An Average Case Approximation Bound for Course Scheduling by Greedy Bipartite Matching

Gary Lewandowski[1], Prakash Ojha[2], Jennifer Rizzo[1], and Abigail Walker[1]

[1] Xavier University, Mathematics and Computer Science Department, Cincinnati OH 45207-4441, USA {lewandow,rizzo,awalker}@cs.xu.edu
[2] Hope College, Computer Science Department, Holland MI 49423 ojha@cs.hope.edu

1 The Problem we are studying

1.1 Which scheduling problem?

We are studying the combined problem of generating a timetable and student schedules from initial data that includes student course requests, the number of sections allowed of each course, the maximum number of students allowed into each course, and the maximum number of times available in the timetable.

Our approach extends to handle issues of time constraints for instructors, time requests for students, and limitations on room availability but we do not consider these in this paper.

The goal, given the initial input, is to develop a timetable indicating the time at which each section of a course will be offered as well as a schedule for each student indicating the times at which the courses requested are scheduled. The timetable and schedules should be constructed to minimize the number of conflicts experienced by the students.

1.2 Metric for counting conflicts

Initial input to this problem is a set of course requests from each student. Upon completion each student has an actual schedule listing the particular sections into which the student has been placed. The difference between the number of courses requested and the number of timeslots used by the actual sections is the number of conficts experienced by the student.

This metric recognizes two ways of handling conflicts: a scheduling algorithm could either schedule the student into sections that are scheduled simultaneously or simply not schedule the student into sections that will have conflicts (i.e. if two sections are at the same time, the student will get one of those courses but not both).

2 Our focus

Our interest in this study is approximation bounds for the problem. In particular, we want to know if we can prove any approximation bound at all on the quality generated by a heuristic solution.

While many techniques have been used to approach school timetable problems, little work has appeared discussing the quality of solution. T-coloring graphs is related, and bounds have been proven using randomization, by Vitanyi [3], and semi-definite programming, by Frieze and Jerrum [2]. However, the definition of a conflict in coloring is different from a conflict in course scheduling so these results do not immediately apply. Berry, Condon and Halsey [1] were able to adapt Vitanyi's result to show that in the case of single-section courses, randomly scheduling each course yields a $1/(1-1/e)$ approximation algorithm, but it has proven difficult to adapt Frieze and Jerrum's result.

Our study examines bipartite matching as the basis for a greedy approximation algorithm. The heuristic to build the timetable and student schedules is very simple. Given a randomly ordered list of student course choices, each student's choices are processed as follows: construct a bipartite graph consisting of the courses desired by the student in one component, and the T timeslots in the other component. Edges are placed between course i and time j if the course has an unfilled section scheduled at time j, or if the course has a section that has not yet been scheduled. After building this bipartite graph, bipartite matching creates a match between courses and timeslots. The matching provides a schedule for the student and may also cause a section to be scheduled if the timeslot with which a course is matched has not already been assigned a section of the course.

3 Theoretical Results

The quality of approximation achievable using the bipartite matching heuristic varies with the number of sections available for each course. Our analysis makes a simplifying assumption that students choose their courses randomly. While this does not appear realistic, our empirical tests indicate this assumption appears to be a cause for underestimating the quality rather than overestimating.

Our first result shows that when each course has a single section, this heuristic achieves an approximation ratio $< 1/(1-1/e)$, i.e, we can prove the performance is better than random assignment of times to sections, but cannot prove it is much better.

Our second result analyzes the case of each course having two sections. We are able to show that the approximation ratio ρ is between $1/(1-1/e^2)$ and 1.255. That is, average use of greedy bipartite matching yields results with at most 25.5% more conflicts than optimal. Empirical results show that in practice we see an average of about 18.5%.

Analyzing the algorithm beyond the case of two sections per course proves untractable and we turn instead to simulations and empirical studies.

4 Empirical Evidence

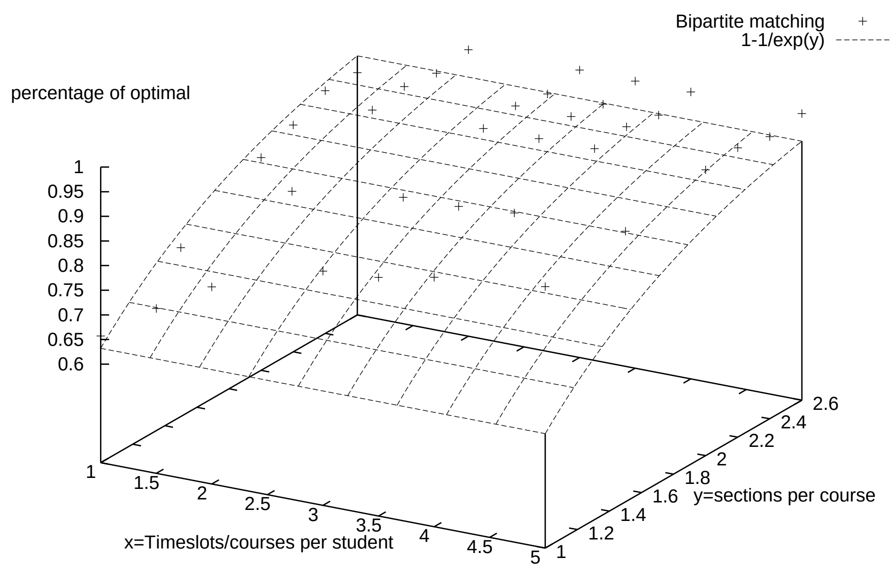

Fig. 1. Varying T/r as well as section ratios

We simulated the algorithm on a wide variety of cases, varying the number of timeslots, number of courses requested by the students, number of section, and number of student. Figure 1 shows our main results, plotted as 1/approximationRatio. We see that

- as the *average* number of sections, s, increases, the quality of the solution increases correspondingly, following the curve $1/(1 - 1/e^{\Theta(s)}$.
- if the number of timeslots T does not equal the number of courses chosen by each student, r, then the approximation bound follows a curve which is roughly $1/(1 - 1/e^{\Theta(s*T/r)}$.

5 How this compares to other aproximation techniques

Having analytically discovered the greedy bipartite matching algorithm should give promising results, we next studied it in comparison with simulated annealing.

We report here early results and are working on a more tuned version of simulated annealing for a better comparison. Our initial results indicate that the matching heuristic is competitive in quality with simulated anealing. In Table 1 we display the results of our first study. The data sets are named as numberOfSections-totalSeatsDesired-totalAvailableSeats, e.g. 2-653-900. The seats desired is simply the total number of course requests by students. The total available seats is the number of seats allowed per section times the total number of sections in the timetable. We report the highest, lowest, and the median number of conflicts over five runs of each algorithm. We also include the approximation ratio.

Table 1 indicates that simulated annealing does much better if only one section is available per course, but the two methods are relatively comparable when there are two sections of each course.

Data Set	Simulated Annealing				Bipartite Matching			
	h	l	m	approx ratio	h	l	m	approx ratio
1-40-100	40	40	40	1.000	39	34	36	1.111
1-70-100	70	70	70	1.000	70	57	59	1.186
1-650-900	650	650	650	1.000	512	480	504	1.290
1-2800-3750	2767	2608	2713	1.032	1935	1891	1926	1.454
2-38-100	38	38	38	1.000	38	36	38	1.000
2-36-200	36	36	36	1.000	36	35	36	1.000
2-72-200	72	72	72	1.000	72	70	72	1.000
2-653-900	639	635	637	1.025	622	605	617	1.058
2-2556-3000	2397	2384	2395	1.067	2378	2358	2364	1.081
2-6922-15000	6278	6244	6258	1.106	6186	6173	6181	1.120
2-6952-25000	6425	6396	6413	1.084	6309	6256	6282	1.107
2-10382-15000	9320	9257	9262	1.121	9207	9185	9195	1.129
2-10422-25000	9475	9404	9439	1.104	9329	9278	9307	1.120
2-13849-15000	12402	12211	12243	1.131	12230	12117	12154	1.139
2-13880-25000	12509	12159	12443	1.115	12336	12301	12322	1.126
2-20363-25000	18208	18157	18190	1.119	18128	18103	18110	1.124

Table 1. Simulated Annealing and Bipartite Matching results

References

1. C. Berry, A. Condon, B. Halsey. Best Schedule, manuscript, 1995.
2. A. Frieze and M. Jerrum. Improved approximation algorithms for MAX k-CUT and MAX BISECTION. Algorithmica, 18(1):67–81, May 1997.
3. P.M.B. Vitányi. How well can a graph be n-colored?, *Discrete Mathematics*, 34, 1981, 69-80.