Two-Sided Matching for the U.S. Navy Detailing Process with Market Complication*

Wei Yang Joseph Giampapa Katia Sycara †

November 2003

Abstract

The U.S. Navy detailing process is the matching process for assigning Sailors to available billets. This paper studies a new two-sided matching process for the detailing process to reduce the number of detailers, simplify the assignment process, and increase the satisfaction of Sailors and Commands. We focus on two-sided matching with market complications such as married couples looking for related positions. The existence of stable matchings is established by assuming all couples have responsive preferences, which means the unilateral improvement of one partner's job is considered beneficial for the couple as well. Based on its unique features and special requirements, we design a two-sided matching algorithm for the detailing process with the consideration of market complications including married couples, priority billets that must be filled, and high fill rate for Sailors. We believe that this algorithm deals with these market complications in an appropriate manner.

*CMU-RI-TR-03-49

†Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, weiy@cs.cmu.edu, garof@cs.cmu.edu, katia@cs.cmu.edu.

1 Introduction

With an active-duty personnel force of 371,800 (FY 2001 DoN Budgets [3]), the US Navy has considerable manpower assets to manage. A significant part of managing the Navy's manpower assets involves assignment processes, which seek to place the right person in the right job at the right time. An assignment process that achieves this optimal mix will maximize fleet readiness and ensure Sailors' careers and skills are developed and utilized to their potential. The assignment process involves three key groups of players. These are the Sailors to be assigned, the Commands seeking Sailors and the detailers who advise both Sailors and Commands. Detailers are also the primary decision-makers in the assignment process. To manage the careers and assignments of 314,450 enlisted personnel (FY 2001 DoN Budgets) requires considerable effort by approximately 294 detailers (Short (2000) [10]).

The Navy is not the only organization to face problems matching elements of its workforce with positions. Roth and Peranson (1999) [7] explores in detail the situation facing the entry-level labor market for American new physicians. This market is organized via a centralized clearinghouse called the National Resident Matching Program (NRMP). Each year around 20,000 jobs are filled through NRMP. Graduating physicians and other applicants first interview at residency programs throughout the country. They then compose and submit Rank Order Lists (ROLs) to the NRMP, each indicating an applicant's preferences ordering among the positions for which he/she has interviewed. Similarly, the residency programs submit ROLs of the applicants they have interviewed, along with the number of positions they wish to fill. The NRMP processes these ROLs and capacities to produce a matching of applicants to residency programs.

We believe that the Navy detailing process situation resembles the labor market for American new physicians. Both involve assigning the agents in one category to the agents in the other category. What makes these two assignment processes different from the single marriage market (see Gale and Shapley (1962)) [1] is the presence of market complications. One major complication is that married couples are looking for positions that are linked to each other. In early 1970s, a significant number of married couples declined to participate in the NRMP procedure, or to accept jobs assigned to them by that procedure, which indicates that the presence of couples introduces instability into the market. The Navy, and the U.S. Armed Services in general, take special consideration into account when two service members are married. This complicates the assignment process because the assigned locations to the couple are expected to be coordinated.

This paper focuses the stability issues of the Navy detailing process with market complications, especially with married couples. We investigate assumptions about the preferences of couples that assure the existence of a stable matching. A matching algorithm is presented to automate and simplify the Navy detailing process by assigning both single Sailors and couples to Navy positions. The paper is organized as follows. Section 2 introduces simple matching market without couples. Section 3 shows the complication that the presence of couples adds to the matching market, and analyzes the assumption on the couples' preference that lead to the existence of stable matchings. In Section 4 we introduce the Navy detailing process, discuss its unique features, and present a customized matching algorithm with considerations of market complications. We summarize related works in Section 5 and conclude in Section 6.

2 Matching Without Couples

A two-sided matching market is represented by two distinct categories of agents, with agents from each category seeking a match with agents from the other category. A two-sided matching model is a process that seeks to match these agents to each other so that the match is acceptable to each agent and the agents are collectively satisfied with the outcome. Two-sided matching models are described according to the market characteristics to which the model applies. The two basic matching models of interest are one-to-one and many-to-one matching models, which are discussed in the following sections.

2.1 One-to-One Matching

One important objective in the matching process is to find an assignment that would be stable. By stable we mean that no two Sailors prefer each other's position to their currently assigned ones. Let us concentrate on an artificial matching problem that captures the essential complexities of finding stable matchings. Suppose that you have the honored position of being the matchmaker in a small, traditional community. The task is to recommend suitable mates within the community for each of the men and women who wish to marry this year. Recommendations are based on the preferences of the individuals, each of whom has given you a ROL of the opposite sex, from the most preferred to the least preferred, with no ties allowed. Without some further goal or constraint, it is easy for everybody gets matched up with somebody. The questions are: What to optimize? and how to take the preferences into account?

The elements of the formal models are as follows. There are two finite and disjoint sets M and W: $M = m_1, m_2, ..., m_i$ is the set of men, and $W = w_1, w_2, ..., w_j$ is the set of woman. Each man has preferences regarding the women, and each woman has preferences regarding the men. These preferences may be such that a man m would prefer to remain single, which is denoted by u (unmarried), rather than be married to some woman w he does not care for. To express these preferences concisely, the preferences of each man m will be represented by an ordered list of preferences, $P(m)$, on the set $W \cup u$. That is, a man m's preferences might be of the form $P(m) = w_1, w_2, u, w_3, ..., w_p$ indicating that his first choice is to be married to woman w_1, his second choice is to be married to woman w_2, and his third choice is to remain single. Similarly, each woman w in W has an ordered list of preferences, $P(w)$, on the set $M \cup u$. We will usually describe an agent's preferences by writing only the ordered set of people that the agent prefers to being single. Thus the preferences $P(m)$ just described will be abbreviated by $P(m) = w_1, w_2$. A specific marriage market will be denoted by the triple $(M, W; P)$. We write $w \succ_m w'$ to mean m prefers w to w'. If an individual is not indifferent between any two acceptable alternatives, he or she has *strict preferences.* Let $P^W = \{P(w)\}_{w \in W}$ and $P^M = \{P(m)\}_{m \in M}$.

An outcome of the marriage market is a set of marriages. Formally we have

Definition 1 *A **matching** μ is a one-to-one correspondence from the set $M \cup W$ onto itself such that if $\mu(m) \neq u$ then $\mu(m) \in W$ and if $\mu(w) \neq u$ then $\mu(w) \in M$. We refer to $\mu(x)$ as the **mate** of x.*

A matching is unstable if a man and a woman who are married to other people would both rather be married to each other. Finding a stable matching is the stable marriage problem.

Definition 2 *A man-woman pair is a **blocking pair** for a matching if they are matched to others when both would rather be matched to each other. A matching is **stable** if there is no blocking pair for the matching (and is unstable if there is).*

The major mathematical questions that arise for achieving a stable matching are: (1) How to tell if a matching is stable? (2) Does a stable matching always exist? and (3) If there is a stable matching, how to find it?

We now examine a particular situation to see examples of blocking and non-blocking pairs, and stable and unstable matchings. We use m_i to stand for men's names and w_j for women's names where $i = 1, ..., 4$ and $j = 1, ..., 4$.. The preference rankings are

$$
\begin{aligned}
P(m_1) &= \{w_1, w_2, w_3, w_4\} \quad , \quad P(w_1) = \{m_1, m_3, m_2, m_4\}\,, \\
P(m_2) &= \{w_4, w_3, w_1, w_2\} \quad , \quad P(w_2) = \{m_4, m_2, m_3, m_1\}\,, \\
P(m_3) &= \{w_4, w_3, w_1, w_2\} \quad , \quad P(w_3) = \{m_1, m_4, m_2, m_3\}\,, \\
P(m_4) &= \{w_2, w_1, w_3, w_4\} \quad , \quad P(w_4) = \{m_3, m_4, m_1, m_2\}\,.
\end{aligned}
$$

For man m_1 his number one choice is w_1, followed by w_2, w_3 and w_4. Given the preference rankings, the matching $\mu(m) = w_3, w_1, w_4, w_2$, i.e.,

$$(m_1, w_3),\ (m_2, w_1),\ (m_3, w_4),\ (m_4, w_2)$$

is unstable, because (m_1, w_1) is a blocking pair. In fact, each member of the (m_1, w_1) pair is

Inputs:	preference lists for each man m and woman w
Output:	prints a blocking pair or prints “match is stable”
Precondition:	each list includes each potential partner once
Postcondition:	output statement is correct

Algorithm:

```
Stable = true no blocking pair found yet
for each man m
    for each woman w
        if ((m prefers w to his current partner) and
            (w prefers m to her current partner))
            write "m and w are a blocking pair"
            Stable = false
            exit loops no point continuing
        endif
    endfor
endfor
if Stable
    write "match is stable"
endif
```

Table 1: Algorithm to check if a matching is stable

the other member's first choice, but they are currently matched to other people.

To determine whether a matching is stable, we implement a checking method in an algorithm shown in Table 1. We illustrate the algorithm with the preferences given above for unstable matchings. We examine the men and women in numerical order. The algorithm ends quickly only because the elements of the blocking pair (m_1, w_1) are both the first in the preference orderings. If they had been last, we would have gone through all of the possible combinations of men with women before finding out that the matching is unstable.

For some mathematical problems, it is possible to prove that a solution exists without showing how to find one. We call it an existence proof. Better than an existence proof is a constructive proof, which uses an algorithm to produce a solution. Gale and Shapley (1962) [1] give a constructive proof that there is always a stable matching starting from any preference lists.

Theorem 1 *A stable matching exists for every marriage market.*

Proof Gale and Shapley (1962) [1] published the following algorithm for the stable marriage problem. The algorithm proceeds by rounds. Each round consists of two parts: Men make proposals of marriage, then women reject or (tentatively) accept. We will see that reversing these once-traditional roles can produce vastly different results.

In the first round each man proposes to the woman whom he most prefers, even if someone else has already proposed to her. Then, from the proposals that she receives, each woman tentatively accepts the proposal from (becomes engaged to) the proposer whom she prefers the most; she rejects all the other proposals. A woman who does not receive any proposals waits for the next round.

In each subsequent round men who are currently engaged do nothing. Each man who is not engaged makes a new proposal, to the woman highest in his preference ranking who has not already rejected him, whether or not she is already engaged. A woman accepts the proposal from the man highest in her ranking, rejecting all others and breaking her current engagement to become engaged to a man higher in her ranking if necessary. A woman who does not receive any proposals in this round waits for the next round.

As long as there are unengaged men at the end of a round, conduct another round. □

This algorithm is called a “deferred acceptance” procedure to emphasize the fact that women are able to keep the best available man at any step engaged, without accepting him outright. There are several important properties involved in this procedure. First, the deferred acceptance algorithm terminates finitely. No man proposes to the same woman twice, so each man can make at most n proposals. Altogether, all of the men together can make no more than n^2 proposals.

When the algorithm terminates, all the men are engaged. Since each man is engaged to exactly one woman, and there are exactly as many women as men, we have a matching. In addition, the matching is stable. If an arbitrary man prefers another woman to the one to whom he is matched, then he must have proposed to that other woman in some round. She must have rejected him because she preferred someone else. Hence, the man and woman in question cannot be a blocking pair. But since the man was arbitrary, and the woman was any woman whom he preferred over the one to whom he was matched, we have shown that there are no blocking pairs in the matching. Another interesting feature is that the matching produced is the same. No man's proposal itself is affected in any way by what another man does. Likewise, no woman's choice is affected by what other women do. So, in each round it doesn't matter in what order the proposals are made or in what order the women make their choices. Finally, no man could be better off in any other stable matching than he is in the deferred acceptance one. In other words, in the deferred acceptance matching, each man is as well off as he could be in any stable matching. Each woman ends up only as badly off as she might, given any stable matching process.

What the last property shows is that the deferred acceptance algorithm is not a symmetrical algorithm. Among stable matchings, it yields a matching that is the best possible for the group that does the proposing. It yields the worst possible matches for the group that receives the proposals, as compared to all other stable matchings. Of course, there are instances in which the same stable matching results when the roles of proposer and proposee are reversed; but in general, the two variations do not give the same matching.

2.2 Many-to-One Matching

There are clear similarities between the simple Navy detailing process without couples and the marriage market studied in the previous section. There are two kinds of agents, Navy Commands and Sailors looking for jobs, and the function of the market is to match them. Similarly, Navy Commands are able to rank order the Sailors who have applied for their positions

Inputs:	preference lists for each man m and woman w
Output:	a matching $\mu(w)$, containing the man woman w should marry
Preconditions:	each list includes each potential partner once; same number of men and women
Postcondition:	a matching is produced that is stable

Algorithm:

```
μ(w) = u
while there is an unmarried man
    pick an unpaired man, m_i
    w = first woman on m_i's list. Remove it.
    if w is engaged (μ(w)! = 0)
        if w prefers m_i to μ(w)
            new man is better than old one
            set man μ(w) to unpaired
            μ(w) = m_i
            set man m_i to paired
        else m_i remains unpaired
    else woman was not previously engaged
        μ(w) = m_i
        set man m_i to paired
```

Table 2: The deferred acceptance algorithm for a simple marriage matching.

and Sailors have preferences over the requisitions on the Navy job market. The major difference from the marriage market is that each Navy Command may recruit more than one Sailor, although each Sailor can take only one position.

The elements of the formal models are as follows. There are two finite and disjoint sets, $N = \{n_1, n_2, ..., n_n\}$ and $S = \{s_1, s_2, ..., s_m\}$, of Navy Commands and Sailors respectively. Similar to the marriage market, preferences are represented by ordered lists with $P(n) = s_1, s_2, \emptyset, s_3, ...$ denoting that Navy Command n prefers to hire Sailor s_1 rather than s_2 and it prefers to leave one position unfilled, denoted by $\emptyset$, rather than filling it with Sailors other than s_1 or s_2. $P(s) = n_2, n_1, n_3, u, ...$ represents the preferences of Sailor s precisely as in the marriage market. Navy Command n's preferences $P(n)$ over individual Sailors is represented by $s_i \succ_n s_j$ which means that n prefers Sailor s_i to s_j. $n_i \succ_s N_j$ indicates that Sailor s prefers Command n_i to n_j. For each n_i, there is a positive integer q_i called the quota of N_i, which indicates the maximum number of positions the Navy Command may fill. The definition of many-to-one matching is as follows.

Definition 3 *A matching μ is a function from the set $N \cup S$ into the set of unordered families of elements of $N \cup S$ such that:*

1. $|\mu(s)| = 1$ *for every Sailor s and $\mu(s) = s$ if $\mu(s) \notin N$;*
2. $|\mu(n)| = q_n$ *for every Navy Command n, and if the number of Sailors in $\mu(n)$, say r, is less than q_n, then $\mu(n)$ contains $q_n - r$ copies of n;*
3. $\mu(s) = n$ *if and only if s is in $\mu(n)$.*

The Sailors' preferences are exactly the same as in the simple marriage market. Next, we describe the preferences of Navy Commands regarding groups of Sailors. A Navy Command n's preferences over groups of Sailors are called **responsive** to its preferences $P(n)$ over individual Sailors if, for any two assignments that differ by only one Sailor, it prefers the assignment containing the more preferred Sailor (and is indifferent between them if it is indifferent between the Sailors). We will henceforth assume that Navy Commands have preferences over groups of Sailors that are responsive to their preferences over individual Sailors. With this assumption, stable matchings in the Navy detailing process can be identified using only the preferences P over *individuals* without knowing the preferences that each Navy Command has over *groups* of Sailors. This suggests that the Navy detailing model may be very similar to the marriage model. In fact, we can consider a related marriage market, in which each Command n with quota q_n is broken into q_n "pieces" of itself, so that in the related market, the agents will be Sailors and individual Navy positions, each having a quota of one. That is, we replace Command n by q_n positions of n by $n_1, n_2, ..., n_{q_n}$. Each of these positions has preferences over individuals that are identical with those of n. Since each position c_i has a quota of one, we do not need to consider its preferences over groups of Sailors.

If the preferences over individuals are strict, there is a natural one-to-one correspondence between matchings in the Navy detailing problem and matchings in the marriage market derived from it in this way. Furthermore, this one-to-one correspondence preserves the stability of the matching. A matching of the Navy detailing problem is stable if and only of the corresponding matchings of the related marriage market are stable.

One of the successful application of many-to-one matching algorithm is in the labor market for American medical interns (See Roth and Peranson (1999) [7]). The National Resident Matching Program (NRMP) is a private, non-for-profit corporation established in 1952, providing a uniform date of appointment to positions in graduate medical education in the United States. Referring to the NRMP algorithm for assigning medical interns to hospital programs in the US, we present a matching algorithm for the simple Navy detailing process.

Each Navy Command rank orders the Sailors who have applied to it, marking "X" any Sailors who are unacceptable. Each Sailor rank orders the Navy positions to which he has applied, similarly indicating any which are unacceptable. These lists are sent to the central clearinghouse, where they are edited by removing any unacceptable Navy positions and Sailors from their corresponding lists. The edited lists are thus rank orderings of acceptable alternatives.

These lists are entered into an algorithm consisting of a matching phase and an updating phase. The first step of the matching phase (the step $1:1$) checks to see if there are any Sailors and Navy Commands which are top-ranked in one another's ranking. (If n_i has a quota of q_i then q_i highest Sailors in its ranking are top-ranked.) If no such matches are found, the matching phase proceeds to the step $2:1$, at which the second ranked Navy position on each Sailor's ranking is compared with the top-ranked Sailors on that Navy Command's ranking. At any step when no matches are found, the algorithm proceeds to the next step, so the generic step $k:1$ of the matching phase seeks to find Sailor-position pairs such that the Sailor is top-ranked on the Navy Command's ranking and the Navy position is kth ranked by the Sailor. At any step where such matches are found, the algorithm proceeds to the tentative-assignment-and-update phase.

When the algorithm enters the updating phase from the step $k:1$ of the matching phase, the $k:1$ matches are tentatively made; i.e., each Sailor who is a top-ranked choice of his kth choice Navy Command is tentatively assigned to that position. The ranking of the Sailors and Navy Commands are then updated in the following way. Any Navy position which a Sailor s_j ranks lower than his tentative assignment is deleted from his ranking so the updated ranking of a Sailor s_j tentatively assigned to his kth choice now lists only his first k choices. At the same time, Sailor s_j is deleted from the ranking of any Navy Command which was deleted from s_j's ranking so the updated rankings of each Navy Command now include only those applicants who have not yet been tentatively assigned to a position more preferable than those excluded. Note that, if one of a Navy Command's top-ranked candidates is deleted from its ranking, then a lower-ranked choice moves into the top-ranked category, since the Command's updated ranking has fewer Sailors, but the same quota as its original ranking. When the rankings have been updated in this way, the algorithm returns to the start of the matching phase, which examines the updated rankings for new matches. Any new tentative matches found in the matching phase replace prior tentative matches involving the same Sailor. The algorithm terminates when no new tentative matches are found, at which point tentative matches become final. Any Navy position or Sailor which was not tentatively assigned during the algorithm is left unassigned, and must make subsequent arrangements by directly negotiating with other unmatched Sailor

Inputs:	strict preference lists for each Sailor s_i and Navy Command n_j
Output:	a matching μ, containing the Navy Sailor assignment
Preconditions:	each list includes each potential alternative once;
Postcondition:	a matching is produced that is stable
Algorithm:	
	Initial editing of rank-order lists
Matching phase:	
	Are there any (new) $1:1$ matches?
	If yes, then go to "Updating phase";
	Are there any (new) $2:1$ matches?
	If yes, then go to "Updating phase";
	......
	Are there any (new) $k:1$ matches?
	If yes, then go to "Updating phase";
	......
	Are there any (new) $n:1$ matches?
	(m=max number of Navy positions on any Sailor's list)
	If yes, then go to "Updating phase";
	STOP: All Sailors are now assigned to the Navy position on the bottom of their updated list.
Update phase:	
	Make all indicated tentative assignments.
	Delete all lower ranked Navy positions from each assigned Sailor's list.
	Delete tentatively assigned Sailors from the list of each Navy position that they ranked lower than their tentative assignment.

Table 3: The acceptance deference algorithm for the simple Navy detailing process

or positions.

Note that the matching procedure does not allow Navy Commands or Sailors to express indifference between alternatives, so the submitted rank orderings are strict preference lists. The procedure described above can be summarized into the following algorithm in Table 3.

This matching algorithm for the simple Navy detailing process is substantially different from the deferred acceptance algorithm for the marriage market. However, these two different algorithms are equivalent as stable matching mechanisms.

Theorem 2 *The matching algorithm for the simple Navy detailing process is a stable matching mechanism.*

Proof When the algorithm terminates, each Navy Command n_i is matched with the top q_i choices on its final updated rank-order list, which follows since the algorithm dies not terminate while tentative $k : 1$ matches can still be found. This assignment is stable, since any Sailor s_j who some Navy position n_i originally ranked higher than one of its final assignees was deleted from n_i's ranking when s_j was given a tentative assignment higher in his or her ranking than n_i. Hence the final assignment gives s_j a position he or she ranked higher than n_i. So the final matching is not unstable with respect to any such n_i and s_j. □

We use an example in Table 4 to illustrate the matching algorithm for the simple Navy detailing process. There are three Sailors and two Navy Commands, each with a quota of one. The preference lists are $P(n_1) = s_1, s_2, s_3, P(n_2) = s1_, s_2, s_3, P(s_1) = n_1, n_2, P(s_2) = n_1, P(s_3) = n_1, n_2$. At the initial step, it applies the initial editing to the rank order lists of the Sailors and Commands. Since n_2 is unacceptable to Sailor s_2, s_2 should be removed from the list of n_2. Hence, the edited preference of n_2 is $P(n_2) = s_1, s_3$. At the first step of the matching phase, there is a $1 : 1$ matching between n_1 and s_1, which means s_1 is the top-ranked in n_1's list and vice versa. Therefore, s_1 and n_1 are tentatively matched to each other. Next the algorithm enters the tentative-assignment-and-update phase. Since n_2 ranks lower than s_1's tentative match n_1, n_2 is deleted from s_1's rank list. At the same time, s_1 is also removed from n_2's ranking. The algorithm returns to the matching phase. There is a $2 : 1$ matching between n_2 and s_3. Since no new tentative matches can be found, the tentative match becomes final with an unique stable matching as $\mu(s) = n_1, s_2, n_2$, where Sailor s_1 and s_3 are assigned to position n_1 and n_2 respectively and s_2 remains unemployed.

P^N		P^S		
n_1	n_2	s_1	s_2	s_3
s_1	s_1	n_1	n_1	n_1
s_2	s_2	n_2		n_2
s_3	s_3			

Table 4: Simple example for the matching algorithm without couples

3 Matching with Couples

Over the last decades the proportion of women joining the US Navy has steadily been increasing. Therefore, it is not surprising that the number of couples searching jointly for a job in the same Navy labor market has been increasing as well. In addition to individual job quality, couple's preferences may capture certain "complementarities" that are induced by the distance between jobs, which means that the valuation of one partner's job may crucially depend on the other partner's job. The presence of complementarities may imply that some "desirable" economic outcomes, such as the Nash equilibrium and stable matching, fail to exist. For convenience and without loss of generality, we use a simple example to study the properties of the matching market with couples. There are 4 Navy Commands and 2 pairs of Sailors; $N = \{n_1, n_2, n_3, n_4\}, S = \{s_1, s_2, s_3, s_4\}$, and $C = \{c_1, c_2\} = \{(s_1, s_2), (s_3, s_4)\}$ are the sets of Navy Commands, Sailors and couples. Each Navy Command has exactly one position to be filled. All of our results can easily be adapted to more general situations that include other couples as well as single agents and Navy Commands with multiple positions. We first address the open question raised in Roth and Sotomayor (1990) [8], namely to find classes of "real world preferences" of married couples that ensure the existence of stable matching. We then show the possibility of stable matching with couples under certain preference assumptions.

3.1 Responsive Preferences

The preferences of Navy Commands and individual Sailors are defined the same way as in the matching market without couples. Each Navy Command $n \in N$ has a strict, transitive, and complete preference relation $\succeq_n$ over the set of Sailors and the prospect of having its position unfilled, which is denoted by $\emptyset$. We assume that typically each n prefers its position filled by some Sailor rather than leaving it unfilled. Let $P^N = \{P(n)\}_{n\in N}$. Similarly, each Sailor $s \in S$ has a strict, transitive, and complete preference relation $\succeq_s$ over the set of Navy Commands and the prospect of being unemployed, denoted by u. We assume that typically each s prefers being employed to being unemployed. Let $P^S = \{P(s)\}_{s\in S}$.

Each couple $c \in C$ has a strict, transitive, and complete preference relation $\succeq_c$ over all possible combination of ordered pairs of different Navy Commands and the prospect of being unemployed. Couple c's preferences can be represented by a strict ordering of the elements in $\mathcal{N} := [(N \cup \{u\}) \times (N \cup \{u\})] \setminus \{(n,n) : n \in N\}$. We denote a generic element of $\mathcal{N}$ by (n_p, n_q), where n_p and n_q indicate either a Navy Command or being unemployed. For instance, $P(c) = (n_1, n_3), (n_2, n_4), (n_2, u)$, etc., indicates that couple $c = (s_1, s_2)$ prefers being matched to n_1 and n_3 respectively, to being matched to n_2 and n_4 respectively, and so on. Let $P^C = \{P(c)\}_{c\in C}$. The one-to-one two-sided matching market with couples is denoted by (P^N, P^C).

Next, we introduce possible restrictions on the couples' preference.

If a couple prefers full employment to the employment of only one partner and the unemployment of both partners, we say that it is *strongly unemployment averse.* Formally, for a couple c, for all $n_p, n_q, n_r \neq u$, $(n_p, n_q) \succ_c (n_r, u) \succ_c (u, u)$ and $(n_p, n_q) \succ_c (u, n_r) \succ_c (u, u)$.

In our previous analysis we do not require any relation between Sailors' individual preferences and couples' preferences. In fact, we can not or do not always wish to specify individual preferences when couples are concerned. However, we do study some situations in which there is a clear relationship. This is the case when the unilateral improvement of one partner's job is considered beneficial for the couple as well.

Definition 4 *Couple $c = (s_k, s_l)$ has* ***responsive preferences*** *if there exist preferences $\succeq_{s_k}$ and $\succeq_{s_l}$ such that for all $n_p, n_q, n_r \in N \cup \{u\}, n_p \succ_{s_k} n_r$ implies $(n_p, n_q) \succ_c (n_r, n_q)$ and*

$n_p \succ_{s_l} n_r$ implies $(n_q, n_p) \succ_c (n_q, n_r)$. If these preferences $\succeq_{s_k}$ and $\succeq_{s_l}$ exist, then they are unique.

Responsive preferences may reflect situations where couples search for jobs in the same geographic area. If one partner switches to a job he/she prefers and the couple can still live together, then the couple is better off. Since responsiveness essentially excludes complementarities in couples' preferences that are caused by distance consideration, some results from the singles' market may be applied to the matching markets with couples.

We define an outcome μ for a couples market (P^N, P^C) to be a matching of Sailors and positions such that each Sailor is assigned to at most one position in N or to u, and each position in N is assigned to at most one Sailor. For a matching to be stable, it should always be better for Sailors (one or both members in a couple) to accept the position(s) offered by the matching instead of voluntarily choosing unemployment. For the Navy it should always be better to accept the Sailor assigned by the matching instead of leaving the position unfilled. A matching μ is *individually rational* if

(i1) for all $c = (s_k, s_l) \in C, (\mu(s_k), \mu(s_l)) \succeq_c (\mu(s_k), u), (\mu(s_k), \mu(s_l)) \succeq_c (u, \mu(s_k))$, and $(\mu(s_k), \mu(s_l)) \succeq_c (u, u)$;

(i2) for all $n \in N, \mu(n) \succeq_n \emptyset$.

If one partner in a couple can improve the given matching for the couple by switching to another position such that the Navy Command that holds the position is better off as well, then we would expect this mutually beneficial trade to be carried out, rendering the given matching unstable. A similar statement holds if both Sailors in the couple can improve. For a given matching μ, $((s_k, s_l), (n_p, n_q))$ such that $c = (s_k, s_l) \in C$ and (n_p, n_q) is a blocking coalition if

(b1) $(n_p, n_q) \succ_c (\mu(s_k), \mu(s_l))$;

(b2) $[n_p \in N$ implies $s_k \succeq_{n_p} \mu(n_p)]$ and $[n_q \in N$ implies $s_l \succeq_{n_q} \mu(n_q)]$.

A matching is *stable* if it is individually rational and if there are no blocking coalitions.

3.2 Existence of Stable Matchings

Roth (1984) [5] shows that stable matching may not exist without reasonable assumptions on couples' preferences. However, if the preferences of couples are responsive, there are no negative externalities from one partner's job for the other partner, or for the couple. Thus we can treat the market as if singles participate. By doing this, we can guarantee the existence of a stable matching. This would be the case if couples only apply for jobs in the same geographic area so that different regional preferences or travel distance are no longer part of the couples' preferences.

Let (P^N, P^C) be a couples market and assume that couples have responsive preferences. From the couples' responsive preferences we can uniquely determine the associated individual preferences for all agents. By $(P^N, P^S(P^C))$ we denote the associated singles market we obtain by replacing couples in (P^N, P^C) by individual Sailors with their associated individual preferences.

Theorem 3 *Let (P^N, P^C) be a couples market where couples have responsive preferences. Any matching that is stable for the associated single market $(P^N, P^S(P^C))$ is also stable for (P^N, P^C). In particular, there exists a stable matching for (P^N, P^C).*

Proof Let μ be a stable matching for $(P^N, P^S(P^C))$. Suppose that μ is not stable for (P^N, P^C). Hence, either there exists a blocking coalition or μ is not individually rational because $(i1)$ is violated.

Assume that $((s_k, s_l), (n_p, n_q))$ is a blocking coalition. By $(b1)$ we have $(n_p, n_q) \succ_c (\mu(s_k), \mu(s_l))$ and by $(b2)$ we have $[n_p \in N$ implies $s_k \succeq_{n_p} \mu(n_p)]$ and $[n_q \in N$ implies $s_l \succeq_{n_q} \mu(n_q)]$.

Either $n_p \succ_{s_k} \mu(s_k)$ or $n_q \succ_{s_l} \mu(s_l)$ together with the corresponding statement acceptability for the Navy Command in $(b2)$ would contradict the stability of μ in $(P^N, P^S(P^C))$. Hence, $\mu(s_k) \succeq_{s_k} n_p$ and $\mu(s_l) \succeq_{s_l} n_q$. But then responsiveness implies $(\mu(s_k), \mu(s)l)) \succeq_c (n_p, \mu(s_l)) \succeq_c (n_p, n_q)$, which would contradict $(b1)$.

Now assume $(i1)$ is violated. Then there exists a couple $c = (s_k, s_l)$ such that $(\mu(s_k), u) \succ_c (\mu(s_k), \mu(s_l)), (u, \mu(s_l)) \succ_c (\mu(s_k), \mu(s_l))$, or $(u, u) \succ_c (\mu(s_k), \mu(s_l))$. So, $((s_k, s_l), (\mu(s_k), u))$, $((s_k, s_l), (u, \mu(s_l)))$, or $((s_k, s_l), (u, u))$ is a blocking coalition. Using the same argument as before we obtain a contradiction.

Hence, μ is also stable for (P^N, P^C). Finally by Gale and Shaley (1962) [1] a stable matching for $(P^N, P^S(P^C))$ always exists. □

4 Matching Design with Market Complication

This section discusses the conceptual design of a matching algorithm for the U.S. Navy detailing process with market complications including married couples as well as other special features involved. To propose an algorithm to automate the process, it is essential to have an adequate understanding of the current processes.

4.1 Overview of Manpower and Personnel Processes

There are four sub-processes involved in the Manpower and Personnel Process shown in Figure 1. The first sub-process is known as the Manpower Requirements process, where force size and shape is determined based upon the national military strategy, which considers threats to the nation and the desired military capabilities. The second one is Manpower Programming that determines end strength and budget figures based on the manpower requirements. Personnel Planning is the third sub-process. It involves recruiting, training and strength planning based on the personnel who comprise the Navy. The final sub-process, the Personnel Distribution, is our focus in this paper, where the Navy detailing process takes place. It consists of three processes that are shown in Figure 1.

The Allocation process in Personnel Distribution involves apportioning Sailors available for

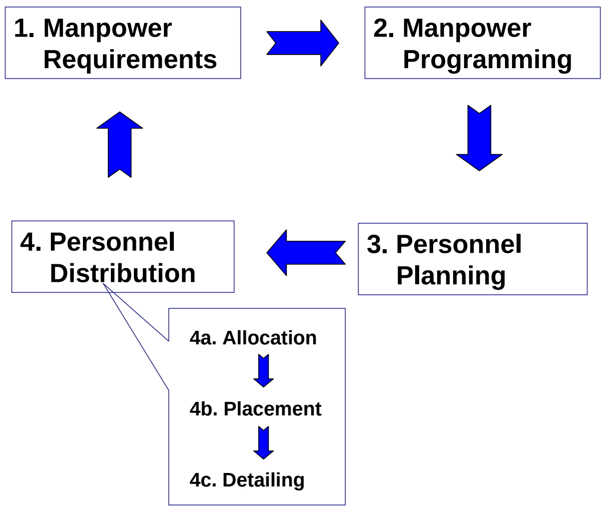

Figure 1: The U.S. manpower and personnel process

distribution to the four Manning Control Authorities (MCAs); Pacific Fleet, Atlantic Fleet, Reserve Forces and Bureau of Personnel. The Placement process then equitably spreads the projected strength across each of the four activities in each MCA: sea, submarine, air and shore. Finally, the Detailing process allocates specific Sailors to existing or projected vacancies, where for the first time Sailors' preferences are considered in the overall process.

4.2 The Navy Detailing Process

Short (2000) [10] presents the complete Navy detailing process. Instead of showing all the details, we focus on the issues related to applying two-sided matching to the detailing process.

When assigned to a billet within the Navy, a Sailor is given a Projected Rotation Date (PRD) which indicates the time window within which the Sailor's next assignment is expected to occur. To post a Sailor to a billet, detailers need to identify the projected future vacancies.

Vacancies are determined through requisitions, which are generated with the aid of various information systems that consider PRD and other factors. The detailing process occurs on a fortnightly cycle. During this cycle, upcoming vacancies are advertised online through the Job Advertising Selection System (JASS). After reviewing JASS, Sailors, or career counsellors as the Sailors' representatives, submit preferences to detailers during the fortnightly cycle. Once the cutoff time for applications is reached, JASS is temporarily closed to external viewing by Sailors, and detailers commence the assignment process.

Before choosing among various billet vacancies, a Sailor is required to evaluate a variety of aspects related to each position such as location, duty type (sea or shore), potential promotions. Preferences are then submitted directly to the detailer or through JASS. Detailers match Sailors with vacancies based on several criteria. The most important factor is a Sailor's rate, which refers to the pay grade, and rating, which refers to the occupation specialty. A detailer can not assign a Sailor with rate E-2 to fill the vacancy of an E-6, and it would be far from desirable for an E-6 to fill the position of E-2. The use of eligibility criteria (rate and rating) in the Navy detailing is a significant difference from the situation in NRMP, where medical graduates are generally homogeneous in terms of their skills and their position levels to which they apply (entry level). Other factors and policies that provide guidance to detailers in the matching process include gender, PRD, and sea/shore rotation.

4.3 Special Requirements in the Detailing Process

There are several considerations required in the Navy detailing process when a two-sided matching algorithm is applied.

First, similar to the NRMP process, as the number of women joining the Navy has been increasing over the last decade, more married couples are looking for positions that are related to each other, which complicates the detailing process. It is not hard to see why the presence of couples will cause matching problems. The deferred acceptance algorithm discussed above for the simple markets is a "one-pass" algorithm. Each Sailor moves down his/her preference list only once. The reason this produces a stable matching is that no Navy Command ever regrets any rejections it issues, since it only does so when it has a better Sailor in hand. So there is never a need for Sailor to reapply to a position that has already rejected him/her. Now consider the case in which a couple (s_k, s_l) applies for jobs together, and suppose that at some point in the algorithm their applications are being held by (n_p, n_q). Assume that in order to hold s_l's application, Navy Command n_q had to reject some other Sailor s_j. Suppose at the next step of the algorithm, n_p, holding the application of s_k, gets an offer it cannot refuse, and rejects Sailor s_k. Suppose further that couple (s_k, s_l)'s next preferred choice after (n_p, n_q) is (n_x, n_y). In order to move down the couples' preference list, Sailor s_l has to be withdrawn from n_q. This creates a potential instability involving Navy Command n_q and Sailor s_j such that n_q may now regret having rejected s_j. Therefore an algorithm like the deferred acceptance procedure will no longer be able to operate in a one pass through Sailors' preference. An alternative solution is expected to resolve the instabilities repeatedly and finally converges to a stable matching.

Secondly, in the detailing process all Sailor applicants are expected to be matched up positions and some priority billets can not be left unfilled. In 2000 the NRMP attempted to match 33,528 medical students with 3,769 programs offering 22,722 positions. Among the medical students, 25,056 remained active and were ranked by hospitals. Upon the completion of the match, 72.3% of the programs were filled and 74.7% of active applicants were matched to a position. The rest of the unmatched applicants apply for positions through personal channels. Hospitals with unfilled positions even search for international candidates to fill their positions. However, in the Navy assignment situation, it is unacceptable to leave 25% of Sailors without a match and expect them to spend time appealing to various Commands in search of employment. It is also not feasible for the Navy to recruit from an international force. Therefore, a two-sided matching algorithm applied to the detailing process is expected to ensure a 100% fill up rate for the Sailors. Further, while some Navy billets may be vacant at various times, certain critical billets must be filled. Such billets are usually requested directly by the Navy Commands and indicated by different priority levels showing the necessity of filling the billets.

Finally, one problem facing the Navy is the occasional need to assign Sailors to billets that they do not desire. This condition contradicts the assumption of two-sided matching that agents have the freedom to remain unmatched rather than being forced to an undesirable match. Therefore, incentives should be provided to encourage Sailors to accept positions that are generally regarded as undesirable. Matching with incentive is beyond the scope of the paper. In the next section on designing a matching algorithm for the Navy detailing process, we consider only the first two market complications.

4.4 Algorithm Design for Detailing Process with Market Complications

It seems unlikely that a "one-pass" algorithm would be feasible with the presence of couples. Therefore, any algorithm would need to check for and resolve potential instabilities at intermediate stages. Roth and Vande Vate (1990) [9] provide a basis for the conceptual design of the matching algorithm with couples. They show that instabilities could be resolved one at a time so that the process would always converge to a stable matching. The idea is that, starting from an unstable matching, a new matching can be created by "satisfying" one of the blocking pairs, i.e., creating a new matching in which the Command and applicant in some blocking pair are matched with one another, perhaps leaving an applicant previously matched to n_p unmatched, and a position previously occupied by s_l, unfilled. When the matching starts from scratch, the algorithm will proceed the same way as in the applicant-proposing deferred acceptance algorithm, by satisfying blocking pairs involving unmatched applicants. As potential instabilities develop due to the presence of couples, these would be resolved one at a time.

As for the requirement on the fill rate for Sailors, it is unlikely to achieve a 100% fill rate using an automated matching process without any manual manipulation. However, higher percentage of match up will be ensured when the preference lists can be extended. From the Command's perspective, billets must extend their preference lists to include all Sailors who are eligible, with additional lattitude for certain eligibility criteria at lower preference levels. On the other hand, Sailors are expected to extend their preference lists to include all billets they are qualified for, as well as positions that they are either over-qualified for (e.g., an E-5 for an E-4 billet), or positions for which they almost qualify for (e.g., an E-6 may almost qualify for an E-7 billet).

To ensure that priority billets are filled, one option is to divide the billets into two groups: those designated as high priority and those not. The high priority billets are matched against the entire set of Sailors in the first phase. The rest of the unmatched Sailors then enter in the second phase to be matched up with the regular billets.

Based on the above discussion on dealing with market complications, we summarize a matching algorithm for the Navy detailing process in Table 5. The matching process is divided into two phases. The first phase, called the priority phase, deals with the priority billets that must be filled. We use the standard deferred acceptance algorithm to match priority billets with eligible Sailors. In case the Sailor matched with a priority billet belongs to a couple looking for co-locations, the other member will be assigned to a billet as well, based on the couple's preference. All Sailors and couples assigned billets are removed from the applicants' pool.

In the second phase, denoted the regular phase, we first select an applicant, either an individual or a couple, and let that Sailor start at the top of the Sailor's preference list, and work down until the most preferred Navy Command or Commands willing to hold the application is reached. Sailors who are displaced in this process continue working their way down their preference lists, and when this causes a member of a couple to be withdrawn from a Navy Command, that Navy billet becomes vacant to be checked later for potential match. The applicants who form blocking pairs with a Navy Command will be put back on the "Sailor stack" to be considered again, starting with their most preferred billets.

To understand the algorithm, we use a single example to illustrate the matching process with couples in the regular billet phase. Consider the couples market (P^N, P^C) where preferences are given by Table 6 and the Sailors' individual preferences P^S as $P(s_1) = P(s_2) = n_1, n_2, n_3, n_4, u,$ $P(s_3) = n_2, n_1, n_3, n_4, u,$ and $P(s_4) = n_3, n_4, n_2, n_1, u$. Applying the algorithm in Table 5, we obtain one stable matching $\mu(S) = n_2, n_4, n_1, n_3$. The detail in resolving the

Inputs:	strict preference lists for single Sailor, couples and Commands
Output:	a matching μ, containing the Navy Sailor assignment
Preconditions:	each list includes each potential alternative once
Postcondition:	a matching assigning both singles and couples,
	filling the priority billets, and achieving high fill rate
Algorithm:	
Priority phase:	Apply deferred acceptance algorithm matching priority billets.
	Remove matched Sailors (including couples) from applicant pool.
Regular phase:	
0.0	Initialization: Stack contains all the rest Sailors (couples at bottom);
	Initial matching: $\mu = \emptyset$ (all regular billets unfilled, all Sailors unmatched).
	$t = 1$.
$1.t$	Any Sailors in stack? If no, go to $8.t$; else,
$2.t$	Select the individual Sailor or couple ($s = s_i$ or $c = (s_i, s_j)$ at the top of
	the Sailor stack (and remove from stack): set $m = 1$. $q = 1$.
$3.t - 1.q$	Sailor's preference list has at least n entries preferred to μ?
	If no, go to 9.0; else,
$3.t - 2.q$	Sailor applies to mth choice on preference list (if it is a couple,
	this may involve an application to two distinct Navy Commands).
$3.t - 3.q$	Does (each) Navy Command ($n = n_i$ or $n = (n_i, n_j)$) applied to either
	have a vacancy, or have no vacancy but prefer applicant to the least
	preferred other applicant currently held?
	If no, $m = m + 1, q = q + 1$, go to $3.t - 1.q$;
	else, (Navy Command now "holds" new applicant),
$4.t$	Does (either) Navy Command need to reject previously held applicant
	to make room for holding new applicant?
	If no, $\mu = \mu \cup (n, s)$ and
	$[(n, s) = \{n_i, s_i\} \text{or} \{(n_i, s_i), (n_j, s_j)\}]$, go to 9.0; else,
$5.t$	Put rejected applicant(s) s' at the top of the stack.
$6.t$	Is a rejected Sailor s_i, a member of a couple (s_i, s_k) AND
	is s_k's application currently held by some Navy Command n_k?
	If no, $\mu = \mu \cup (n, s) \setminus (n, s')$, go to 9.0; else,
	(Loop detector here: same couple displaced by same applicant?)
$7.t$	Withdraw a_k's application from n_k (making n_k's position vacant).
	$\mu = \mu \cup (n, s) \setminus \{(n_i, s_i), (n_k, s_k)\}$. Go to 9.0.
$8.t$	Check the stability of the matching at which each n_i is matched to the
	applicant it is holding. Stable?
	If yes, stop. Current matching is final matching; else, ...
9.0	$t = t + 1$, go to $1.t$.

Table 5: The matching algorithm for the Navy detailing process with market complications

P^N				P^C	
n_1	n_2	n_3	n_4	$\{s_1, s_2\}$	$\{s_3, s_4\}$
s_3	s_2	s_4	s_2	n_1, n_2	n_2, n_3
s_2	s_3	s_3	s_3	n_1, n_3	n_2, n_4
s_1	s_1	s_1	s_1	n_1, n_4	n_2, n_1
s_4	s_4	s_2	s_4	n_2, n_1	n_1, n_3
				n_2, n_3	n_1, n_4
				n_2, n_4	n_1, n_2
				n_3, n_1	n_3, n_4
				n_3, n_2	n_3, n_2
				n_3, n_4	n_3, n_1
				n_4, n_1	n_4, n_3
				n_4, n_2	n_4, n_2
				n_4, n_3	n_4, n_1

Table 6: Simple example for the matching algorithm with couples

matching problem is illustrated in the Appendix.

5 Literature Review

The game-theoretical analysis of marriage market is begun by Gale and Shapley (1962) [1]. They show that the set of stable matching in this simple market is never empty. Roth and Sotomayor (1990) [8] study a variety of issues involved in the two-sided matching market including one-to-one matching and many-to-one matching. There are many papers on the application of two-sided matching to the entry-level labor market for American physicians. Roth and Peranson (1999) [7] introduce the history of NRMP and the development of matching algorithms for this market. They describe how a new algorithm is designed to deal with couples looking for related positions in an appropriate manner. Using computational simulations and analyzing previous data, they show that the new algorithm is to be expected to perform well in practice. Roth (2002) [6] gives a more recent review of the redesign of the NRMP algorithm in the context of analyzing the engineering aspects of economic design. He also provides a good overview on how to address the presence of couples.

The Navy enlisted distribution problem has drawn a lot of attention in recent years. Short

(2000) [10] provides a complete and detailed introduction on the Navy enlisted detailing process. Robards (2001) [4] discusses the issues in applying two-sided matching to the detailing process without presenting any algorithm. Gates and Nissen (2002) [2] focus on the design of electronic Navy employment markets using information agents technology. They use an exploratory experiment to assess the performance of different employment market designs and analyze the social welfare implications.

6 Conclusion

The U.S. Navy detailing process is the matching process for assigning Sailors to available billets. This paper studied the issues involved in applying two-sided matching to the detailing process. We focused on two-sided matching with market complications, such as married couples looking for related positions. The existence of stable matchings is established by assuming all couples have responsive preferences, which means the unilateral improvement of one partner's job is considered beneficial for the couple as well. Based on its unique features and requirements, we designed a matching algorithm for the detailing process, with consideration given to market complications. We believe that this algorithm deals well with couples, priority billets, and a high fill rate requirement, in an appropriate manner.

A Solving Matching Example in Table 6

0.0 $\mu^0(N) = \emptyset, \emptyset, \emptyset, \emptyset$.

1.1 Yes.

2.1 (s_3, s_4) is selected and set $n = 1$.

3.1 − 1.1 Yes, (s_3, s_4) has more than one entry preferred to $\mu^0(s_3, s_4) = (u, u)$.

3.1 − 2.1 (s_3, s_4) applies to (n_2, n_3).

3.1 − 3.1 Yes, n_2 "holds" s_3 and n_3 "holds" s_4.

4.1 No, no rejection is needed; $\mu^I(N) = \emptyset, s_3, s_4, \emptyset$.

1.2 Yes.

2.2 $(s_1, s_2$ is selected and set $n = 1$.

3.2 − 1.1 Yes, (s_1, s_2) has more than one entry preferred to $\mu^I(s_1, s_2) = (u, u)$.

3.2 − 2.1 (s_1, s_2) applies to (n_1, n_2).

3.2 − 3.1 Yes, n_1 "holds" s_1 and n_2 "holds"s_2.

4.2 Yes, n_2 rejects s_3; $\mu^{II}(N) = s_1, s_2, \emptyset, \emptyset$.

5.2 (s_3, s_4) is at the top of the stack.

6.2 s_3 is rejected and s_4 is currently being held by n_3.

7.2 s_4 is withdrawn from n_3; $\mu^{II}(N) = s_1, s_2, \emptyset, \emptyset$.

1.3 Yes.

2.3 (s_3, s_4) is selected and set $n = 1$.

3.3 − 1.1 Yes, (s_3, s_4) has more than one entry preferred to $\mu^{II}(s_3, s_4) = (u, u)$.

3.3 − 2.1 (s_3, s_4) applies to (n_3, n_4).

3.3 − 3.1 No, n_2 prefers s_2 to s_3. Set $n = 2$.

(to be continued)

3.3 − 1.2 Yes, (s_3, s_4) has more than two entries preferred to $\mu^{II}(s_3, s_4) = (u, u)$.

3.3 − 2.2 (s_3, s_4) applies to (n_2, n_4).

3.3 − 3.2 No, n_2 prefers s_2 to s_3. Set $n = 3$.

3.3 − 1.3 Yes, (s_3, s_4) has more than three entries preferred to $\mu^{II}(s_3, s_4) = (u, u)$.

3.3 − 2.3 (s_3, s_4) applies to (n_2, n_1).

3.3 − 3.3 No, n_2 prefers s_2 to s_3 and n_1 prefers s_1 to s_4. Set $n = 4$.

3.3 − 1.4 Yes, (s_3, s_4) has more than four entries preferred to $\mu^{II}(s_3, s_4) = (u, u)$.

3.3 − 2.4 (s_3, s_4) applies to (n_1, n_3).

3.3 − 3.4 Yes, n_3 has a vacancy and n_1 prefers s_3 to s_1.

4.3 Yes, n_1 rejects s_1; $\mu^{4.3}(N) = s_3, s_2, s_4, \emptyset$.

5.3 (s_1, s_2) is at the top of the stack.

6.3 s_1 is rejected and s_2 is currently being held by n_2.

7.3 s_2 is withdrawn from n_2; $\mu^{III}(N) = s_3, \emptyset, s_4, \emptyset$.

1.4 Yes.

2.4 (s_1, s_2) is selected and set $n = 1$.

3.4 − 1.1 Yes, (s_1, s_2) has more than one entry preferred to $\mu^{III}(s_1, s_2) = (u, u)$.

3.4 − 2.1 (s_1, s_2) applies to (n_1, n_2).

3.4 − 3.1 No, n_1 prefers s_3 to s_1. Set $n = 2$.

3.4 − 1.2 Yes, (s_1, s_2) has more than two entries preferred to $\mu^{III}(s_1, s_2) = (u, u)$.

3.4 − 2.2 (s_1, s_2) applies to (n_1, n_3).

3.4 − 3.2 No, n_1 prefers s_3 to s_1. Set $n = 3$.

3.4 − 1.3 Yes, (s_1, s_2) has more than three entries preferred to $\mu^{III}(s_1, s_2) = (u, u)$.

3.4 − 2.3 (s_1, s_2) applies to (n_1, n_4).

3.4 − 3.3 No, n_1 prefers s_3 to s_1. Set $n = 4$.

(to be continued)

$3.4-1.4$	Yes, (s_1, s_2) has more than four entries preferred to $\mu^{III}(s_1, s_2) = (u, u)$.
$3.4-2.4$	(s_1, s_2) applies to (n_2, n_1).
$3.4-3.4$	No, n_1 prefers s_3 to s_2. Set $n = 5$.
$3.4-1.5$	Yes, (s_1, s_2) has more than five entries preferred to $\mu^{III}(s_1, s_2) = (u, u)$.
$3.4-2.5$	(s_1, s_2) applies to (n_2, n_3).
$3.4-3.5$	No, n_3 prefers s_4 to s_2. Set $n = 6$.
$3.4-1.6$	Yes, (s_1, s_2) has more than six entries preferred to $\mu^{III}(s_1, s_2) = (u, u)$.
$3.4-2.6$	(s_1, s_2) applies to (n_2, n_4).
$3.4-3.6$	Yes, n_2 and n_4 are vacant.
4.4	No.
1.5	No.
8.5	The matching $\mu(N) = s_3, s_1, s_4, s_2$ is stable.

References

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

[2] B. Gates and M. Nissen. Two-sided matching agents for electronic employment market design: Social welfare implications. Technical report, Naval Postgraduate School, Monterey, California, 2003.

[3] Department of the Navy. Fiscal year (fy) 2001 budget estimates. Technical report.

[4] P. Robards. Applying two-sided matching processes to the united states navy enlisted assignment process. Master's thesis, Naval Postgraduate School, Monterey, California, 2001.

[5] A. Roth. The evolution of the labor market for medical interns and residents: A case study in game theory. *Journal of Political Economy*, 92(6):991–1016, 1984.

[6] A. Roth. The economist as engineer: Game theory, experimentations, and computation as tools for design economics. *Econometrica*, 70(4):1341–1378, July 2002.

[7] A. Roth and E. Peranson. The redesign of the matching market for american physicians: Some engineering aspects of economic design. *The American Economic Review*, 89(4):748–780, September 1999.

[8] A. Roth and M. Sotomayor. *Two-Sided Matching: A Study in Game-Theoretic Modelling and Analysis*. Econometric Society Monograph Series. Cambridge University Press, 1990.

[9] A. Roth and J. Vande Vate. Random paths to stability in two-sided matching. *Econometrica*, 58(6):1475–80, 1990.

[10] M. Short. Analysis of the current navy enlisted detailing process. Master's thesis, Naval Postgraduate School, Monterey, California, 2000.