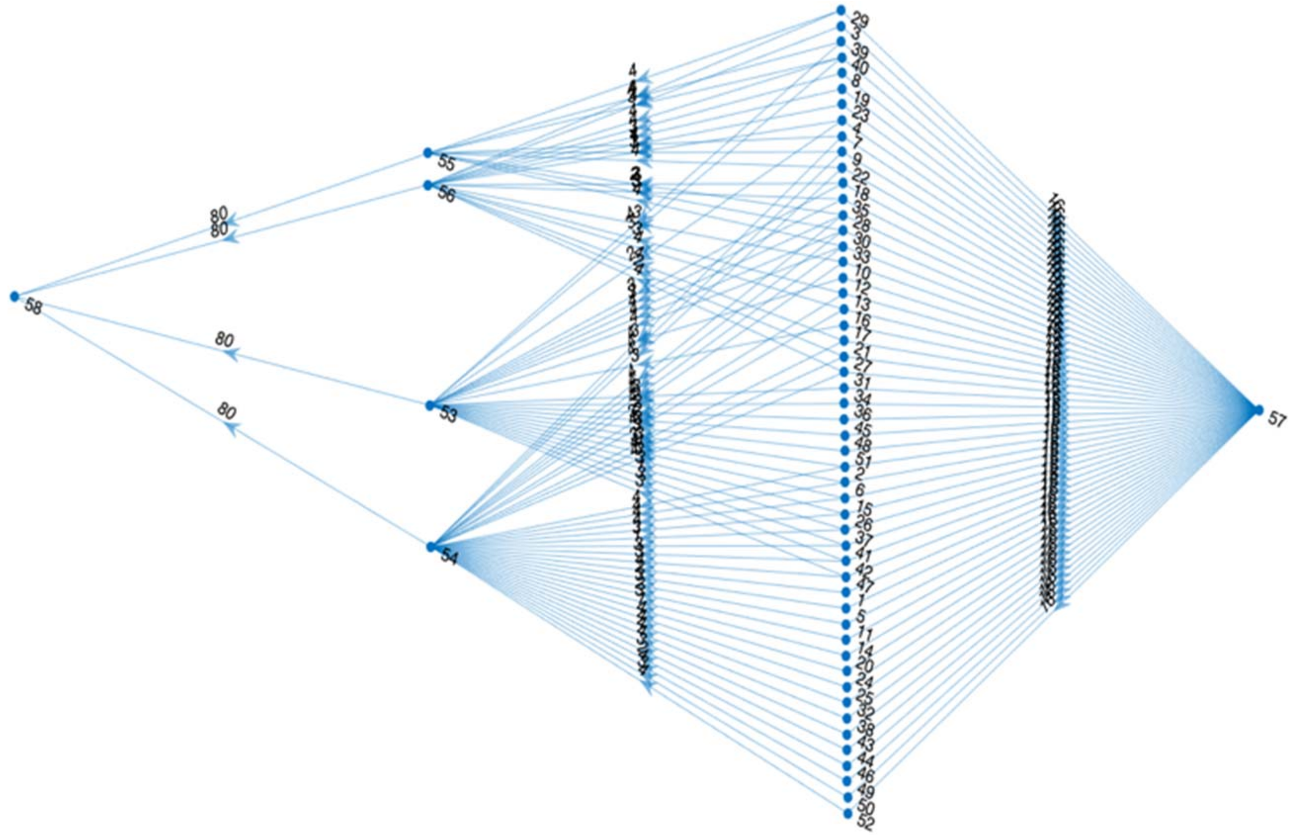

Fig. 11 Preference level as weights (1)

This system evaluates the registration system based on the students' preference level. If A and B both want the same course, the one with higher preference will be the first one considered by the system.

V. Numerical Results

Here we demonstrate the registration results of the two systems mentioned in Section IV.

A. Model 1

Firstly, we consider the number of students as weights and use F.F. algorithm in Section III: 57 as source, 58 as sink, and try to maximize the flow out from node 58.

The maximum capacity of every 400 level courses is 20. The graph is shown in Fig. 12.

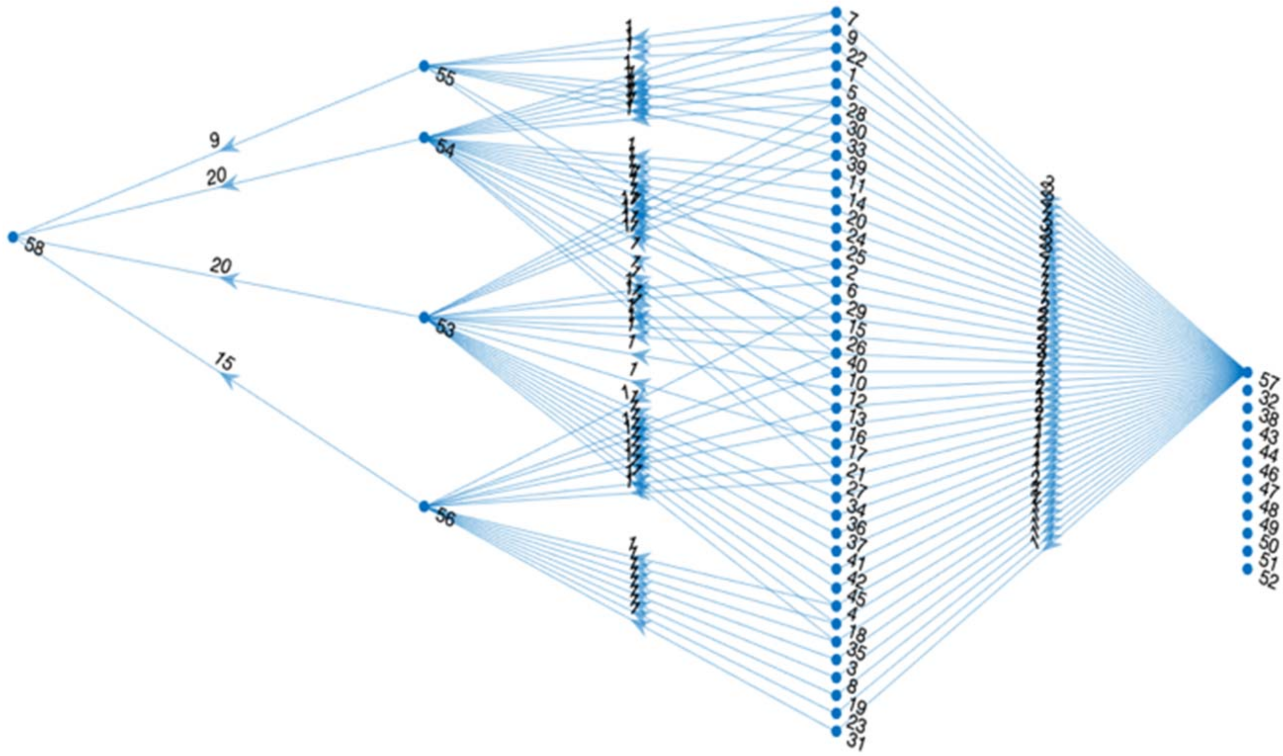

Fig. 12 Number of registered as weights

As mentioned before, this system is based on the chronological order. There are two courses reached capacity, 53 and 54, which are Econ 422 and Econ 424. Only a few students choose 55 and 56, which are Econ 485 and Econ 483. We also calculate the preference level on each course: there is 33 total preference level toward 485, 71 total preference level toward 424, 60 total preference level toward 422, and 45 total preference level toward 483. We could find something interesting from this data: the average preference level of each student in Econ 485 is 3.67; Although 424 and 422 are the most popular courses, their average preference level of each student in these two is 3.55 and 3, which means students taking 485 are happier than those taking 424 and 422.

11 students are failing to register for any courses under this model.

The average courses students have chosen is 1.57: 26 students choose 1 course; 22 students choose 2 courses; 4 students choose 3 courses; 0 students choose 4 courses.

B. Model 2

The second situation chooses the preference of students as weights. F.F. algorithm in Section III is applied here. We set 57 as source and 58 as the sink, then maximize the flow out from node 58. In this case, we only consider that students will only choose the courses they've submitted in the survey (*the question about what they will choose in their formal quarter plan). However, we find that there are 23 students paired with Econ 422 (53) and 35 students paired with the Econ 424 (54), which is larger than the courses' largest capacity. This is because, as mentioned before in Fig. 11, we set 80 (maximum capacity * maximum level of preference) as the weights to the sink 58. It is unusual for all students to choose the same course to have the highest level of preference. Therefore, the F.F. Algorithm keeps putting students into those courses until the total preference levels reach 80. The reason why this situation did not occur to two other courses is that the demand for those two is much less than Econ 422 (53) and 424 (54). To solve this problem, we adjust the matchings to 422 (53) and 424 (54): we only keep the 20 students with the highest preference scores to

International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:14, No:9, 2020

422 (53) and 424 (54), respectively. According to the feature of the F.F. Algorithm, the student-course pairings are selected from the highest weight to the lowest weight. Therefore, deleting smaller weight edges will not affect the paring results.

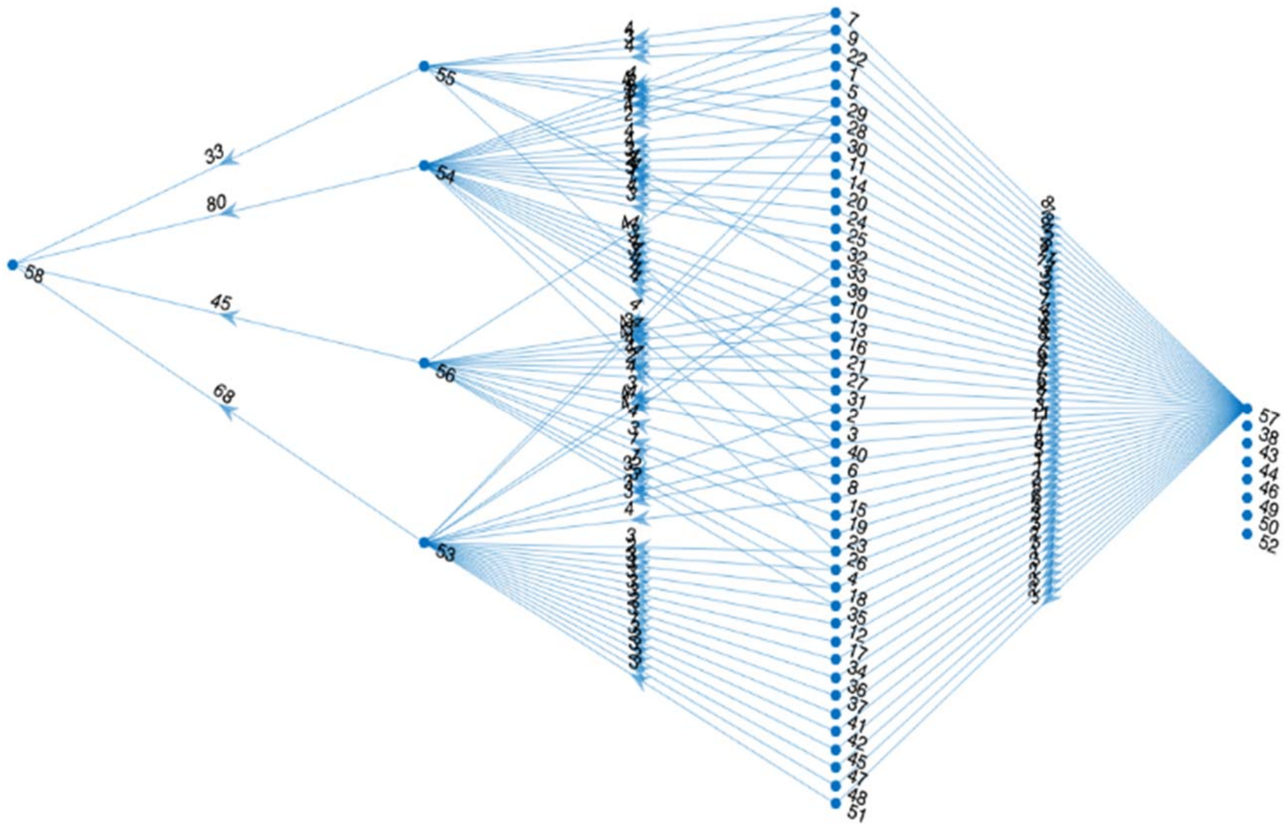

Fig. 13 Preference as weights (1)

Right now, all four courses have enrollments less than or equal to 20. Specifically, both Econ 422 (53) and Econ 424 (54) have 20 students. 9 students register into Econ 485 (55) and 15 students register into Econ 483 (56).

The total preference level in Model 2 is higher than that in Model 1 as calculated in Section V *A*: here, the preference level toward 424 is 75, which is higher than 71 toward 424 in the previous model. Also, the preference level toward 422 is 63, which is higher than 60 toward 422 in the previous model. By calculating the average preference level, we find that the Econ 485 and Econ 424 are the courses making students most satisfied since their average preference level is 3.67 and 3.75. Moreover, there are fewer people get registered in Model 2 compared to Model 1: 11 students cannot get register under Model 1 while 13 students left unenrolled in any courses under Model 2. In both cases, 38, 43, 46, 48, 49, 50, 52 students cannot register any course: this is because they only choose one course and the system is designed to maximize the level of the preference: for example, knowing that A has a level 3 preference and B has a level 2 preference toward Econ 424, if B also shows a preference of level 4 to Econ 383 and A only considers to take 424, the system will let B register for 424 since total preference is larger in this case (4+2=6 > 3).

VI. Conclusion and Future Work

In summary, we have applied graph theories and algorithms in our research: first, we use bipartite graphs and matchings to present the relationship of the 400 level courses and students. We also illustrate the weights can be added to matching to resemble the flow of resources. Based on that, we then use the F.F. algorithm to maximize the number of matchings between these two, which is exactly what course registration is supposed to do.

We propose two models by setting different values as weights to achieve different goals: if we set the number of students as weight, the model maximizes the number of registered students and the student in front order preempt the limited places of the courses; if we set the level of preference as weight, the model maximizes the level of preference and student with higher preference level has higher priority. Furthermore, we compare and contrast the results of two models: at least in this case, when a normal registration system, which registers students by measuring their order instead of the preference, is utilized, more students can get registered. But students will be much more satisfied if we use preference as weights. It is hard to conclude which model is preferred due to several reasons:

1. The size of samples is not larger enough so the representation of this dataset may not fully characterize the whole student body. For example, a large portion of the students in this limited dataset has a much higher preference for Econ 424, which may exaggerate the difference in preference level between two models.
2. There is no standard metric to quantify the quality of paring models when different objectives are applied to the problems. On the one side, it is reasonable to maximize the number of registered students since they need to graduate on time. On the other side, maximizing students' preference is also important for improving students' study efficiency. [8]

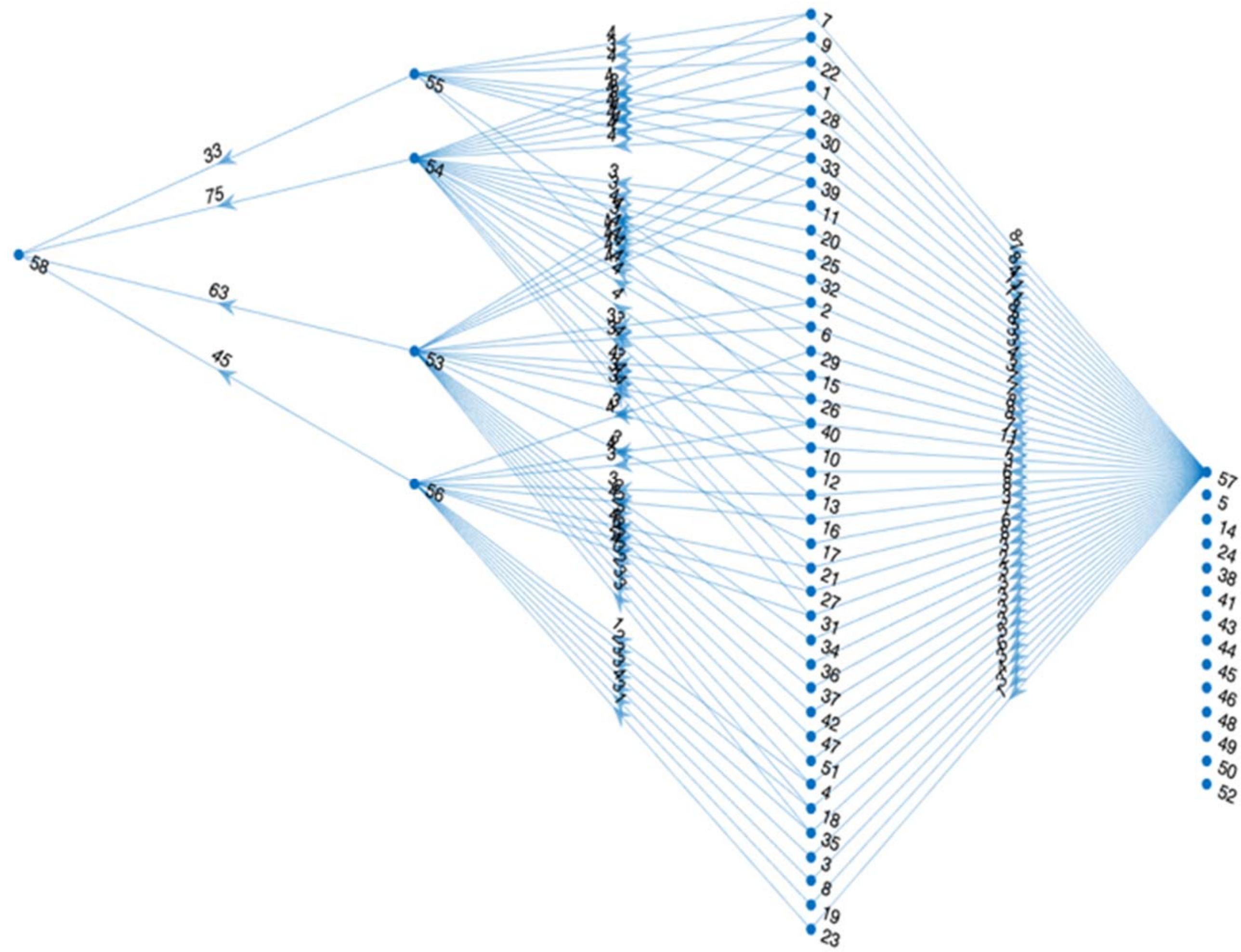

Fig. 14 Preference as weights (2)

As future work, the above mentioned two points can be addressed: if a larger dataset is collected and analyzed, we believe that the first point could be resolved since it will reflect the overall student body's distribution more accurately. The second point suggests that the flexibility in choosing the objective of the model. As of the current models, we only consider one feature (number of students in Model 1 and level of preference in Model 2) at a time when maximizing the student-course parings. However, one can combine multiple features at the same time and reflect them as graph weights, then the model will depict the real-life situation in a more well-rounded manner.

REFERENCES

[1] Marwan Abboud, Lina Mariya Abou Jaoude, and Ziad Kerbage." Real Time GPS Navigation System ", Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.560.3002&rep=rep1&type=pdf

[2] Zhu, Zhiguo, et al. "Measuring Influence in Online Social Network Based on the User-Content Bipartite Graph." Computers in Human Behavior, vol. 52, 2015, p. 184.

[3] Asratian, Armen S., et al. Bipartite Graphs and Their Applications. 1998.

[4] Strang, Gilbert. Linear Algebra and Its Applications. 3rd ed., Harcourt, Brace, Jovanovich, Pulishers, 1988.

[5] Weisstein, Eric W. "Weighted Graph". From MathWorld—A Wolfram Web Resource. https://mathworld.wolfram.com/WeightedGraph.html

[6] Gibbons, Alan. Algorithmic Graph Theory. Cambridge University Press, 1985.

[7] Ford, L. R, and Fulkerson, D. R. "Maximal Flow through a Network." *Canadian Journal of Mathematics*, vol. 8, 2018, pp. 399–404.

[8] Adamidis, Panagiotis, and Kynigopoulos, Georgios. "EvoWebReg." *International Journal of Operations Research and Information Systems*, vol. 5, no. 1, 2014, pp. 1–18.