An Integer Programming model for a matching problem

Sofiat Olaosebikan, Duncan Milne and David Manlove
School of Computing Science, University of Glasgow

1. What matching problems are

- Matching problems generally involve
  - assigning a set of agents to another set of agents;
  - based on the preferences of the agents, and
  - some problem-specific constraints.
- First studied by Gale and Shapley [1]
  - they described the *College Admissions problem* which involves assigning applicants to colleges;
  - they also described the *Stable Marriage problem* which involves the optimal assignment of n men to n women.

2. Example applications

1. The National Resident Matching Program (NRMP) in the United States [2] employs a matching algorithm to allocate medical students to hospitals.

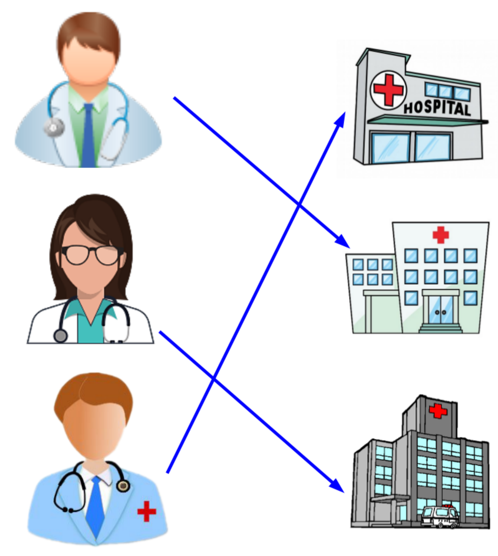


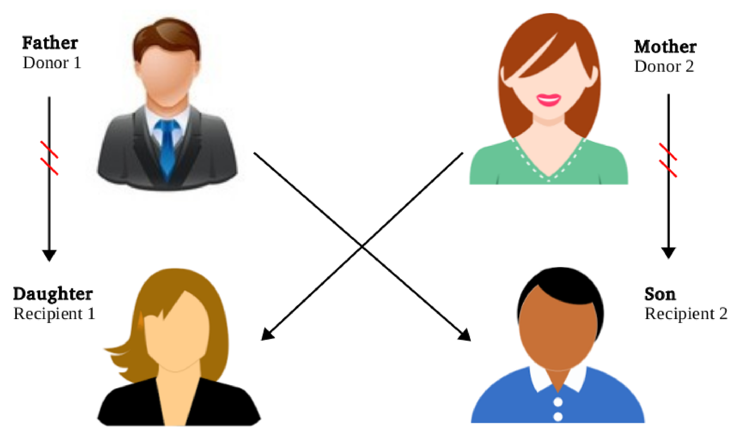
Figure 1: *Hospitals-Residents problem* (HR).

2. A generalisation of HR arises when university departments seek to allocate students to projects.

Students				Lecturers
s_1:	p_3	p_2	p_1	l_1 offers p_1 and p_2
s_2:	p_1	p_2		l_2 offers p_3
s_3:	p_3			

Figure 2: *Student-Project Allocation problem* (SPA).

3. The *Kidney exchange problem*



Where my research comes in

The inherent complexity of some of the open problems and their important applications motivate my research in the area of *efficient (polynomial-time) algorithms for matching problems.*

3. A matching problem definiton

A variant of SPA where:

- students and lecturers have preferences over projects,
- projects and lecturers have positive capacities,

is known as the *Student-Project Allocation problem with preferences over Projects* (SPA-P) [3].

4a. An instance of SPA-P

Preferences

Students				Lecturers		
s_1:	p_3	p_2	p_1	l_1:	p_1	p_2
s_2:	p_1	p_2		l_2:	p_3	
s_3:	p_3					

Figure 3: Preference lists are strictly ordered, student s_1 prefers p_3 to p_2, and so on. Each project has capacity 1. Lectures l_1 and l_2 have capacity 2 and 1 respectively.

The goal is to find a *matching* such that:

- each student is assigned at most one project;
- the capacities of projects and lecturers are not exceeded.

4b. Unstable matchings

With respect to Figure 3, we have:

Students				Lecturers		
s_1:	p_3	p_2	p_1	l_1:	p_1	p_2
s_2:	p_1	p_2		l_2:	p_3	
s_3:	p_3					

Figure 4: Matched projects are circled in blue. (s_2, p_1) forms a *blocking pair*, s_2 and l_1 both prefer p_1 to p_2.

Students				Lecturers		
s_1:	p_3	p_2	p_1	l_1:	p_1	p_2
s_2:	p_1	p_2		l_2:	p_3	
s_3:	p_3					

Figure 5: $\{s_1, s_2\}$ forms a *coalition*, s_1 and s_2 would rather swap their assigned projects to be better off.

4c. We seek stable matchings

- one with no blocking pair and no coalition.

Students				Lecturers		
s_1:	p_3	p_2	p_1	l_1:	p_1	p_2
s_2:	p_1	p_2		l_2:	p_3	
s_3:	p_3					

Figure 6: A stable matching of size 2.

Students				Lecturers		
s_1:	p_3	p_2	p_1	l_1:	p_1	p_2
s_2:	p_1	p_2		l_2:	p_3	
s_3:	p_3					

Figure 7: A stable matching of size 3.

The varying sizes of these stable matchings leads to the problem of finding maximum cardinality stable matching given an instance of SPA-P, which we denote by MAX-SPA-P.

4d. Existing results for MAX-SPA-P

- MAX-SPA-P is NP-hard and approximable to within 2 [3].
- MAX-SPA-P is approximable to within $\frac{3}{2}$ [4];
  - this is the best known approximation algorithm for MAX-SPA-P, with a lower bound of $\frac{21}{19}$,
  - it produces a stable matching whose size is at least two-thirds of that of a maximum stable matching.

Question: Can we solve MAX-SPA-P to optimality?

Answer: Yes! – An Integer Programming (IP) model for MAX-SPA-P

We give a general construction of the model:

- create binary-valued variables to represent the assignment of students to projects;
- enforce the following classes of constraints:
  - find a matching;
  - ensure matching does not admit a blocking pair;
  - ensure matching does not admit a coalition;
- describe an objective function to maximize the size of the matching.

***Theorem**: Given an instance I of SPA-P, there exists an IP formulation J of I such that a maximum stable matching in I corresponds to an optimal solution in J and vice-versa.*

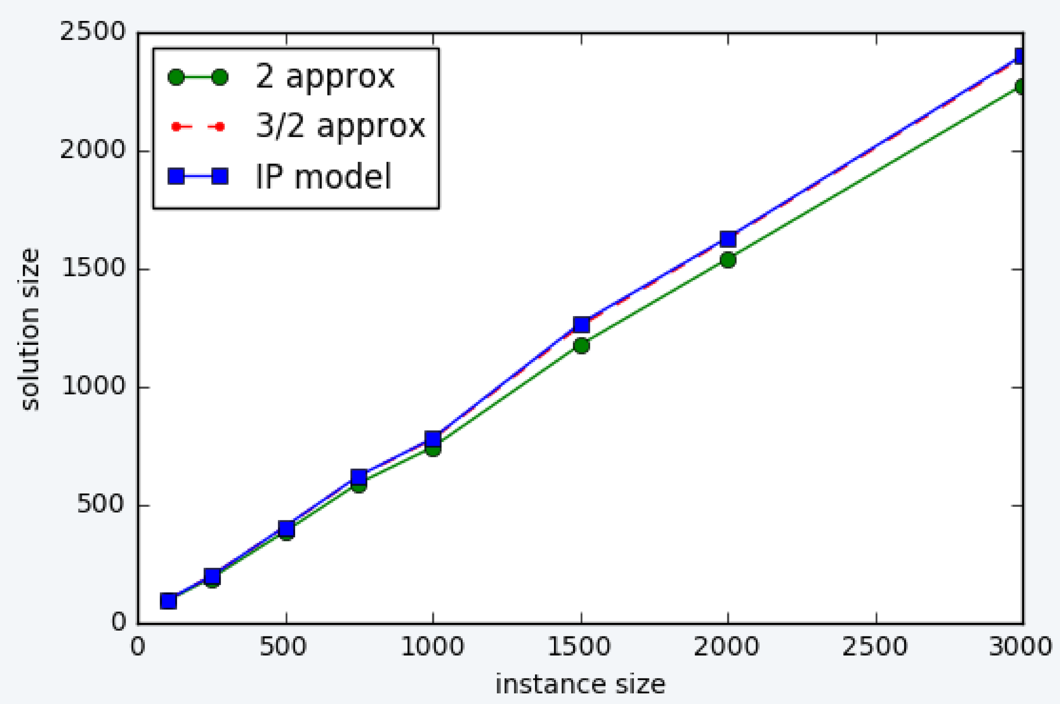


Figure 8: An empirical analysis that compares the approximation algorithms and the IP model for randomly generated SPA-P instances.

Conclusion: The solution produced by the $\frac{3}{2}$ approximation algorithm is extremely close to optimal!
Future work: To study properties of the preference lists that would lead to a significant difference between the solution produced by the IP model and the $\frac{3}{2}$ approximation algorithm.

References

[1] D. Gale and L.S. Shapley.
College admissions and the stability of marriage.
American Mathematical Monthly, 69:9–15, 1962.

[2] www.nrmp.org. Accessed 05-05-17.

[3] D.F. Manlove and G. O'Malley.
Student project allocation with preferences over projects.
Journal of Discrete Algorithms, 6:553–560, 2008.

[4] K. Iwama, S. Miyazaki, and H. Yanagisawa.
Improved approximation bounds for the student -project allocation with preferences over projects.
Journal of Discrete Algorithms, 13:59–66, 2012.

Contact Information

- Web: www.dcs.gla.ac.uk/∼sofiat
- Email: s.olaosebikan.1@research.gla.ac.uk

First year PhD student, supervised by David Manlove and Kitty Meeks. Supported by the College of Science and Engineering, University of Glasgow.

SICSA theme: Theory, Modelling and Computation.

University of Glasgow

60 YEARS OF COMPUTING AT GLASGOW