Algorithmic Dynamic Matching

Detailing Marketplace Pilot for US Navy

Ian Shaw, LTJG, USN [*] Zac Dannelly, LTJG, USN [†]
Drew Calcagno, LTJG, USN [‡] Richard Kuzma, LTJG, USN [§]
Brian Yarborough, 1LT, USAF [¶]

March 8, 2019

Abstract

In this project we apply Alvin Roth's work in mixed integer programming solution to the college admissions problem to the placement of U.S. Navy sailors to jobs. We further the body of work by providing alternate formulation to the integer program and metrics that are gleaned from the expressed ordered preferences of both the sailors for jobs and the job owners preferences over sailors. These metrics are: specialization, competitiveness, similarity, and preference correlation. Though our presentation is tailored to meet the stated fiscal year 2019 goals of the Chief of Naval Personnel, the work is agnostic and can be applied to any matching market not requiring stability as market participants can be compelled. [1]

[*] E stuff
[†] Z stuff
[‡] X Stuff
[§] Q Stuff
[¶] Brian Stuff

[1] The code to demonstrate the matching algorithms, optimization, and preference-based metrics can be found in Ian Shaw's Github Repository `Dynamic_Manning`. https://github.com/ieshaw/Dynamic_Manning

Contents

1 Introduction

Under the backdrop of the President of the United States issuing the Executive Order on Maintaining American Leadership in Artificial Intelligence in February 2019, we created the algorithmic foundation for advancement of artificial intelligence and machine learning (AI/ML) in military talent allocation. Before the US Government can accomplish its both lethal and non-lethal AI/ML goals, however, it must embrace the essential building blocks underlying the buzzwords. One of those cornerstones is algorithmic linear programing, specifically the use of mixed integer programming (MIP). In this piece, we reveal the most exquisite MIP algorithm completed thus far for the US Navy.

We believe the world to be arcing toward the need for humans to grow as increasingly specialized workers, and in the vicious games of statecraft and war, allocating talent with hyper specificity will become a necessity for victory. To that end, we theorized, coded, and prototyped an MIP algorithm that matches job seekers (sailors) and job owners (commanding officers).

1.1 Problem Description

Human talent allocation is a resource intensive process for an institution. This paper outlines the mathematics necessary to lay the groundwork for such a task within one of the most complex institutions of the American psyche – the Department of Defense (DoD). We dove into this problem with a specific service branch in mind. Focusing our work on the Navy, and within it, the sailors assigned to the National Security Agency (NSA) for cryptologic and cyber warfare efforts, we honed our test case to a group of officers most aligned with technical advancement. Furthermore, despite the small prototype sample, the algorithm is greatly scalable.

Market forces appear are largely the drivers behind a job marketplace – this concept is not new for the American economy. However, given a captive audience such as service members ordered to specific jobs, the mathematical benefit of benevolent autocracy is immense. Rather than the suboptimal fate to which current servicemembers are subjected, the DoD is shopping for a better way forward.

In the Navy, terminology of a detailing marketplace refers to the attempts to detail, or place, warfighters into their next job. This detailing marketplace is a system by which military members can transparently rank their job preferences and the people who own those jobs can supply their preference of incoming personnel. This problem is rather unique to the military due to the aforementioned captive market of many members obligated to remain in service due to contract, desiring to stay in for a pension, wishing to stay in through a sense of service, complemented by an inability for lateral entry – almost all members needing to start from the entry level. Additionally, servicemembers change jobs nearly every one to three years.

There are ontological parallels to other communities, too. This process is shared in some regards by the medical school graduates applying to the residency stage of their training. These graduates are applying to a pool of US residency programs. This pool is also rather narrow, mostly coming from US based medical schools, and the specificity of skill set required for success is better understood than success as a military officer.

This general similarity is the reason our initial matching algorithm (Gale-Shapely Deferred Acceptance Algorithm) is similar one used by that process, the National Residency Match Program. The Gale-Shapely algorithm's application to the residency matching problem earned the Nobel Prize in 2012.

This paper also proposes an optimization based solution to the matching process. The optimization is not found in the National Residency Match Program as medical students have the ability to reject their assigned position, a choice not always given to military members. Thus an optimization can often more optimal solution for the DoD system at the expense of a few forced members.

Due to the importance of co-locating dual-military households (where two family members are in the military), we focus our matching algorithm and optimization proposals on solutions that would guarantee 95% or greater co-location rate.

Acknowledging the difficulty of wrangling disparate and dated personnel data, this paper also explores helpful metrics that can be gleaned simply from submitted, ordered preferences by job seekers and job owners. These are competitiveness, similarity, generalism, and specialization. Interestingly, due to the fact that preferences are expressed on job seekers and the jobs themselves, these metrics can be developed about the jobs or the job seekers.

Further the paper ends with suggested metrics that would be gleaned if personnel data beyond preferences was accessible, clean, and structured. These include a similarity measure based on quality encodings and a suggested ordering of possible jobs or applicants. The latter is proposed to be enabled by deep learning (the underlying technology of Artificial Intelligence (AI)). The suggested ordering would not make decisions on placement, but rather provide job seekers and job owners with metrics distilling the vast amount of information about

2 Literature Review

The most important people in this field, and the winners of the 2012 Nobel Prize in Economics for their work in stable marriage matching, are Roth [2] and Shapely [3]. We also draw much from Roth's former advisee, now University of Chicago's Booth Business School faculty economist Eric Budish [4].

1962

The intent of this algorithm is to provide stable pairings between job owners and job seekers based on their ranked preferences. The algorithm's initial conception and definition of stability can be found in Gale and Shapely's 1962 publication in the January *The American Mathematical Monthly* [4]. The algorithm completes in polynomial time and was originally written for application in collage admissions.

1982

Roth explores the incentives of conveying true preferences and whether is it in everyone's best interest to do so. [6] In his work he specifically points to the applications to "civil servants with civil service positions".

1984

Roth explores the stable marriage problem specifically in the terms of 'firms and workers', specifically that "The stable outcome which is best for all the firms is shown to be worst for all the workers, and vice-versa". [7]

1985

Roth explores the stable marriage problem specifically in the terms of 'firms and workers', also calling upon the lens of game theory. [9] He discussed an extension of the model from one assignment for each worker or firm, to multiple workers for each firm, to a situation where each firm can have multiple workers and each worker could have multiple firms. He also explored, under the constraint of stability, how in each model the optimal assignment set for one party (eg: firms) is the least optimal for the other (eg: workers). He elaborates that this final phenomenon creates difficulty in the institutional decision of how to formulate the matching algorithm.

Later that year Roth goes on to prove that in fact the collage admissions problem is a generalization of the stable marriage problem, but in fact not the same. This is due to the fact that colleges need to express preferences over groups of students, not just the students themselves. [8]

1988

Roth and Sotomayor investigate the two-sided matching market further, specifically on interior points rather than the well-researched extrema. [10]

1989

Irving explored indifference preferences and the follow on adoption for the Gale-Shapely algorithm. [5] This provides the theoretical framework allowing for indifference in our own formulation. Though much of

[2] http://stanford.edu/~alroth/PapersPDF.html
[3] http://www.econ.ucla.edu/shapley/ShapleyBiblio.1.html
[4] https://faculty.chicagobooth.edu/eric.budish/

his focus is on differing forms of stability (weak, strong, and super) these lie outside of our investigation due to the Navy's authority to compel its members to placement.

Roth and Sotomayor together, again, revisit the collage admissions. [11] They prove several aspects of the stability of this problem, such as that a solution is group stable if and only if it is stable (in the individual sense).

1993

Roth, Rothblum, and Vande Vate explored the concept of partial matches, discovering in fact this forms a lattice of solutions as well. These fractional matches could represent lotteries or time splitting. [2]

1994

Khuller et.al have developed an algorithm for stable matching on-line (matching people as they enter the system), as opposed to the typical formulation of having complete market participants and preferences at the time of matching. [12] This could be interesting in future work of understanding the Navy detailing process as a continuous, rather than discrete, process.

3 Algorithm

A basic algorithm, like Gale-Shapelys set of instructions for stable marriages, is an important first step in talent allocation. We moved onto a linear programming concept, oriented around solvers – algorithms that are adaptable. These solvers have varying types like linear, convex, and mixed integer. Our group followed a path paved by Stanford operations researcher Alvin Roth and adapted a mixed integer solution to leverage its flexibility. While Gale-Shapelys algorithm can be solved in n-squared time, ours is n-p-complete, indicating polynomial time completion. What we lose in speed, we gain in the ability to alter the formulation without total overhaul. These qualities make the approach particularly conducive to constraint creation, a vital requirement for stakeholders utilizing our solution.

Ours is a unique subset of mixed integer programming (MIP), binary optimization, to reflect the nature of either placement in a specific job, or not (i.e. only 1 or 0). The resulting lattice has an incredibly large number of dimensions manifesting every potential job placement for every individual. Subsequently, we apply constraints – the inherent reason we utilize MIP. These constraints cut away pieces of the lattice to reveal a viable search space to run the minimization function. Finding the minimum point within that constrained lattice provides us with a matrix of the optimally matched jobs and sailors.

The constraints we apply are the same as the college admissions problem [9], where each job can fill up to but not exceed the allocated number of positions and each sailor can only be assigned to one position. The explicit program can be found in Appendix B.

3.1 Strategy Proofness

4 Metrics

4.1 Pre-Match Metrics

4.1.1 Competitiveness

Definition

Competitiveness measures the relative desirability of a given sailor or job based on the expressed preference ranking of the other party. We wanted to consider a more sophisticated way of determining this metric beyond an average ranking for either the sailor or job as it is very susceptible to a right tail bias. For example, a position ranked 1st by ten individuals and hundredth by twenty individuals has the same average ranking as a job ranked seventh by all thirty individuals; yet the prior is much more competitive. To attempt to compensate for this, we adjust the average by a power of one half is to lessen the impact of very low preferences thereby weighting favorable preference more in our score consideration. In order to generalize the competitiveness score we scale the resulting average by the total number of jobs or sailors within the system. This allows for competitiveness to range from 1 being most competitive, and 0 being not competitive at all.

$$F_j = 1 - \frac{a_j}{M} \qquad \texttt{\% of openings job } j \texttt{ is of all openings in the Navy} \tag{1}$$

$$S_i^S = 1 - \frac{1}{nm}\sum_j \sqrt{P_{ij}^O} \qquad \texttt{sought after metric} \tag{2}$$

$$C_i^S = S_i^S \qquad \texttt{competitiveness of a job seeker} \tag{3}$$

$$C_j^O = F_j S_j^O \qquad \texttt{competitiveness of a job} \tag{4}$$

$$\tag{5}$$

To follow the development of this metric, more details are in Appendix (C).

Impact

Competitiveness will be used to rapidly identify the most attractive candidates. (E.g. Sailor A has a 0.98 competitiveness score and is therefore a top ranked candidate amongst all job owners.) These candidates would be good fits in many jobs and therefore could be considered ideal individuals to screen for command. On the job side, a low competitiveness would be a great way to decide on incentive structures. These jobs are those that very few sailors prefer and therefore either promotion based assignment or additional compensation could to targeted towards these positions.

4.2 Specialization

Definition

Specialization measures the extent to which the maximum desire for a sailor or job differs from the mean. Sailor A is highly specialized if one job owner highly desires them much more than the mean job owner (e.g. Sailor A is the 1st preference choice of Job Owner X, but is on average the 21st ranked preference). A sailor is a specialist to the extent that their skill set will serve a specific function well, with a comparative advantage to their peers and with a disproportionate positive contribution to the Department of Defense compared to the other roles they are qualified for.

$$\texttt{Specialization}_i^S = \frac{\mu(P_i^O) - \min\{P_i^O\}}{\mu(P_i^O) + \min\{P_i^O\}}$$

Impact

A sailor with a high specialization has some skill set that will serve a specific job function well; the seeker has an advantage over peers for a single role compared to the other roles the seeker is qualified for. If most job owners rate a seeker with low preference but a single owner rates that seeker with a high preference, the marginal gains of that seeker being matched with that owner are high. Matching the sailor with this particular role utilizes their specialization and produces a disproportionate positive contribution to the Department of the Navy. Intuitively, the matching algorithm will favor creating high-specialization matches. Failing to make these matches has a more negative impact on the system than failing to make non-specialized matches, because the usefulness of the sailor goes down significantly between their optimal match and average matches. This metric may be useful in the future to examine if very particular aspects of jobs or sailors make them attractive to small subsets of the other population.

4.3 Preference Correlation

Definition

Preference correlation is the R-squared score of job owner and job seeker preferences. A strong positive correlation indicates that job owners and seekers mutually desire each other.

[example plot + line...strong correlation]

[example plot + line...weak correlation]

Impact

A lack of correlation indicates an information gap between job seekers and owners; seekers may not understand what skills a particular job requires or job owners may lack an understanding of what skills would be beneficial to have on their team.

4.3.1 Post-Match Metrics

Preference Allocation Windows

Preference allocation indicates how many individuals received their top preference, how many individuals got a top three, top five, and top 10 preference, and how many failed to be matched with any preference.

Talent Distribution

Some job owners have multiple jobs. Ideally, such job owners would get an approximately equal set of preferences; one owner with five job slots would not get their 1st-5th choices while another job owner with five job slots gets their 20th-25th preferences.

Preference Difference Gap

This metric helps to illuminate the separation between job owner and sailor preferences. This can be an indicator for senior leaders that neither sailors nor job owners had their preferences unduly weighted within the optimization system.

Pareto Optimality

5 Pilot

5.1 Overview

The essential effort of this piece is to demonstrate the mathematical underpinnings of algorithmic matching for a defense application. That pilot application, however, was a very human process. The most difficult aspects were hardly the mathematics, however exquisite, but rather the communication needed to get stakeholders engaged as well as finding and declassifying the data. Nobel laureate Alvin Roth created a foundational matching algorithm, and he too experienced the same human and bureaucratic difficulties.

Roth suggests, "overall, one lesson from the [matching project] is that mechanism design in a political environment requires that not only policy makers themselves be persuaded of the virtues of a new design, but that they be able to explain and defend the mechanism to the various constituencies they serve." [1] The explainability of our algorithm is relatively simple in that it matches preferences from two sides – seekers and owners – yet any sort of deviation from existing processes creates concern and risk, real or imagined.

5.2 Data

We ran our analysis on 5 data sources

1. Med Corps
2. EOD
3. CMS-ID
4. Air Force
5. NSA

We collected the EOD and NSA data. We were provided the Med Corps, CMS-ID, and Air Force data. In these three preferences were not expressed completely; not every market participant expressed their ordinal preference for each opportunity. To create complete preferences we use the process developed by Shaw [13].

5.3 Results

6 Alternate Formulations

6.1 Co-Location

Alternate to deterministic algorithms (such as Deferred Acceptance [4]) is a linear programming (optimization) approach. This allows system owners to input strategic objectives in the seeker-owner job matching process. The importance of job seeker preference can be weighted to be more important than owner, or vice versa. Requirements can also be added, as you will see in the formulation below.

The optimization function takes the form:

$$\min \quad \sum_{i=1}^{i=n} \sum_{j=1}^{j=m} f(x_{i,j}) \tag{6}$$

$$\text{such that} \quad \sum_{j=1}^{m} x_{i,j} \leq 1 \quad \forall i \in \{1, \ldots n\} \quad \texttt{only one job per person} \tag{7}$$

$$\sum_{i=1}^{i=n} \sum_{j=1}^{j=m} x_{i,j} == \min(n, m) \quad \forall j \in \{1, \ldots m\} \tag{8}$$

$$\texttt{either all the jobs are filled or everyone has a job} \tag{9}$$

$$\sum_{i=1}^{n} x_{i,j} \leq a_j \quad \forall j \in \{1, \ldots m\} \quad \texttt{all jobs are at or below capacity} \tag{10}$$

$$\frac{1}{n_c} \sum_{i=1}^{n_e} C(e_1, e_2) \geq 0.95 \quad \texttt{at least 95\% of couples are co-located} \tag{11}$$

The Goodness Function f is the strategic objective function of the assignment process. For the sake of this paper, we set it to value the preference of the seeker twice as much as the preference of th job owner.

$$f(x_{i,j}) = 2P^S_{i,j} + P^O_{j,i}$$

In matrix form this can be re-written (with $tr()$ indicating the trace):

$$
\begin{aligned}
\min \quad & tr(2X^T P^O) + tr(XP^S) && (12)\\
\text{such that} \quad & \sum_{j=1}^{m} x_{i,j} \leq 1 \quad \forall i \in \{1, \ldots n\} \quad \texttt{only one job per person} && (13)\\
& X^T \bullet 1 \leq A \quad \texttt{all jobs are at or below capacity} && (14)\\
& \frac{1}{n_c} \sum_{i=1}^{n_e} X^T C X D \geq 0.95 \quad \texttt{at least 95\% of couples are co-located} && (15)
\end{aligned}
$$

Co-location Function C returns 1 if the couple is considered co-located, 0 if not or if single. Here we choose 50 miles between job locations to be consider co-located because that is the threshold for receiving dislocation allowance (DLA) for a permanent change of station (PCS) according to the Joint Travel Regulations (JTR). The location function $L(S_i)$ returns the lat/long location of the stationing for Seeker i.

$$
C(e_1, e_2) = \begin{cases} 0 & \text{if } e_2 == 0 \\ \mathbb{1}\big(||L(S_i) - L(S_j)|| \leq 50\big) & \text{otherwise} \end{cases}
$$

The inspiration and initial formulation of this optimization was done by a young Air Force officer who has since moved onto the private sector.

The extension of the formulation to include the Co-Location Function C and have more than one positions for each job a_i are the contributions of this team.

6.1.1 Explanation of Co-location constraint math

$$
\frac{1}{n_c} \sum \sum X^T C X \geq 0.95
$$

This works because

$$
\begin{aligned}
(CX)_{ij} &= \begin{cases} 1 & \text{Seeker } i\text{'s mate is assigned to job } j \\ 0 & \text{otherwise} \end{cases}\\
(X^T)_{ij} &= \begin{cases} 1 & \text{Seeker } i \text{ is assigned to job } j \\ 0 & \text{otherwise} \end{cases}\\
(X^T C X)_{ij} &= \begin{cases} n & \text{Number of couples assigned to the } (i,j) \text{ job pairing} \\ 0 & \text{otherwise} \end{cases}\\
D_{ij} &= \begin{cases} 1 & \text{The job pairing } (i,j) \text{ is considered co-location} \\ 0 & \text{otherwise} \end{cases}\\
(X^T C X)\dot{D} &= \text{ The Hadamard (element-wise) multiplication of } (X^T C X) \text{ and } D\\
\big((X^T C X)\dot{D}\big)_{ij} &= \begin{cases} n & \text{Number of co-located couples assigned to the } (i,j) \text{ job pairing} \\ 0 & \text{otherwise} \end{cases}\\
\sum \sum (X^T C X)_{ij} &= \text{Number of couples co-located}\\
\frac{1}{n_c} \sum \sum (X^T C X)_{ij} &= \text{Ratio of couples co-located}
\end{aligned}
$$

6.2 Tweaking Objective Function

Instead of a even weighting of job owner and seeker preferences in the objective function

$$f(x_{ij}) = x_{ij}(P_j^S + P_i^O)$$

Think of one where the Navy is seeking to enforce the "Needs of the Navy" upon the billeting process much more, weighting heavier the preferences of the job owners by a factor of ten

$$f(x_{ij}) = x_{ij}(P_j^S + 10P_i^O)$$

6.3 Specialization Objective

The objective function could be changed to maximize (minimize the negative) specialization aggregate of the Navy.

$$L_{ij} = \texttt{Specialization}_i^S \bullet \texttt{Specialization}_j^O \bullet (n - P_i^O) \bullet (m_a - P_j^S) \tag{16}$$

$$f(X) = -\sum\sum L \bullet X \tag{17}$$

The function f is the sum of the elements of the Hadamard (elementwise) product of the matrices L and X.

7 Future Work

This is where our future work will go.

1. Explore implications of a "Separation" (u) preference
2. Explore strategic Importance of positions. This can either be an iteration where billets in priority tranche's are run iterative (tier 1 billets all matched, those sailors and billets are taken out of the pool, then tier 2 billets all matched, etc.). Or a weighting where in a single matching optimization the preferences of higher tiered billets are given a greater weighting
3. Does adding weight for specialization Im objective function help the Navy better? Did job owners who wanted specialized sailors get them? Did sailors who wanted specialized jobs get them?
4. Tailored Compensation decisions.
5. Incorporate timeliness of expected rotation date for availability windows.
6. Multiple firms and multiple workers, each sailor can opt into collaterals and other roles [6]
7. K-Means Clustering Analysis (or some other unsupervised machine learning) of preferences

A possible move away from the approach of mixed-integer programming with ordinal preferences would be to explore the viability of Budish's wagering formulation of approximate competitive equilibrium from equal incomes. [3].

Appendices

A Notation

Throughout the paper, we will reference the notation listed in this section.

$$
\begin{align}
m &= \text{number of different jobs available} \tag{18}\\
n &= \text{number of persons} \tag{19}\\
\vec{P_i^S} &= \text{Preference vector of job seeker } i, \ \in \mathbb{Z}^{+,m\times 1} \tag{20}\\
P^S &= [\vec{P_i^S}|\dots|\vec{P_n^S}] \in \mathbb{Z}^{+,m\times n}, \tag{21}\\
&\quad \text{Preference Matrix of Seekers} \tag{22}\\
\vec{P_i^O} &= \text{Preference vector of job owner } j, \ \in \mathbb{Z}^{+,n\times 1} \tag{23}\\
P^O &= [\vec{P_i^O}|\dots|\vec{P_n^O}] \in \mathbb{Z}^{+,n\times m}, \tag{24}\\
&\quad \text{Preference Matrix of Job Owners} \tag{25}\\
\vec{A} &= \text{Position Available vector } \ \in \mathbb{Z}^{+,m\times 1} \tag{26}\\
a_j &= \text{Amount of positions for job } j, \ \in \mathbb{Z}^{+} \tag{27}\\
M &= \sum_i a_j \tag{28}\\
&= \ \text{number of openings across Navy} \tag{29}\\
M &\geq m \tag{30}\\
X &= \text{Placement Matrix} \in \{0,1\}^{n\times m} \tag{31}\\
x_{i,j} &= \begin{cases} 1 & \text{if } S_i \text{ is slated for job } j \\ 0 & \text{otherwise} \end{cases} \tag{32}\\
C &= \text{Co-Location Matrix, upper triangular} \tag{33}\\
C_{ij} &= \begin{cases} 1 & j > i, \text{ and Seeker } i \text{ requests co-location with Seeker } j \\ 0 & j \leq i, \text{ or Seeker } e_{i,1} \text{ does not request co-location} \end{cases} \tag{34}\\
n_c &= \text{number of couples requesting co-location} \tag{35}\\
&= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij} \tag{36}\\
& \tag{37}
\end{align}
$$

B Matching Algorithms

Alternate to deterministic algorithms (such as Deferred Acceptance [4]) is a linear programming (optimization) approach. This allows system owners to input strategic objectives in the seeker-owner job matching process. The importance of job seeker preference can be weighted to be more important than owner, or vice versa. Requirements can also be added, as you will see in the formulation below.

The optimization function takes the form:

$$\begin{aligned}
\min \quad & \sum_{i=1}^{i=n}\sum_{j=1}^{j=m} f(X_{i,j}) && (38)\\
\text{such that} \quad & \sum_{j=1}^{m} X_{i,j} \leq 1 \quad \forall i \in \{1,\dots n\} && (39)\\
& \sum_{i=1}^{i=n}\sum_{j=1}^{j=m} X_{i,j} = \min(n,m) \quad \forall j \in \{1,\dots m\} && (40)\\
& \sum_{i=1}^{n} X_{i,j} \leq a_j \quad \forall j \in \{1,\dots m\} && (41)\\
& && (42)
\end{aligned}$$

The Goodness Function f is the strategic objective function of the assignment process. For the sake of this paper, we set it to value the preference of the seeker twice as much as the preference of th job owner.

$$f(X_{i,j}) = X_{ij}(2P^S_{i,j} + P^O_{j,i})$$

In matrix form this can be re-written (with $tr()$ indicating the trace):

$$\begin{aligned}
\min \quad & 2tr(XP^O) + tr(X^TP^S) && (43)\\
\text{such that} \quad & \sum_{j=1}^{m} X_{i,j} \leq 1 \quad \forall i \in \{1,\dots n\} && (44)\\
& (X \bullet 1) \bullet 1 = \min(n,m) \quad \forall j \in \{1,\dots m\} && (45)\\
& X^T \bullet 1 \leq A && (46)\\
& && (47)
\end{aligned}$$

The constraints in lines (39) and (44) ensure that each seeker receives only one job, lines (40) and (45) ensure that either all the jobs are filled or everyone has a job, and lines (41) and (46) ensures that all jobs are at or below capacity (do not exceed capacity).

B.1 Matrix formulation of Objective Function

$$\begin{aligned}
\sum_{i=1}^{i=n}\sum_{j=1}^{j=m} f(X_{i,j}) &= \sum_{i=1}^{i=n}\sum_{j=1}^{j=m} X_{ij}(2P^S_{i,j} + P^O_{j,i})\\
&= \sum_{i=1}^{i=n}\sum_{j=1}^{j=m} 2X_{ij}P^S_{i,j} + \sum_{i=1}^{i=n}\sum_{j=1}^{j=m} X_{ij}P^O_{j,i}\\
&= \sum_{i=1}^{i=n}\sum_{j=1}^{j=m} 2X_{ij}P^S_{i,j} + \sum_{j=1}^{j=m}\sum_{i=1}^{i=n} X^T_{ji}P^O_{j,i}\\
&= \sum_{i=1}^{i=n} 2(XP^S)_{ii} + \sum_{j=1}^{j=m} (X^TP^O)_{jj}\\
&= 2tr(XP^O) + tr(X^TP^S)
\end{aligned}$$

C Competitiveness Extended Explanation

C.0.1 Average Ranking

The competitiveness score for any given job could be defined as the average preference ranking of the job across all seekers.

$$\texttt{Competitiveness}_j = \frac{1}{n}\sum_{i=1}^{m} P^S_{i,j}$$

C.0.2 Adapted Sciorintino Ratio

The issue with the *Average Ranking* approach is that, though conveying the desirability of a position, loses much of the information contained in the distribution of preferences. Consider two jobs out of a pool of ten jobs with five applicants.

$$O_1 = [1, 1, 1, 1, 5] \tag{48}$$
$$C_1 = 1.8 \tag{49}$$
$$O_2 = [2, 2, 2, 2, 1] \tag{50}$$
$$C_2 = 1.8 \tag{51}$$

We see that though job 1 is more competitive, job 2 has the same score, thus considered metrically, equally competitive.

An average of preference rankings, . For example, a position ranked first by ten individuals and hundredth by a twenty individuals has the same average ranking as a job ranked seventh by all thirty individuals; yet the prior is much more competitive.

The Sciorintino Ratio is a metric from the finance industry used to measure the performance of a investment vehicle based on the distribution of their returns. An investment vehicle with consistent, small positive returns is much different than one with consistent negative returns and one big win, even though the average return may be the same. We see the same problem in finance as the detailing marketplace, conveying the nature of a distribution in a single metric. The difference in this ratio, as compared to the more popular Sharpe Ratio, is that the Sciorintino ratio does not let positive volatility negatively effect the score, only volatility on the negative side is punishing. Here we want to adapt the the concept that the volatility of preference is reflected only with higher ranked preferences. For example, if a job has an average ranking of seven, the fact that many people ranked it first is much more important for competitiveness than the fact that many other people ranked it hundredth.

The formulation for our *Adapted Sciorintino Ratio* for the competitiveness of job j takes the form

$$\mu_j = \frac{1}{n}\sum_{i=1}^{m} P^S_{i,j} \tag{52}$$
$$\sigma_j = \frac{1}{m^2}\sum_{i=1}^{m} \mathbb{1}\left(P^S_{i,j} < \mu\right)\left(P^S_{i,j} - \mu\right)^2 \tag{53}$$
$$\texttt{Competitiveness}_j = \frac{\mu_j}{\sigma_j} \tag{54}$$

C.0.3 Weighted Scaling

The issue with the *Adapted Sciorintino Ratio* approach is that it has trouble adjusting for consistent ranking around the mean. Consider two jobs out of a pool of ten jobs with five applicants.

$$O_3 = [3, 3, 3, 2, 1] \tag{55}$$
$$C_3 = 56 \tag{56}$$
$$O_4 = [4, 3, 4, 2, 1] \tag{57}$$
$$C_4 = 24 \tag{58}$$

We see that though job 1 is more competitive, job 2 has a lower score, thus considered metrically more competitive.

So now we turn to a different ranking that scales based on the number of people, the number of jobs available, and the number of positions available in each job. This job scales from 1 being most competitive, and 0 being not competitive at all.

$$\texttt{Competitiveness}_j = 1 - \frac{1}{mn\sqrt{a_j}} \sum_{i=1}^{n} \sqrt{P^S_{i,j}}$$

Applying this to the previous example (assuming each has only one position available) we see the scores are in the proper order (job 2 is less competitive than job 1).

$$O_3 = [3, 3, 3, 2, 1] \tag{59}$$
$$C_3 = 0.912 \tag{60}$$
$$O_4 = [4, 3, 4, 2, 1] \tag{61}$$
$$C_4 = 0.906 \tag{62}$$

Yet we see that this is just the mean ranking squared scaled by the number of jobs available and the number of positions for the job. So, this scoring method still suffers from the issue of the average ranking, heavy tailed distribution of preferences may have the same competitiveness score as a normal distribution, even though the former is more competitive. To attempt to compensate for this, we adjust the average by a power of one half is to lessen the impact of preferences closer to m on the score, essentially weighting favorable preference more in our score consideration.

This can be seen by returning to our first example.

$$O_1 = [1, 1, 1, 1, 5] \tag{63}$$
$$C_1 = 0.892 \tag{64}$$
$$O_2 = [2, 2, 2, 2, 1] \tag{65}$$
$$C_2 = 0.885 \tag{66}$$

C.0.4 Weighted Scaling: Group Formulation

$$
\begin{aligned}
F_j &= 1 - \frac{a_j}{m_a} \\
&= \text{ \% of openings job } j \text{ is of all openings in the Navy} \\
S_i^S &= 1 - \frac{1}{nm} \sum_j \sqrt{P_{ij}^O} \\
&= \text{ sought after metric} \\
C_i^S &= S_i^S \\
&= \text{ competitiveness of a job seeker} \\
C_j^O &= F_j S_j^O \\
&= \text{ competitiveness of a job}
\end{aligned}
$$

TODO: put an example in here.

References

[1] P. Pathak A. Abdulkadiroglu, A. Roth and Sonmez T. Changing the boston school choice mechanism. Technical report, Boston Public Schools, January 2006.

[2] U. Rothblum A. Roth and J. Vande Vate. Stable matchings, optimal assignments, and linear programming. *Mathematics of Operations Research*, 18(4), November 1993.

[3] E.Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6), 2011.

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

[5] R. Irving. Stable marriage and indifference. *Discrete Applied Mathematics*, 48:261–272, 1994.

[6] A. Roth. The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7(4), November 1982.

[7] A. Roth. Stability and polarization of interests in job matching. *Econometrica*, 52(1):47–57, 1984.

[8] A. Roth. The college admissions problem is not equivalent to the marriage problem. *Journal of Economic Theory*, 36:277–288, 1985.

[9] A. Roth. Conflict and coincidence of interest in job matching: Some new results and open questions. *Mathematics of Operations Research*, 10(3), August 1985.

[10] A. Roth and M. Sotomayor. Interior points in the core of two-sided matching markets. *Journal of Economic Theory*, (45):85–101, 1988.

[11] A. Roth and M. Sotomayor. The college admissions problem revisited. *Econometrica*, 57(3):559–570, May 1989.

[12] S. Mitchell S. Khuller and V. Vazirani. On-line algorithms for weighted bipartite matching and stable marriages. *Theoretical Computer Science*, 127:255–267, May 1994.

[13] I. Shaw. Completing incomplete ordinal preferences. Paper is in the process of being published., March 2019.