Fractional solutions for NTU-games

Péter Biró
Department of Computing Science
University of Glasgow
pbiro@dcs.gla.ac.uk

Tamás Fleiner
Department of Computer Science and Information Theory
Budapest University of Technology and Economics
fleiner@cs.bme.hu

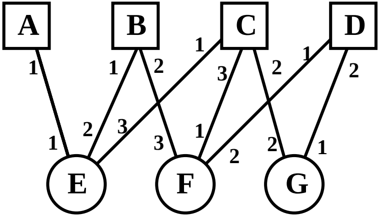
COMSOC, Düsseldorf
15 September 2010

University of Glasgow

Outline

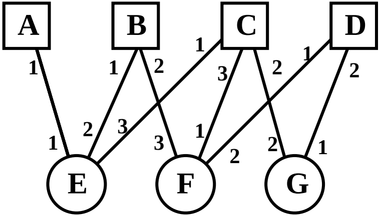
- Introduction of fractional core through stable matchings
- An example: matching with couples
- Finding stable allocations by Scarf's algorithm
- Experiments
- Open problems

Stable Marriage problem by Gale and Shapley (1962)



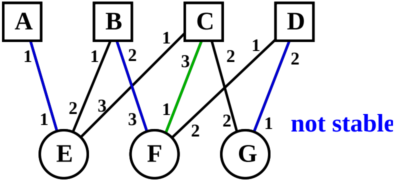
"Each person ranks those of the opposite sex in accordance with his or her preferences for a marriage partner."

Stable Marriage problem by Gale and Shapley (1962)



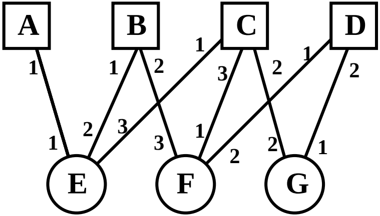
"Each person ranks those of the opposite sex in accordance with his or her preferences for a marriage partner."

A set of marriages is stable, if there is no "blocking pair": a man and a woman who are not married to each other but prefer each other to their actual mates.



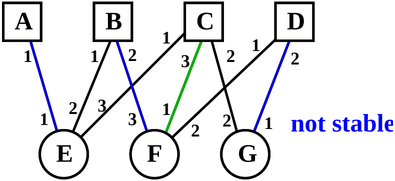
(C,F) blocking pair

Stable Marriage problem by Gale and Shapley (1962)



"Each person ranks those of the opposite sex in accordance with his or her preferences for a marriage partner."

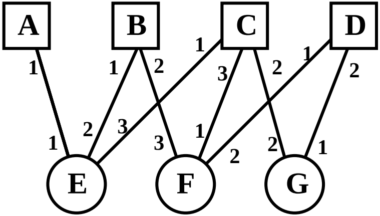
A set of marriages is stable, if there is no "blocking pair": a man and a woman who are not married to each other but prefer each other to their actual mates.



(C,F) blocking pair

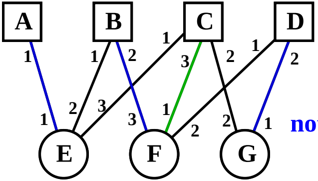
The Gale-Shapley algorithm finds a stable matching in $O(m)$ time.

Stable Marriage problem by Gale and Shapley (1962)



"Each person ranks those of the opposite sex in accordance with his or her preferences for a marriage partner."

A set of marriages is stable, if there is no "blocking pair": a man and a woman who are not married to each other but prefer each other to their actual mates.



not stable

(C,F) blocking pair

The Gale-Shapley algorithm finds a stable matching in $O(m)$ time.

set of stable matchings = core of the corresponding NTU-game

Stable (fractional) matchings vs (fractional) core

bipartite graph		
Marriage problem Gale-Shapley ‘62: ∃ stable matching		

Stable (fractional) matchings vs (fractional) core

bipartite graph	**nonbipartite graph**	
Marriage problem Gale-Shapley '62: $\exists$ stable matching	Roommates problem	

For every vertex v, let $<_v$ be a linear order on the edges incident with v. A weight-function $x : E(G) \to \{0, 1\}$ is a **matching** if $\sum_{v \in e} x(e) \leq 1$ for every $v \in V(G)$.

Stable (fractional) matchings vs (fractional) core

bipartite graph	**nonbipartite graph**	
Marriage problem Gale-Shapley '62: $\exists$ stable matching	Roommates problem	

For every vertex v, let $<_v$ be a linear order on the edges incident with v. A weight-function $x : E(G) \to \{0, 1\}$ is a **matching** if $\sum_{v \in e} x(e) \leq 1$ for every $v \in V(G)$.

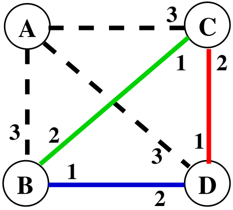
A matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = 1$.
(every non-matching edge is "dominated" at some vertex.)

Stable (fractional) matchings vs (fractional) core

bipartite graph	**nonbipartite graph**	
Marriage problem Gale-Shapley '62: $\exists$ stable matching	Roommates problem	

For every vertex v, let $<_v$ be a linear order on the edges incident with v. A weight-function $x : E(G) \to \{0,1\}$ is a **matching** if $\sum_{v \in e} x(e) \leq 1$ for every $v \in V(G)$.

A matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = 1$.



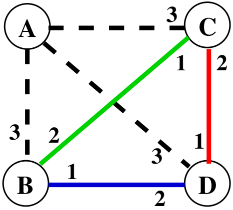
- Gale-Shapley (1962):
  Stable matching may not exist!

Stable (fractional) matchings vs (fractional) core

bipartite graph	**nonbipartite graph**	
Marriage problem Gale-Shapley '62: $\exists$ stable matching	Roommates problem	

For every vertex v, let $<_v$ be a linear order on the edges incident with v. A weight-function $x : E(G) \to \{0,1\}$ is a **matching** if $\sum_{v \in e} x(e) \leq 1$ for every $v \in V(G)$.

A matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = 1$.



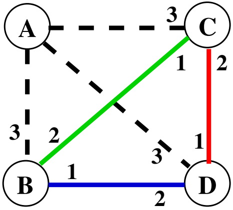
- Gale-Shapley (1962):
  Stable matching may not exist!
- Irving (1985): A stable matching can be found in $O(m)$ time, if one exists.

Stable (fractional) matchings vs (fractional) core

bipartite graph	**nonbipartite graph**	
Marriage problem Gale-Shapley '62: $\exists$ stable matching	Roommates problem Tan '90: $\exists$ stable half-matching	

For every vertex v, let $<_v$ be a linear order on the edges incident with v. A weight-function $x : E(G) \to \{0, 1\}$ is a **matching** if $\sum_{v \in e} x(e) \leq 1$ for every $v \in V(G)$.

A matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = 1$.



- Gale-Shapley (1962):
  Stable matching may not exist!
- Irving (1985): A stable matching can be found in $O(m)$ time, if one exists.
- Tan (1990): Stable **half-matching** always exists! i.e. $x(e) \in \{0, \frac{1}{2}, 1\}$.

Stable (fractional) matchings vs (fractional) core

bipartite graph	nonbipartite graph	**hypergraph**
Marriage problem Gale-Shapley '62: $\exists$ stable matching	Roommates problem Tan '90: $\exists$ stable half-matching	Coalition Formation Game Aharoni-Fleiner '03 (Scarf '67): $\exists$ stable fractional matching

For every vertex v, let $<_v$ be a linear order on the hyper-edges incident with v. A weight-function $x : E(G) \to \{0, 1\}$ is a **matching** if $\sum_{v \in e} x(e) \leq 1$ for every $v \in V(G)$.

A matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = 1$.

Aharoni-Fleiner (2003): Stable fractional matching always exists (i.e. $x(e) \in [0, 1]$) $\sim$ the **fractional core** of a CFG is nonempty.

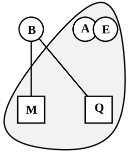
An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)

Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



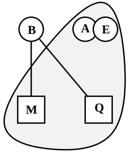
An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)

Couples can submit joint preference lists...

Applicants:	Bill	Adam and Eve
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam

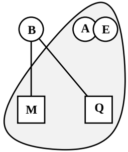


An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	Bill	Adam and Eve
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



An example for CFG: matching with couples

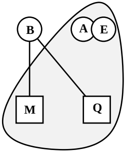
National Resident Matching Program (since 2009 in SFAS too)

Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill

ranking of **NY Memorial:** Bill, Adam

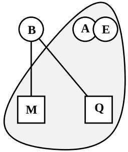


An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



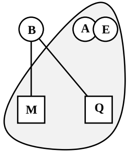
An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)

Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam

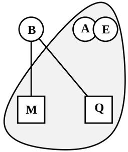


An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam

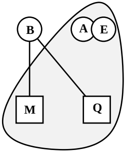


An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



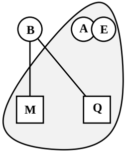
Roth (1984): Stable matching may not exist.

An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	Bill	Adam and Eve
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



Roth (1984): Stable matching may not exist.

Ronn (1990): The related decision problem is NP-complete.

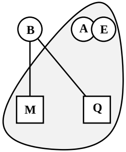
Biró-Irving (2010): NP-complete even for master lists.

An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	**Bill**	**Adam and Eve**
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



Roth (1984): Stable matching may not exist.

Ronn (1990): The related decision problem is NP-complete.

Biró-Irving (2010): NP-complete even for master lists.

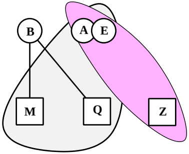
But what is the meaning of a fractional solution?

An example for CFG: matching with couples

National Resident Matching Program (since 2009 in SFAS too)
Couples can submit joint preference lists...

Applicants:	Bill	Adam and Eve
1st choice:	Queens	(Memorial, Queens)
2nd choice:	Memorial	

ranking of **NY Queens:** Eve, Bill
ranking of **NY Memorial:** Bill, Adam



Roth (1984): Stable matching may not exist.

Ronn (1990): The related decision problem is NP-complete.

Biró-Irving (2010): NP-complete even for master lists.

But what is the meaning of a fractional solution?

Stable b-matchings: agents with capacities

bipartite graph		
College Admissions Gale-Shapley '62: $\exists$ stable matching		

Let $b : V(G) \to \mathbb{Z}_+$ be vertex-bounds.
A weight-function $x : E(G) \to \{0, 1\}$ is a **(b)-matching** if
$\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

Stable b-matchings: agents with capacities

bipartite graph		
College Admissions Gale-Shapley '62: $\exists$ stable matching		

Let $b : V(G) \to \mathbb{Z}_+$ be vertex-bounds.
A weight-function $x : E(G) \to \{0, 1\}$ is a **(b)-matching** if $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

A b-matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.
(every non-matching edge is "dominated" at some vertex.)

Stable b-matchings: agents with capacities

bipartite graph	**nonbipartite graph**	
College Admissions Gale-Shapley '62: $\exists$ stable matching	Stable Fixtures Biró-Fleiner '03: $\exists$ stable half-matching	

Let $b : V(G) \to \mathbb{Z}_+$ be vertex-bounds.
A weight-function $x : E(G) \to \{0, 1\}$ is a **(b)-matching** if $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

A b-matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.

Biró-Fleiner (2003): A stable half-matching can be found efficiently for nonbipartite graphs.

Cechlárová-Fleiner (2005), Irving-Scott (2007): A stable (b-)matching can be found in $O(m)$ time, if one exists ("Stable Multiple Activities" or "Stable Fixtures").

Stable b-matchings: agents with capacities

bipartite graph	nonbipartite graph	**hypergraph**
College Admissions Gale-Shapley '62: $\exists$ stable matching	Stable Fixtures Biró-Fleiner '03: $\exists$ stable half-matching	CFG with agent-capacities Biró-Fleiner '10 (Scarf '67): $\exists$ stable fractional matching

Let $b : V(G) \to \mathbb{Z}_+$ be vertex-bounds.
A weight-function $x : E(G) \to \{0, 1\}$ is a **(b)-matching** if $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

A b-matching is **stable** if for every $e \in E(G)$, either $x(e) = 1$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.

Biró-Fleiner (2010): A stable fractional matching can be found by Scarf's algorithm for hypergraphs.

Stable Allocations: cooperations with capacities

bipartite graph		
Part-time jobs		

Beside the vertex-bounds, let $c : E(G) \to \mathbb{R}_+$ be edge-capacities. A weight-function $x : E(G) \to \mathbb{R}_+$ is an **allocation** if $x(e) \leq c(e)$ for every $e \in E(G)$ and $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

Stable Allocations: cooperations with capacities

bipartite graph		
Part-time jobs		

Beside the vertex-bounds, let $c : E(G) \to \mathbb{R}_+$ be edge-capacities.
A weight-function $x : E(G) \to \mathbb{R}_+$ is an **allocation** if $x(e) \leq c(e)$ for every $e \in E(G)$ and $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

An allocation is **stable** if for every $e \in E(G)$, either $x(e) = c(e)$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.
(every "non-saturated" edge is "dominated" at some vertex.)

Stable Allocations: cooperations with capacities

bipartite graph		
Part-time jobs Baïou-Balinski '02: $\exists$ integral stable allocation		

Beside the vertex-bounds, let $c : E(G) \to \mathbb{R}_+$ be edge-capacities. A weight-function $x : E(G) \to \mathbb{R}_+$ is an **allocation** if $x(e) \leq c(e)$ for every $e \in E(G)$ and $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

An allocation is **stable** if for every $e \in E(G)$, either $x(e) = c(e)$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.

Baïou-Balinski (2002): An integral stable allocation can be found in $O(m^2)$ time for **bipartite graphs**.

Stable Allocations: cooperations with capacities

bipartite graph	**nonbipartite graph**	
Part-time jobs Baïou-Balinski '02: $\exists$ integral stable allocation	P2P networks B-F '10, D-M '10: $\exists$ half-integral stable allocation	

Beside the vertex-bounds, let $c : E(G) \to \mathbb{R}_+$ be edge-capacities. A weight-function $x : E(G) \to \mathbb{R}_+$ is an **allocation** if $x(e) \leq c(e)$ for every $e \in E(G)$ and $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

An allocation is **stable** if for every $e \in E(G)$, either $x(e) = c(e)$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.

Biró-Fleiner (2010): A half-integral stable allocation can be found in $O(m^3 \log B)$ time for the integral stable allocation problem on **nonbipartite graphs**, where B is the maximal vertex-bound.

Dean-Munshi (2010): A half-integral stable allocation can be found in $O(m \log n)$ time with high probablity.

Stable Allocations: cooperations with capacities

bipartite graph	nonbipartite graph	**hypergraph**
Part-time jobs Baïou-Balinski '02: $\exists$ integral stable allocation	P2P networks B-F '10, D-M '10: $\exists$ half-integral stable allocation	CFG with capacities Biró-Fleiner '10 (Scarf '67): $\exists$ stable allocation

Beside the vertex-bounds, let $c : E(G) \to \mathbb{R}_+$ be edge-capacities.
A weight-function $x : E(G) \to \mathbb{R}_+$ is an **allocation** if $x(e) \leq c(e)$ for every $e \in E(G)$ and $\sum_{v \in e} x(e) \leq b(v)$ for every $v \in V(G)$.

An allocation is **stable** if for every $e \in E(G)$, either $x(e) = c(e)$, or there is a vertex $v \in e$ s.t. $\sum_{e \leq_v f} x(f) = b(v)$.

Biró-Fleiner (2010): A stable allocation can be found by Scarf's algorithm for hypergraphs.

Some experiments with couples

in NRMP and SFAS...


Applications Places System
11:20 AM
National Resident Matching Program - Konqueror
Location Edit View Go Bookmarks Tools Settings Window Help
Location: http://www.nrmp.org/
Google Search
Microsoft Outlook Web Access
Google
research (2)
Péter Biró's home page
marriage
eu
others
people
conferences
kidney
college
kereso
newconference
HOME | CONTACT
NRMP
National Resident Matching Program
Residency Match
Fellowship Matches
About NRMP
Register/Login
Search
The National Resident Matching Program (NRMP) is a private, not-for-profit corporation established in 1952 to provide a uniform date of appointment to positions in graduate medical education (GME) in the United States.
News from the NRMP!
New>NRMP TO IMPLEMENT MATCH WEEK CHANGES
The NRMP Board of Directors has voted to proceed with changes to Match Week 2012. A new Supplemental Offer and Acceptance Program will be implemented for unmatched applicants and unfilled programs.
New>LARYNGOLOGY JOINS THE NRMP!
The NRMP is pleased to welcome Laryngology as a new fellowship match for the 2012 appointment year. Sponsored by the American Laryngological Association (ALA), the Laryngology Fellowship Match will open for registration on September 29, 2010 with Match Day on February 2, 2011. For more information about the Laryngology Fellowship Match, including the Schedule of Dates, click on Fellowship Matches at the top of this page or contact our Helpdesk Specialists toll free at 1-866-617-5834.
MEDICAL GENETICS JOINS THE NRMP
To participate in a NRMP match, click Register/Login above.
Main Residency Match
Registration for the 2011 Match opens on August 15th for applicants and September 1st for institutions and programs.
The 2010 Main Residency Match was the largest in NRMP history, encompassing more than 37,000 applicants, 4,100 graduate medical education programs, and 25,500 residency training positions. For more information, read the press release and listen to an interview with NRMP Executive Director Mona M. Signer.
Communications
Visit the Communications page for more information about and access to recent NRMP web conferences and webcasts.
Data and Reports
Visit the Data and Reports section for recent reports and historic NRMP match data.
New>Results of the 2010 NRMP Program Director Survey (PDF, 164 pages) This report presents the results of selected items from the 2010 NRMP Program Director Survey. Data are reported for 19 specialties and include: (1) factors
Page loaded.
[Inbo... [Te... [Te... *sca... scarf... Ter... [CO... [MA... [Edit... [Betti... [sfas... [H:/... hype... Fract... Natio...

NRMP: Independent Applicants - Konqueror

http://www.nrmp.org/res_match/special_part/ind_app/couples.html

members of the couple must be active applicants in the Match.

Step 1

Each partner should first arrange an individual preference list on separate sheets of paper. In the example, the letters refer to a specific program in a particular hospital in that city.

Partner I	Partner II
1) New York City - A	1) Chicago - X
2) Chicago - A	2) Chicago - Y
3) Evanston - B	3) Boston - X
4) Los Angeles - A	4) Chicago - Z
5) New York City - B	5) New York City - X
	6) New York City - Y

Step 2

Next, both partners must decide together how to prepare their lists as pairs of programs. For example, they could consider all the possible pairings where the hospital programs are in the same general location, as indicated in the list below. In some cases one rank in the pair may be designated "No Match" to indicate that one partner is willing to go unmatched if the other is matched to a position. Note that the list below is not necessarily in the order that will eventually be submitted.

Partner I	Partner II
New York City - A	New York City -X
New York City - A	New York City -Y
Chicago - A	Chicago -X
Chicago -A	Chicago -Y
Chicago -A	Chicago -Z
Evanston -B	Chicago -X
Evanston -B	Chicago -Y
Evanston -B	Chicago -Z
New York City -B	New York City -X
New York City -B	New York City -Y
New York City -A	No Match

Page loaded.

The Redesign of the Matching Market for American Physicians: Some Engineering Aspects of Economic Design

By ALVIN E. ROTH AND ELLIOTT PERANSON*

We report on the design of the new clearinghouse adopted by the National Resident Matching Program, which annually fills approximately 20,000 jobs for new physicians. Because the market has complementarities between applicants and between positions, the theory of simple matching markets does not apply directly. However, computational experiments show the theory provides good approximations. Furthermore, the set of stable matchings, and the opportunities for strategic manipulation, are surprisingly small. A new kind of "core convergence" result explains this; that each applicant interviews only a small fraction of available positions is important. We also describe engineering aspects of the design process. (JEL C78, B41, J44)

The entry-level labor market for new physicians in the United States is organized via a centralized clearinghouse called the National Resident Matching Program (NRMP). Each year, approximately 20,000 jobs are filled in a process in which graduating physicians and other applicants interview at residency programs throughout the country and then compose and submit Rank Order Lists (ROLs) to the NRMP, each indicating an applicant's prefer-

employment, rather than waiting to participate in the larger market. (By the 1940's, contracts were typically being signed two years in advance of employment.) Although the matching algorithm has been adapted over time to meet changes in the structure of medical employment, roughly the same form of clearinghouse market mechanism has been used since 1951 (see Roth, 1984). The kind of market failure that gave rise to this clearinghouse has since been

Applications Places System 11:23 AM

SFAS-About Us - Konqueror

Location Edit View Go Bookmarks Tools Settings Window Help

Location: http://www.nes.scot.nhs.uk/sfas/About/default.asp

Microsoft Outlook Web Access Google research (2) Péter Biró's home page marriage eu others people conferences kidney college kereso newconference

NHS Education for Scotland (NES)

NHS SCOTLAND

SCOTTISH FOUNDATION ALLOCATION SCHEME (SFAS)

NAVIGATION

- **About SFAS**
- Prog. by Region
- How to Apply
- Terms & Conditions
- Help
- Comments
- Open Days
- Useful Links
- Disclaimer

- Back to NES Home

Home > Sfas > About

SFAS 2010 Timetable - click to view

Post Compatibility (xls) - click to view (Compatible Posts are marked in Blue) for linked applications

Scottish Scoring Comparison Chart (xls)

The SFAS matching scheme

The SFAS Matching Scheme uses a computer program that aims to produce a matching that best satisfies the applicants' preferences. The algorithm that underlies this program was developed in the Department of Computing Science at the University of Glasgow, and is based on state-of-the-art research into optimal matching.

Introduction

The matching algorithm takes account of the following factors:

- the number of places in each programme
- the preference list of each individual applicant
- the score of each applicant
- which pairs of applicants are linked
- the compatibility information on programmes (from the viewpoint of linked applicants).

The algorithm is complicated by the need to deal with linked pairs in a fair way, giving them neither an advantage nor a disadvantage over single applicants, and ensuring that, if they are matched, then it is to compatible programmes. The description below is initially in terms of single applicants, and then an indication is given of the adaptations needed to accommodate linked pairs of applicants.

The algorithm – main idea

The first step is a *tie-breaking* step in which applicants with equal scores are randomly ordered. In effect, each applicant is given a unique score, but if applicant *a* had a higher original score than applicant *b* this will still be true for the revised scores.

The main body of the algorithm can be viewed as a sequence of attempts to match an applicant to a programme. At any point during the progress of the algorithm, an applicant is either matched (at least temporarily) or unmatched. Initially, each applicant's *best achievable* preference is the first entry on his/her preference list. At each step of the algorithm, a *random* applicant is chosen from those who are unmatched, and an attempt is made to match this applicant to his/her best achievable preference. If the programme has at least one free place then the match is accepted. Otherwise, the match is only accepted if a lower scoring applicant can be displaced from the programme – in this case the assigned applicant with the lowest score is displaced; if not the match is rejected. A rejection, or a displacement, results in the best achievable preference being advanced by one position in the list of the applicant concerned. The process terminates when each applicant is either matched or has been rejected by, or displaced from, all of the programmes on his/her preference list.

The resulting matching has the crucial *stability* property, namely:

- there can be no applicant *a* who would prefer to be matched to programme *p*, and at the same time *p* has an unfilled place or an assigned

[Inbo... [Te... [Te... *sca... scarf... Ter... [CO... [MA... [Edit... [Betti... [sfas... [H:/... hype... Fract... SFAS...

The algorithm is complicated by the need to deal with linked pairs in a fair way, giving them neither an advantage nor a disadvantage over single applicants, and ensuring that, if they are matched, then it is to compatible programmes. The description below is initially in terms of single applicants, and then an indication is given of the adaptations needed to accommodate linked pairs of applicants.

The algorithm – main idea

The first step is a *tie-breaking* step in which applicants with equal scores are randomly ordered. In effect, each applicant is given a unique score, but if applicant *a* had a higher original score than applicant *b* this will still be true for the revised scores.

The main body of the algorithm can be viewed as a sequence of attempts to match an applicant to a programme. At any point during the progress of the algorithm, an applicant is either matched (at least temporarily) or unmatched. Initially, each applicant's *best achievable* preference is the first entry on his/her preference list. At each step of the algorithm, a *random* applicant is chosen from those who are unmatched, and an attempt is made to match this applicant to his/her best achievable preference. If the programme has at least one free place then the match is accepted. Otherwise, the match is only accepted if a lower scoring applicant can be displaced from the programme – in this case the assigned applicant with the lowest score is displaced; if not the match is rejected. A rejection, or a displacement, results in the best achievable preference being advanced by one position in the list of the applicant concerned. The process terminates when each applicant is either matched or has been rejected by, or displaced from, all of the programmes on his/her preference list.

The resulting matching has the crucial *stability* property, namely:

- there can be no applicant *a* who would prefer to be matched to programme *p*, and at the same time *p* has an unfilled place or an assigned applicant with a lower score than *a*.

In other words, no private 'deal' could be made by an applicant and a programme that would be to the benefit of both.

Linked applicants

To accommodate linked applicants, a *joint* preference list is formed for each such pair, using their individual preference lists and the programme compatibility information. If such a pair, *a* and *b*, have individual preferences *p*1, *p*2, . . . , *p*10 and *q*1, *q*2, . . . , *q*10 respectively (with *a* the higher scoring applicant), then the joint preference list of the pair (*a*,*b*) is (*p*1,*q*1), (*p*1,*q*2), (*p*2,*q*1), (*p*2,*q*2), (*p*1,*q*3), (*p*3,*q*1), (*p*2,*q*3), (*p*3,*q*2), . . ., (*p*9,*q*10), (*p*10,*q*9), (*p*10,*q*10) (except that incompatible pairs of programmes are omitted)

In the main body of the algorithm, the members of a linked pair are handled together, so the match of the pair (*a*,*b*) to the programmes (*p*,*q*) will be accepted only if each of these programmes either has an unfilled place or a lower scoring applicant who can be displaced. A complication arises when one member *x* of a linked pair has to be withdrawn from a programme *p* because his/her partner was displaced from their current assigned programme. In this case, some other applicants may have been rejected by *p* because of the presence of *x*, and any such applicant *a* must be withdrawn from their current programme, if any, and have their best achievable preference reset to *p*. (A corresponding, but more complex reset operation is needed if *a* is a member of a linked pair). This reset operation thereby allows a further opportunity for applicant *a* to be matched to programme *p*.

The algorithm terminates when every single applicant and linked pair is either matched or has been rejected by, or displaced from, every entry in their preference list with no possibility of reconsideration by a programme that has had a withdrawal.

The final matching is stable for single applicants, as before, but also for linked pairs, in the sense that:

- there can be no linked pair (*a*,*b*) of applicants who would prefer to be matched to compatible programmes (*p*,*q*), and at the same time, each of *p* and *q* has an unfilled place or an assigned applicant with a lower score than *a* and *b* respectively.

Frequently Asked Questions

back to top

Applications Places System 11:26 AM

H:/Research/mypapers/Couples/sfas_couples.dvi

File Edit View Go Help

Previous Next 2 of 26 200%

Stable matching with couples – an empirical study

Péter Biró*,† and Robert W. Irving*

School of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK.
Email: {pbiro,rwi}@dcs.gla.ac.uk.

Abstract

In practical applications, algorithms for the classical version of the Hospitals Residents problem (the many-one version of the Stable Marriage problem) may have to be extended to accommodate the needs of couples who wish to be allocated to (geographically) compatible places. Such an extension has been in operation in the NRMP matching scheme in the US for a number of years. In this setting, a stable matching need not exist, and it is an NP-complete problem to decide if one does. However, the only previous empirical study in this context (focused on the NRMP algorithm), together with information from NRMP, suggest that, in practice, stable matchings do exist and that an appropriate heuristic can be used to find such a matching.

The study presented here was motivated by the recent decision to accommodate couples in the Scottish Foundation Allocation Scheme (SFAS), the Scottish equivalent of the NRMP. Here, the problem is a special case, since hospital preferences are derived from a 'master list' of resident scores, but we show that the existence problem remains NP-complete in this case. We describe the algorithm used in SFAS, and contrast it with a version of the algorithm that forms the basis of the NRMP approach. We present an empirical study of the performance of a number of variants of these algorithms, and of a third simpler algorithm based on satisfying blocking pairs, using a range of data sets. The results indicate that, not surprisingly, increasing the ratio of couples to single applicants typically makes it harder to find a stable matching (and, by inference, less likely that a stable matching exists). However, the likelihood of the algorithm finding a stable matching is very high for realistic values of this ratio, and especially so for particular variants of the algorithms.

1 Introduction

[Inbox ... [Term... [Term... scarf.t... scarf_pr Terminal [COMS... [MATL... [Editor... [Betti... [sfas... H:/Res... [SFAS-...

H:/Research/mypapers/Couples/sfas_couples.dvi

Algorithm	Number of couples: 2	5	10	15	20	25	30	35	40	45	50
C-RAN	981	965	909	870	827	801	740	648	604	529	453
C-STA	978	937	831	753	676	640	605	531	508	470	407
C-SGL	981	962	907	862	822	801	753	685	627	545	446
C-CPL	974	927	821	758	712	681	646	586	554	506	451
C-RLP	968	920	825	708	555	424	288	193	136	84	49
BB-RAN	**983**	**966**	**916**	**882**	**851**	**829**	772	701	621	530	430
BB-SCO	968	922	819	722	604	527	444	306	248	172	107
BB-USE	982	962	911	872	839	816	**773**	**705**	**662**	**591**	**507**
BB-USS	968	929	863	805	751	714	686	659	647	582	**507**
BB-SGL	968	931	864	819	779	749	716	699	654	553	429
BB-CPL	981	952	843	687	563	496	410	344	325	329	425
RP-RAN	929	841	704	601	501	411	384	353	274	256	228
RP-SGL	975	925	796	705	613	536	477	394	336	280	211
RP-CPL	917	843	693	586	489	405	358	304	266	222	220
Total	984	967	921	888	861	848	825	793	769	728	672

Table 1: Instances of size 100 (1 second per instance)

Some experiments with couples (100 applicants)

	Number of couples					
Algorithm	2	5	10	15	20	25
Roth-Perantson approach	975	925	796	705	613	536
Best heuristic of Biró-Irving	983	966	916	882	851	826
Scarf (integral solution)	930	838	670	562	483	387
Scarf half-intergral solution	999	991	966	944	902	851
Scarf fractional solution	70	162	330	438	517	613
Av. # of fractional weights	3.4	3.55	3.91	4.27	4.37	4.73
# of fractional weights = 1	27	52	87	104	125	132
# of fractional weights = 2	13	31	58	71	79	91
# of fractional weights = 3	4	9	25	32	51	51
# of fractional weights = 4	5	20	40	64	58	61

Open questions

What is the

- meaning of a fractional solution?
- running time of the Scarf algorithm?
- complexity of the problem of finding a fractional core element?

... for special families of NTU-games?

Further applications?