Efficient Algorithms for Matching Problems

Dr David Manlove

Department of Computing Science, University of Glasgow, Glasgow G12 8QQ
e-mail: davidm@dcs.gla.ac.uk

Funded by a Royal Society of Edinburgh / Scottish Executive Personal Research Fellowship

Abstract

In many practical contexts we seek to allocate applicants to posts using a centralised matching scheme

Typically we have:

- a set of *applicants* $a_1, a_2, \dots, a_n$
- a set of *posts* $p_1, p_2, \dots, p_m$
- applicants have preferences over posts
- posts may have preferences over applicants
- each post has a *capacity* (max no. of applicants it can take on)

This gives rise to a *matching problem*

The aim of this research is to explore the existence of *efficient algorithms* (computer programs) for solving matching problems

Example – allocating students to campus housing

5 students $s_1, s_2, \dots, s_5$ and 5 houses $h_1, h_2, \dots, h_5$
Each house has capacity 1 and preferences are as follows:

- $s_1 : h_1 \; h_2 \; h_3$
- $s_2 : h_1 \; h_2 \; h_3 \; h_4$
- $s_3 : h_1 \; h_2 \; h_3 \; h_4 \; h_5$
- $s_4 : h_1 \; h_3 \; h_5 \; h_4$
- $s_5 : h_2 \; h_5 \; h_4$

- student s_1 prefers house h_1 to house h_2, etc.
- student s_2 does not find house h_5 acceptable
- houses do not have preferences over students

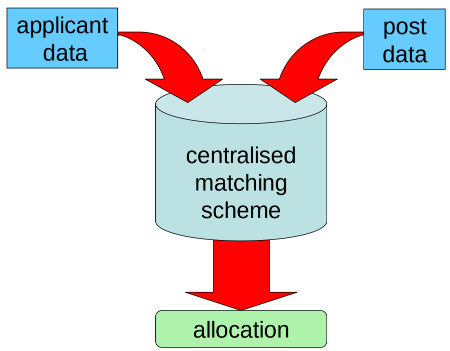
What is a *matching*?

1. Each student is allocated to at most one house
2. No house is allocated more students than its capacity
3. No student is allocated to an unacceptable house

What is a *greedy maximum matching*?

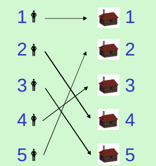
1. Match as many students to houses as possible
2. Subject to 1, match as many students to their 1st-choice house
3. Subject to 2, match as many students to their 2nd-choice house etc.

2: Pareto optimality



Example of a greedy maximum matching

- $s_1 : h_1 \; h_2 \; h_3$
- $s_2 : h_1 \; h_2 \; h_3 \; h_4$
- $s_3 : h_1 \; h_2 \; h_3 \; h_4 \; h_5$
- $s_4 : h_1 \; h_3 \; h_5 \; h_4$
- $s_5 : h_2 \; h_5 \; h_4 \; p_3$



i.e. every student is allocated to an acceptable house;
2 students have 1st-choice, 1 student has 2nd-choice,
1 student has 4th-choice, 1 student has 5th-choice

Other possibilities: (A) *generous maximum matching*

1. Match as many students to houses as possible
2. Subject to 1, match as few students to their r^{th}-choice house
3. Subject to 2, match as few students to their $(r-1)^{th}$-choice house etc.
   where r is the maximum length of a student's preference list

(B) *minimum cost maximum matching*

1. Match as many students to houses as possible
2. Subject to 1, minimise the sum of the ranks of the matched houses in the students' preference lists

Pareto optimal matchings

A matching M_1 is *Pareto optimal* if there is no matching M_2 such that:
(i) Some student is better off in M_2 than in M_1
(ii) No student is worse off in M_2 than in M_1

Example

- $s_1 : h_2 \; h_1$
- $s_2 : h_1 \; h_2$
- $s_3 : h_3$

This matching is not Pareto optimal since s_1 and s_2 could swap houses – then each would be better off

- Pareto optimal matchings have been the focus of much interest, particularly from the economics community
- Greedy, generous & min cost maximum matchings are all Pareto optimal

Pareto optimal matchings can have different sizes

- Example

$s_1 : h_1$
$s_2 : h_1 \; h_2$
size 1

$s_1 : h_1$
$s_2 : h_1 \; h_2$
size 2

Our main result

- Efficient algorithm for finding a Pareto optimal matching that has largest possible size

Future work

- Extend to the case where preference lists may include ties
- Extend to the case where houses may have capacity >1

More information

D.J. Abraham, K. Cechlárová, D.F. Manlove and K. Mehlhorn
Pareto optimality in house allocation problems
To appear in *Proceedings of ISAAC 2004: the 15th Annual International Symposium on Algorithms and Computation*
Lecture Notes in Computer Science, Springer-Verlag
Hong Kong, China, December 20-22, 2004

Acknowledgements

Aspects of the research described in both posters have been undertaken in collaboration with Rob Irving, Tamás Fleiner, Gregg O'Malley, Patrick Prosser and Sandy Scott