3-2023

# The U.S. Army Officer-to-Unit Assignment Problem

Andrea L. Phillips

**THE U.S. ARMY OFFICER-TO-UNIT ASSIGNMENT PROBLEM**

THESIS

Andrea L. Phillips, Second Lieutenant, USAF

AFIT-ENS-MS-23-M-151

**DEPARTMENT OF THE AIR FORCE**
**AIR UNIVERSITY**

***AIR FORCE INSTITUTE OF TECHNOLOGY***

**Wright-Patterson Air Force Base, Ohio**

AFIT-ENS-MS-23-M-151

# THE U.S. ARMY OFFICER-TO-UNIT ASSIGNMENT PROBLEM

THESIS

Presented to the Faculty

Department of Operational Sciences

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Operations Research

Andrea L. Phillips, B.S.A. Mathematics

Second Lieutenant, USAF

March 23, 2023

AFIT-ENS-MS-23-M-151

# THE U.S. ARMY OFFICER-TO-UNIT ASSIGNMENT PROBLEM

THESIS

Andrea L. Phillips, B.S.A. Mathematics
Second Lieutenant, USAF

Committee Membership:

Phillip LaCasse, Ph.D
Chair

Jesse Wales, Ph.D
Member

AFIT-ENS-MS-23-M-151

# Abstract

Every two to three years, U.S. Army officers must change duty stations, which entails a selection process based on preferences. Currently, officers are assigned to units using a stable-marriage algorithm. Two impracticalities occur within this process. First, officers are required to submit strictly ranked preferences, not allowing indifference among units. Second, the stable-marriage algorithm does not give flexibility to alternative priorities.

This research focuses on two modifications to the current model. First, a mixed integer program is created that allows the user, U.S. Army Human Resources Command, to consider other priorities: unit preferences and maximum officer disappointment. Second, generated data allowing officer indifference is tested and compared to the stable-marriage baseline. These solutions are tested using 25 models, each created by perturbing various parameters. Most models produce equivalent solutions, with differences stemming from placing no priority on maximum officer disappointment. These results are quantified and tested using metrics measuring officer satisfaction, providing insightful knowledge to decision makers weighing policy changes in this process.

# Acknowledgements

I would like to thank my research advisor, Dr. Phillip LaCasse, for his patience, guidance, and mentorship throughout my research. It was a pleasure to work together, and thanks to him I have created something that I am truly proud of.

To Dr. Jesse Wales, thank you for the time and feedback you put into my thesis. Your willingness to be my reader from across the world is appreciated and shows the dedication you have for students.

Andrea L. Phillips

# Table of Contents

Page

# List of Figures

# List of Tables

THE U.S. ARMY OFFICER-TO-UNIT ASSIGNMENT PROBLEM

# I. Introduction

This chapter introduces the assignment process for United States (U.S.) Army officers, how the U.S. Army Human Resources Command organizes the officer-to-unit matchings, and the challenges with the current system. Research intent will then be provided and a summary of the paper concludes Chapter I.

## 1.1 Background/Introduction

As of July 2022, there are 94,248 active duty officers in the U.S. Army [1]. Of those officers, one-third of them are moved by the U.S. Army Human Resources Command (HRC) annually. An officer typically stays in an assigned location for two to three years, then will receive a Permanent Change of Station (PCS) to a different location. In 2019, to help manage the assignments of officers to future locations, HRC adopted what is called the Army Talent Marketplace Alignment Process (ATAP). Through ATAP, the Assignment Interactive Module (AIM) is an interface that allows officers and units to submit and ordinally rank their preferences [14].

At the beginning of each distribution cycle (DC), officers and units submit an ordinal list of preferences. Officers submit their preferences for units, while units submit their preferences for officers. Preferences are submitted from most desirable to least desirable. HRC then takes the lists of preferences and matches officers to units based on both lists, making this process two-sided.

## 1.2 Problem Statement

Currently, there are some impracticalities that occur within this process. The preference list officers (units) create is a strict preference list, meaning every unit (officer) must be strictly ranked. The quality of a preference list depends on how well the officer (unit) fills out his or her preference list. This potentially creates a problem. Suppose an officer has 100 billets to create an ordinal preference list from, and there are *highly desirable*, *desirable*, *indifferent*, *undesirable*, and *highly undesirable* assignments according to the officer's preferences. He or she ranks the 15 highly desired and desired billets and leaves the rest of the 85 billets unranked. Because the algorithm requires the input of a complete list, the remaining 85 non-ranked assignments are randomly assigned a preference. Within the pool of his or her 85 indifferent, desirable, and highly undesirable preferences, there is no way for the algorithm to know how to place his or her indifference preferences ahead of the undesirable preferences. An indirect potential consequence is the relationship between the officer receiving or failing to receive a desired assignment and that officer's subsequent morale and retention.

In 2020, most officers, about 80%, received an assignment that was within their top ten choices, and 55% received an assignment that was their first choice [14]. However, if an officer did not receive a top three preference, he or she was equally likely to receive the bottom choice as the number four choice. The effect is a very large number of reasonably content officers but a small number who are highly dissatisfied. This poses a question that HRC must consider: Is it preferable to maximize satisfied officers or minimize dissatisfied officers? The current approach favors the former at the expense of a small number of the latter.

In this paper, the main research goal is to analyze the results of these different scenarios and make suggestions based on the findings. Changing the current model to incorporate this type of situation improves the overall happiness of officers and assignment managers, which indirectly may affect other measures such as morale and attrition rate.

A final challenge for the current AIM model is the need for analysts to manually adjust the pure market solution, which is the initial solution the model produces. Currently, about 5% of assignment pairings are adjusted from the pre-market solution [5]. Some officers have constraints preventing them from being able to be assigned in certain locations. Examples include an officer who is enrolled in the Exceptional Family Member Program (EFMP) or an officer who needs a command assignment.

Minimizing the number of adjustments made on the pure-market solution will decrease the amount of time assignment officers have to spend manually adjusting issues, saving time and resources.

### 1.3 Research Objectives

Specific research objectives are formed:

1. Replicate results of the current stable marriage-based approach with a mixed integer program.
2. Produce and quantify the effects of alternative objective functions on officer-unit matchings in the MIP.
3. Propose and quantity the effects of alternative preferencing inputs.

MIP models are better suited for the addition of constraints and adjusting the objective function relative to the current SMA approach. Therefore, creating a baseline MIP model in the first objective creates a baseline for the latter two objectives. The second and third objectives provide HRC leadership with information to support future policy decisions. The current approach has been in place for a relatively short period of time, and this research represents a continuous improvement process.

## 1.4 Methodology/Approach

This document is organized as follows. Chapter II provides an overview of relevant background information. Chapter III details the process of creating the mixed integer programming model and the stable marriage algorithm, testing these models, and applying sensitivity analysis to each. Chapter IV presents and analyzes the results. Finally, Chapter V discusses the conclusions drawn from the results and proposes directions for future research.

# II. Background and Literature Review

## Overview

This chapter defines the assignment problem and describes methodologies employed to solve it, specifically the stable marriage algorithm (SMA) and mixed integer programming (MIP).

## 2.1 The Assignment Problem

The assignment problem derives from the transportation problem [2]. Figure 1 graphically illustrates the transportation problem. We have two sets: origin points $O_i$ such that $i = 1, 2, ...m$ represents the origin locations, and destination points $D_j$ such that $j = 1, 2, ...n$ represents the destination locations. Each origin node $O_i$ has a given supply, $s_i$, and each destination node $D_j$ has a given demand, $d_j$. Traditionally, the goal of the transportation problem is to minimize transportation costs while fulfilling demand given supply. Equation (1) shows the linear program for the transportation problem.

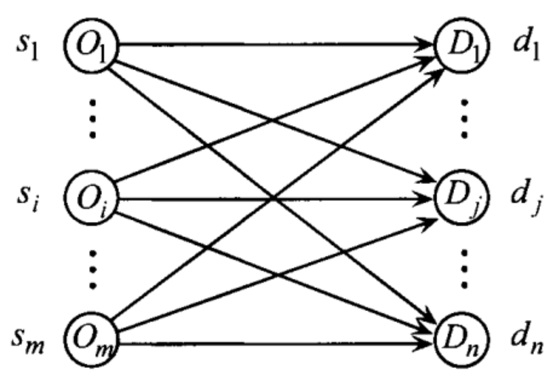


Figure 1: Complete bipartite graph of the transportation problem. [2]

$$
\begin{aligned}
\text{Minimize} \quad & \mathbf{cx} \\
\text{subject to} \quad & \mathbf{Ax} = \mathbf{b} \\
& \mathbf{x} \geq \mathbf{0}
\end{aligned} \tag{1}
$$

In Equation (1), **c**, the cost of transportation, is being minimized. In this instance of the transportation problem, **x** denotes the number of units shipped from origin node $i$ to destination node $j$. Matrix **A** is a node-arc incidence matrix, which indicates which nodes are able to use which arcs. Finally, vector **b** is the supply and demand. In this case, supply and demand are equivalent, hence the equality sign in the first constraint.

The assignment problem is a special case of the transportation problem. In this context of the assignment problem, there are an equal number of $m$ individuals and $n$ jobs, and the goal is to minimize cost while matching sets in a one-to-one fashion. From the transportation problem, let $s_i = 1$ for all $i$ and $d_j = 1$ for all $j$. There is an associated cost, $c_{ij}$, with assigning individual $i$ to job $j$. Only one individual can be assigned to one job, hence the one-to-one matching. This also means $x_{ij}$ is a binary variable where $x_{ij} = 1$ indicates there is a matching from $i$ to $j$, and $x_{ij} = 0$ indicates otherwise.

Many applications to the assignment problem occur throughout the literature. Some examples include matching medical students to hospitals [15], college applicants and colleges [7], and weapons to targets [13]. The two former applications aim to maximize preference in choice between the agents, and the latter aims to maximize destruction probabilities when matching particular weapons to targets. In this paper specifically, the assignment problem is used to maximize the preference matchings between officers and units.

## 2.2 Mixed Integer Programming Approach to the Assignment Problem

The first instances of mixed integer programming (MIP) arguably began with the work of Dantzig, Fulkerson, and Johnson when they created a cutting plane technique to solve the Traveling Salesman Problem [8]. Their work was a precursor to Land and Doig's branch and bound technique which iteratively takes the floor and ceiling of each variable needing to become an integer until an optimal solution is found [10]. Later, Dakin was able to improve Land and Doig's method by adding bounding restraints which shrinks the feasible region at each iteration [4], rather than looking for a specific integer in Land and Doig's work.

Bazaraa, Jarvis, and Sherali recognize the relationship between the transportation problem and the assignment problem, and develop the following mixed integer program for the assignment problem [2]:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{m}\sum_{i=1}^{m} c_{ij}x_{ij} \\ \text{subject to} \quad & \sum_{i=1}^{m} x_{ij} = 1, \quad i = 1, ..., m \\ & \sum_{i=1}^{m} x_{ij} = 1, \quad j = 1, ..., m \\ & x_{ij} = 0 \text{ or } 1, \quad i, j = 1, ..., m. \end{aligned} \tag{2}$$

Assuming the matrix $\mathbf{A}$ is unimodular, we can replace $x_{ij} = 0$ with $x_{ij} \geq 0$ to obtain the following matrix-form MIP:

$$\begin{aligned} \text{Minimize} \quad & \mathbf{cx} \\ \text{subject to} \quad & \mathbf{Ax} = \mathbf{b} \text{ where } \mathbf{b} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{3}$$

In the objective function, $\mathbf{c}$ is the cost associated with matching $i$ and $j$. $x_{ij}$ is a binary variable that indicates if the matching of $i$ and $j$, where $x_{ij} = 1$ indicates the matching of $i$ and $j$ exists, and $x_{ij} = 0$ indicates otherwise. In order to assume modularity in $\mathbf{A}$, $i$ and $j$ must have the same cardinality. If necessary, dummy variables may be employed to satisfy this condition.

Cimen uses a similar MIP to assign Turkish Air Force personnel to assignments [3]. When modeling, three objective functions are considered, from most to least importance: maximize organizational objective, maximize career objective, and maximize personal preference objective. Note that maximizing is the direction of the objective function rather than minimizing it. With these objective functions in mind, and order of importance, Cimen uses a maximin principle. The objective function maximizes the organization objective, and the constraints maximize the minimum value for the latter two constraints. Cimen then runs different maximin constraint inputs to test how it affects the objective function.

### 2.3 The Stable Marriage Algorithm

The SMA was first introduced by Gale and Shapley in 1962 [7]. In their work, the most notable contribution was to show that there is always an optimal solution for a two-sided bipartite problem. They first introduce the SMA in terms of a college admissions selection, with one agent being the applicants and the other being the colleges.

It is important to note the assumptions associated with the Gale-Shapley (GS) algorithm [7]. The first is that there is an equal number of applicants to colleges. The second is that there is a strict ordinal preference among both sides. This means that there are no instances of ties in the preference ranking.

In order to conceptualize the problem in a more suitable context, Gale and Shapley altered the college admissions setting into marriage, hence the name "stable marriage algorithm" [7]. Instead of matching student applicants to colleges, the goal of the marriage problem is to marry, or match, men and women according to their preferences.

The following definitions provide a foundation for all follow-on work related to the SMA [7]:

- Unstable: There exists a man and woman who prefer each other to their current partners. In other words, if there exists two couples in a solution, say $(m_i, w_i)$ and $(m_j, w_j)$ such that $m_i$ and $w_j$ prefer each other to their current partners, an instance of unstable matching occurs. This matching is also referred to as a *blocking pair*.

- Stable: A solution such that no unstable matchings exist.

- Optimality: Every player is at least as well off in their matching solution as they would be in any other solution.

The SMA iterates through the list of men's preferences. If a man is not engaged, the algorithm iterates through his preferences proposing to the most preferred woman on his list. If the woman is not engaged, the man and woman become engaged, or in other words, a matching forms. If the man proposes to a woman who is already engaged, she has a choice of either breaking off her engagement to be with the man or staying with her original match. The algorithm iterates through all the men's preferences until there are no unengaged men. This is illustrated in Algorithm 1, which is adopted from Ferguson[6].

---

**Algorithm 1** Pseudocode for the Gale-Shapely Stable Marriage Algorithm

---

```
M is the set of Men, W the set of Women
for m ∈ M, w ∈ W:
pref_m, pref_w are preference lists for each men, women
S = ∅; the pairs (m, w) in the current matching

procedure STABLE MARRIAGE(pref_m, pref_w)
    while ∃ m ∈ M s.t.(m, w) ∉ S for some w ∈ W do
        m ← select m s.t. m ∉ p, ∀p ∈ S
        w ← pop(pref_m)
        p ← p ∈ S where w ∈ p
        if w ∉ p for some pair p ∈ S then
            S ← (m, w)
        else if w prefers m to current match then
            remove pair p from S
            S ← (m, w)
        else
            do nothing
        end if
    end while
end procedure
```

---

Gale and Shapley determined that a unique optimal solution is possible but not guaranteed; if no unique optimal solution is possible, there can be more than one *best* set of matchings [7]. Depending on whether the man or woman proposes, there will be a stable solution with optimality towards men or women, respectively. In the case above, the solution will be optimal for the men. It should be noted that there are two different kinds of optimality: optimality toward one agent or a unique optimal solution. Regardless of which optimality the algorithm finds, a stable solution or a stable set of matchings, is always able to be found.

The original Gale-Shapley algorithm is a significant basis that has been added onto since it was written. Manlove used the GS algorithm to solve the Hospital Residents (HR) problem[11]. In the HR problem, there are two sets of agents: medical student residents and hospitals. Each resident and hospital provides a strict order of preferences, and if a resident does not include $h_i \in H$ where $h_i$ is a hospital in H, the set of hospitals, then $h_i$ is said to be unacceptable to that resident. The same concept applies to hospitals with residents. If a hospital (resident) is unacceptable, that resident (hospital) will not be on that hospital's (resident's) preference list. This gives no chance that a resident (hospital) will be matched with an unacceptable hospital (resident). The hospitals, $H$, have a quota of residents they must fill. As a resident proposes to the first hospital on their list, if the hospital's quota is oversubscribed, the hospital may reject its worst assigned resident and match with the proposing resident. The biggest finding from Manlove is for any stable matching found between residents and hospitals, all other stable solutions have the same assigned and unassigned residents, and the same hospitals meet or do not meet their quota by the same amount of residents for all solutions.

Roth's National Intern Matching Program (NIMP) algorithm provides a step by step way to solve a problem similar to the HR problem [15]. In order to describe the algorithm, constraints must first be established. There are two sets of agents: medical interns and hospitals. Each intern submits a strictly ranked preference list of hospitals, and each hospital does the same for the interns. Each hospital indicates a quota it aims to fill. This indicates how many interns a hospital desires. In this problem, it is important to note that lists need not be complete, meaning a preference list may have *unacceptable* matchings. If an intern marks a hospital as unacceptable, the intern is indicating he or she would rather be unmatched than be matched with that hospital.

In the first step of NIMP, any 1:1 matchings are created [15]. Specifically, if an intern has a top-ranked hospital, and a hospital has that same intern in their top rankings, a match is formed. If a hospital has quota $q$, the $q$ highest students in the hospital preferences are considered top ranked. At each iteration of this step, preference rankings are updated. Once an intern and hospital are matched, all successive hospitals on the intern's preferences are deleted. Once this is done, 1:1 matchings are looked for again, and preferences are updated until there are no more 1:1 matchings.

The algorithm then looks at 2:1 matchings, meaning any intern who has a hospital second ranked, and a hospital who has a student top-ranked, are matched [15]. The same deletion of successive preferences as described previously is implemented. These steps are iterated through until no 2:1 matchings are found. The algorithm continues this pattern of matching $k$:1 interns to hospitals until $k = n$ where $k = 1, 2, ..., n$ and $n$ is the last ranked hospital on an intern's preference list.

At the completion of NIMP, there may be interns who remain unmatched and hospitals who have not filled their quotas [15]. In the problem of this paper, no officers or jobs should remain unmatched. However, the idea of cycling through top-ranked preferences in the NIMP algorithm will prove useful in the SMA.

### Stable Marriage with Ties

It is impractical to assume a strict ranking of preferences occurs in every situation. In an instance of stable marriage with ties (SMT), a solution is *weakly stable* if there is no $x$ and $y$ such that $x$ and $y$ are not matched, but both prefer each other to their current match [9]. A solution is *strongly stable* if there is no matching $(x, y) \in S$ such that $x$ strictly prefers $y$, and $y$ is indifferent between $x$ and another partner. Finally, a solution is *super stable* if there is no such matching $(x, y) \in S$ such that either partner strictly prefers another partner to their own or is indifferent between them. Section 2.3 illustrates the definitions defined.

Irving describes an algorithm for SMT that considers two agents who must all be

Table 1: Stability Definitions

| Term | Definition | Symbolic Representation |
|---|---|---|
| Weakly Stable | No matching $(x_i, y_i) \notin S$ such that both partners prefer each other to their current partners. | $\forall (x_i, y_i) \notin S$, $x_i : y_i \succ y_{-i}$, $y_i : x_i \succ x_{-i}$ |
| Strongly Stable | No matching $(x_i, y_i) \in S$ such that $x_i$ strictly prefers $y_{-i}$, and $y_i$ strictly prefers $x_i$ or is indifferent. | $\forall (x_i, y_i) \in S$, $x_i : y_i \not\prec y_{-i}$, $y_i : x_i \succeq x_{-i}$ |
| Super Stable | For all $(x_i, y_i) \in S$, neither $x_i$ nor $y_i$ strictly prefer another partner or is indifferent between them. | $\forall (x_i, y_i) \in S$, $x_i : y_i \not\preceq y_{-i}$, $y_i : x_i \not\preceq x_{-i}$ |

matched by the end of the algorithm [9]. However, Irving specifically looks at finding strongly stable and super stable matchings, where this paper is only concerned with weakly stable matchings. Nonetheless, his insights are still useful. Irving names the following algorithm *super*. Men and women are the set of agents. Each agent is allowed to submit ties in his or her preferences. When cycling through the men's preferences, similar to Roth's cycling in the NIMP algorithm, if a proposing man has two or more women tied at the head of his preferences, he proposes to all of them simultaneously. All men inferior to the proposing man on a woman's preference list are deleted. Again, this is similar to the NIMP algorithm where successive hospitals on an intern's list are deleted once a match is made. A series of proposals are iterated through until either a man's preference list is empty or all men are engaged. If there is a man whose list is empty and is unmatched, it is concluded that there is no super stable matching for these sets of agents. Irving goes into detail about a *strong* algorithm that determines if a strongly stable matching is possible; it builds off the super algorithm. For more information, see [9].

As previously mentioned, Gale and Shapley determined there is always a set of stable matchings [7]. However, their conditions required a strict ordinal ranking and same cardinality for each agent. This paper specifically looks at the incorporation of ties in matching preferences. This type of problem does not guarantee a one-sided optimal solution like it does for an instance of SMA with strict preferences, but a stable solution is still able to be found by breaking the ties in some manner. This ensures there is a weakly stable solution in any instance of SMT [12].

### 2.4 Concluding Remarks

The definitions and models in this chapter will be applied to the methodology in this paper. Namely, the mixed integer program and the stable marriage algorithm. In this paper, this aim is to optimize the matchings between Army officers and bidding units. Different objective functions and their weights will be considered in order to find a solution that best fits HRC's needs.

# III. Methodology

## Overview

This chapter describes two methods of solving the U.S. Army Officer assignment problem: mixed integer programming and the stable marriage algorithm.

## 3.1 Assumptions and Data

In each of the models created, data is generated randomly and ties are permitted in different ways depending on the model. Ties in unit preferences for the SMA will not be considered to minimize the complexity of the problem. This is not ideal because units are generally indifferent to their choice of officer; however, the main contribution of this paper is the MIP model, which can accommodate ties on both sides.

One question that must be considered is how ties will be submitted into officer preferences. This question entirely depends on HRC's needs and what they suit best, but the way ties are chosen in this paper seeks to minimize officers being able to manipulate preferences for their own advantage.

The generated random preferences with ties are adjusted so that if an officer has a tie between two units, the next ordinal rank in the preference is skipped over. Assume the initial set of preferences an officer has is $[3, 4, 3, 1, 5]$, where each index represents a unit number, and each input in that respective index is an officer's preference for that unit. This means that the officer has a preference of 3 for unit 1, preference of 4 for unit 2, etc. This list then gets converted to $[2, 4, 2, 1, 5]$. Preferences for unit 1 and 3 are changed because there was no preference of 2, and unit 4 (the officer's highest preference) is not tied with any other unit. The officer's preference for unit 2 needs no changing because the ties in units 1 and 3 cause a skip in the ordinal list from 2 to 4.

For the remainder of this paper, $i$ is an officer in the set of officers, $I = 1, 2, ..., n$, and $j$ is a unit in the set of units, $J = 1, 2, ..., n$. The cardinality of $I$ and $J$ are assumed to be equal, meaning there are an equal number of officers and units being considered.

## 3.2 Mixed Integer Program

In the context of this particular problem, specific constraints and objective functions are created in order to satisfy the needs of HRC. This section explains each constraint and the objective with all its terms.

### 3.2.1 Sets

Each officer and unit has a preference list, where 1 is the highest preference; $o_{ij}$ denotes the preference of officer $i$ for unit $j$, and $u_{ij}$ gives the preference of unit $j$ for officer $i$.

### 3.2.2 Decision Variables

The decision in this problem is how to assign officers to units. The first decision variable, $x_{ij}$, is a relaxed binary indicator variable that represents the matching from $i$ to $j$. If $x_{ij} = 1$, then officer $i$ is sent to unit $j$; if $x_{ij} = 0$, then officer $i$ does not go to unit $j$.

A second decision variable, $q$, represents the greatest ranking that a matched officer gives to his or her destination unit. In other words, $q$ quantifies the disappointment of the least satisfied officer in the movement slate. In Section 3.2.5, it is shown how exactly $q$ aids the decision variable $x_{ij}$.

### 3.2.3 Assignment Constraints

The following constraints ensure that each officer is assigned to a unit and each unit has an officer assigned to it:

$$\sum_{j \in J} x_{ij} \geq 1, \quad \forall i \in I$$

$$\sum_{i \in I} x_{ij} \leq 1, \quad \forall j \in J.$$

In the first constraint, the decision variables associated with each officer are summed, ensuring each officer is matched with at least one unit. In the second constraint, the decision variables associated with each unit are summed, ensuring each unit is matched with at most one officer. Note that the inequalities in the assignment constraints above differ from the assignment constraints in Equation (2). This adjustment lessens computational complexity without changing the optimal solution. Because each officer must be assigned to at least one unit and each unit must be assigned to at most one officer, and the cardinality of $I$ and $J$ are equal, a one-to-one match among sets remains.

### 3.2.4 Stability Constraint

Like any assignment problem considering preferences, one of the goals of this research is to provide stability among matches. Specifically, given a solution, there are no $i$ and $j$ such that they are not matched but prefer each other to their current partners. This concept was introduced in Section 2.3 with the notion of weak stability. The following set of constraints achieves this desired outcome.

$$\sum_{s \prec_i j} x_{is} + \sum_{t \prec_j i} x_{tj} + x_{ij} \leq 1, \quad \forall i \in I, j \in J.$$

In the first summation, $s$ represents all units desired more than the matched unit in an officer's preference list. In the second summation, $t$ represents all officers desired more than the matched officer in a unit's preference list. When combining the summation of the set of officers $s$ and the set of jobs $t$, and $x_{ij}$, there will be no instance where it is greater than 1. Given an officer-unit matching, every unit an officer preferences more must be assigned to a more preferred officer, and every officer a unit preferences more must be assigned to a more preferred officer.

### 3.2.5 Minimax Constraint

One of the past issues HRC has seen with the current model is that there is a small number of officers who receive highly undesirable unit matchings while the majority receive one of their desired preferences. The effect is a large number of satisfied officers but a small number whose disappointment is highly disproportionate. In order to relieve this issue, and to ensure a minority of officers do not receive a very undesired unit, a minimax constraint is added to the model.

$$\sum_{j \in J} o_{ij} x_{ij} \leq q, \quad \forall i \in I$$

The minimax constraint minimizes the maximum value preference an officer is matched with. The variable being minimized, $q$, represents the least preferred unit any officer receives. It is important to note lower preference numbers are more desired than the higher ones, meaning a minimum preference number is better than a maximum. Therefore, the direction of preference is to minimize. Minimizing the maximum ranked unit an officer receives attempts to ensure each officer has a chance at not receiving a highly undesired unit.

This model only considers a minimax constraint for officer preferences and not the unit preferences. Officers receiving their more preferred match is more valued to

HRC than the units with their preferences. Unit matchings are clearly considered, as seen in Section 3.2.6; however, a minimax constraint for unit preferences is not essential for the purposes of this model.

### 3.2.6 Objective Function

So far in the model, officer preferences, unit preferences, and minimax constraints have been emphsized with importance. Therefore, in the objective function, it is appropriate to minimize a combination of these values.

$$\sum_{j \in J} \sum_{i \in I} w_o o_{ij} x_{ij} + w_u u_{ij} x_{ij} + w_q q$$

Each $w$ represents a weight associated with the following term. $w_o$ represents the weight associated with the officer preferences; $w_u$ represents the weight associated with the unit preferences; and $w_q$ represents in weight associated with the minimiax constraint. These weights allow for easy manipulation of the value put on each term. For instance, if wanting to place more value in the officer preferences, an increase in $w_o$ is an easy adjustment. The example in Section 3.2.7 demonstrates this concept.

Equation (4) displays the full MIP.

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{j\in J}\sum_{i\in I} w_o o_{ij} x_{ij} + w_u u_{ij} x_{ij} + w_q q \\
\text{subject to} \quad & \sum_{j\in J} x_{ij} \geq 1, \quad \forall i \in I \\
& \sum_{i\in I} x_{ij} \leq 1, \quad \forall j \in J \\
& \sum_{s \prec_i j} x_{is} + \sum_{t \prec_j i} x_{tj} + x_{ij} \leq 1, \quad \forall i \in I, j \in J \\
& \sum_{j\in J} o_{ij} x_{ij} \leq q, \quad \forall i \in I \\
& x_{ij} \geq 0, \quad i, j = 1, ..., n.
\end{aligned} \tag{4}
$$

### 3.2.7 MIP Example

The following examples demonstrate how the different algorithms in this chapter operate. Each demonstration will use the data from Table 2.

Table 2: Toy Example

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | (1, 2) | (3, 3) | (3, 1) |
| $i_2$ | (2, 3) | (1, 1) | (3, 3) |
| $i_3$ | (3, 1) | (3, 2) | (1, 2) |

This table is set up in matrix form where each row represents an officer's preferences toward each unit and each column represents a unit's preferences toward each officer. In this case three officers are candidates for assignment to three units. Inside the matrix, each cell has a value of (officer preference, unit preference) format. For example, cell $(i_1, j_1)$ has a value of $(1, 2)$ meaning $i_1$ has a preference of 1 for $j_1$, and $j_1$ has a preference of 2 for $i_1$.

For ease of understanding, each $w$ may have a minimum value of 0 and maximum value of 1. Furthermore, the relationship between the officer and unit weights give and take from one another; the more weight placed on the officers' preferences, the less can go to the units, and vice versa. Therefore, the weights associated with officers and units, $w_o$ and $w_u$, respectively, must sum to 1. The weight $w_q$ is not a part of a special relationship that $w_o$ and $w_u$ share; it can hold values from 0 to 1. Table 3 and Table 4 demonstrate a simple example where different solutions occur when applying different weights to the objective function.

When inputting these preferences into the MIP with different weights, different solutions arise. For simplicity, let this instance of MIP only consider the officer and unit preferences in the objective function, and not the minimax variable. Table 3 shows the solution when $w_o = 1$, $w_u = 0$, and $w_q = 0$, and Table 4 shows the solution when $w_o = 0$, $w_u = 1$, and $w_q = 0$.

Table 3: Officer-Biased Solution

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | (1, 2) | (2, 3) | (2, 1) |
| $i_2$ | (2, 3) | (1, 1) | (3, 3) |
| $i_3$ | (2, 1) | (2, 2) | (1, 2) |

Table 4: Unit-Biased Solution

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | (1, 2) | (2, 3) | (2, 1) |
| $i_2$ | (2, 3) | (1, 1) | (3, 3) |
| $i_3$ | (2, 1) | (2, 2) | (1, 2) |

Table 3 has an optimal solution of $(i_1, j_1)$, $(i_2, j_2)$, $(i_3, j_3)$ where the sum of officer and unit matching preferences is 8. Table 4 has an optimal solution of $(i_3, j_1)$, $(i_2, j_2)$, $(i_1, j_3)$ where the sum of matching preferences is also 8. In this example, both the officer-based solution and unit-based solution provide equivalent sums of matching preferences, but they are not the same solution. There is an officer-optimal solution and a unit-optimal solution. One of the goals of this research is to analyze how solutions may change depending on different objective function parameters. In larger instances such as those in Chapter IV, solutions will vary greatly depending on those parameters.

### 3.3 Stable Marriage with Ties

Similar to the MIP, officer preferences and unit preferences are inputs to the algorithm. The data from both sets of preferences are randomized as two separate data frames, with ties allowed in officer preferences.

The SMT requires a set of officer and unit preferences such that officer preferences may have ties. Suppose we have a set of two officers, $i_1$ and $i_2$, being matched to a set of two units, $j_1$ and $j_2$. Let $(i_1, j_1)$ be a matching $p$ such that $p$ is in $S$. At the beginning of the next iteration, the algorithm begins by taking the first free officer $i_2$ and assigning $j_1$ to be its top preferred unit. If $j_1$ is not part of a matching $p$ such that $p$ is in $S$, $(i_1, j_2)$ is added to $S$. However, this is not the case because $j_1$ is in $S$. If $j_1$ is part of a matching $p$ in $S$, which is the case, the algorithm checks to see if $j_1$ strictly prefers or is indifferent to $i_2$ and its current matching, $i_1$. If this is the case, the previous matching $(i_1, j_1)$ is removed from $S$, and the new matching $(i_1, j_2)$ is added to $S$. An additional step is made if $j_1$ strictly prefers $i_2$ to its current matching: $j_1$ is removed from the previous officer's preferences. This way, $j_1$ can never be matched with $i_1$ again. Every time a matching $p$ is added to $S$, all successors of the matched officer on the unit's preference list are deleted. The algorithm then proceeds to the next free officer and iterates through these steps until none remain.

The goal of this SMT is to find a stable solution when ties are included in officer preferences. The most significant modification from Section 2.3 is the consideration of indifference. If a unit receives a proposal from an officer who is preferred higher or indifferent than their their current officer, the previous match is deleted and the proposing officer and unit are placed in the matching set. If the unit strictly prefers the proposing officer, then that unit is removed from the previous officer's preferences. This avoids blocking pairs and aids the goal of finding a stable solution.

Because the algorithm iterates through the set of free officers, assigning $j$ to be

**Algorithm 2** Pseudocode for Stable Marriage with Ties

I is the set of Officers, J the set of Units
for $i \in I, j \in J$:
$\text{pref}_i, \text{pref}_j$ are preference lists for each officer, unit
$S = \emptyset$; the pairs $(i, j)$ in the current matching
$F = i \in I$ where $i \notin S$ are the officers not currently in $S$

```
procedure STABLE MARRIAGE WITH TIES(pref_i, pref_j)
    while ∃ i ∈ I s.t.(i, j) ∉ S for some j ∈ J do
        i ← pop(F)
        j ← pop(pref_m)
        p ← p ∈ S where j ∈ p
        if j ∉ p for some pair p ∈ S then
            S ← (i, j)
        else if j prefers i or is indifferent to current match then
            if j strictly prefers i then
                remove j from pref_{i∈p}
            end if
            F ← i ∈ p
            remove pair p from S
            S ← (i, j)
        else
            do nothing
        end if
        remove successors of i from pref_j
    end while
end procedure
```

the relevant top preferred unit, the algorithm is officer-biased. Solutions outputted by this SMT will be optimal toward the officers' preferences, not the units'.

In the case of Algorithm 2, the following steps find the solution from Table 2:

Step 1:

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | $(1,2)$ | $(2,3)$ | $(2,1)$ |
| $i_2$ | $(2,3)$ | $(1,1)$ | $(3,3)$ |
| $i_3$ | $(2,1)$ | $(2,2)$ | $(1,2)$ |

Current: $S=\emptyset$, $F=\{i_1,i_2,i_3\}$
Let $i=i_1$, then $j=j_1$
$j \notin p$ such that $p \in S$
$\rightarrow S$ gets $(i_1,j_1)$

Step 2:

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | $(1,2)$ | $(2,3)$ | $(2,1)$ |
| $i_2$ | $(2,3)$ | $(1,1)$ | $(3,3)$ |
| $i_3$ | $(2,1)$ | $(2,2)$ | $(1,2)$ |

Current: $S=\{(i_1,j_1)\}$, $F=\{i_2,i_3\}$
Let $i=i_2$, then $j=j_2$
$j \notin p$ such that $p \in S$
$\rightarrow S$ gets $(i_2,j_2)$

Step 3:

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | $(1,2)$ | $(2,3)$ | $(2,1)$ |
| $i_2$ | $(2,3)$ | $(1,1)$ | $(3,3)$ |
| $i_3$ | $(2,1)$ | $(2,2)$ | $(1,2)$ |

Current: $S=\{(i_1,j_1),(i_2,j_2)\}$, $F=\{i_3\}$
Let $i=i_3$, then $j=j_3$
$j \notin p$ such that $p \in S$
$\rightarrow S$ gets $(i_3,j_3)$

Results:

| | $j_1$ | $j_2$ | $j_3$ |
|---|---|---|---|
| $i_1$ | $(1,2)$ | $(2,3)$ | $(2,1)$ |
| $i_2$ | $(2,3)$ | $(1,1)$ | $(3,3)$ |
| $i_3$ | $(2,1)$ | $(2,2)$ | $(1,2)$ |

$S=\{(i_1,j_1),(i_2,j_2),(i_3,j_3)\}$
$F=\emptyset$

The algorithm first assigns the free officer to be $i_1$, and $j$ to be $j_1$ because $i_1$ has a preference of 1 for $j_1$. There are no matches $p$ in $S$ so the match $(i_1,j_1)$ is added to $S$ and $i_1$ is no longer in the set of free officers $F$. In the second step, $i$ is assigned to the next free officer, $i_2$; $i_2$'s first preference is $j_2$, so $j = j_2$. $j_2$ is not part of a matching in the set $S$, so $(i_2, j_2)$ is added to $S$. $i_2$ is then removed from $F$. In the third step, $i = i_3$, and $j = j_3$ because that is $i_3$'s first preference. Unit $j_3$ is not part of a matching in the set $S$, so $(i_3, j_3)$ is added to $S$. The algorithm terminates because $F$ is empty, and the set $S$ is outputted. This particular solution is a stable solution, meaning there are no blocking pairs.

Figure 2 illustrates what Algorithm 2 does in an instance where two officers have the same preferences. Initially, the set $F = \{i_1, i_2\}$ and $S = \emptyset$. First, $i_1$, the first officer in $F$, proposes to his or her first preference, $j_1$. There are no matchings in $S$, so $j_2$ accepts and the match $(i_1, j_1)$ is added to $S$. Second, $i_2$, the next officer in $F$, proposes to his or her first preference, $j_1$. However, $j_1$ is already in $S$, and $j_1$ strictly prefers $i_2$ over $i_1$. As a consequence, officer $i_1$ is added to $F$ and removed from $j_1$'s preferences, and the match $(i_1, j_1)$ is removed from $S$ while $(i_2, j_1)$ is added. Third, $i_1$ is forced to propose to $j_2$, who is not in $S$, and the match is made. $F$ is empty, and the solution $S = \{(i_2, j_1), (i_1, j_2)\}$ is stable.

Step 1

| | $\rightarrow j_1$ | $j_2$ |
|---|---|---|
| $\rightarrow i_1$ | (1, 2) | (2, 1) |
| $i_2$ | (1, 1) | (2, 2) |

Step 2

| | $\rightarrow j_1$ | $j_2$ |
|---|---|---|
| $i_1$ | (1, 2) | (2, 1) |
| $\rightarrow i_2$ | (1, 1) | (2, 2) |

Step 3

| | $j_1$ | $\rightarrow j_2$ |
|---|---|---|
| $\rightarrow i_1$ | (1, 2) | (2, 1) |
| $i_2$ | (1, 1) | (2, 2) |

Figure 2: Example of Algorithm 2 when two officers have the same preferences.

### 3.4 Evaluation Metrics

To achieve any one of the research objectives from Chapter I, evaluation metrics must be established. The goal of this section is to prelude the sensitivity analysis tested in Chapter IV.

### 3.4.1 Current Metrics

HRC currently uses three metrics to evaluate the quality of their matching solutions:

- Product Preference (PP): $\sum_{i \in I} \sum_{j \in J} o_{ij} u_{ij} x_{ij}$
- Total Product Preference (TPP): $\sum_{j \in J} o_{ij} x_{ij} \quad \forall i \in I$
- Overall Product Preference (OPP): $\sqrt{\frac{TPP}{\text{Number of Pref'd Matches}}}$

Product preference (PP) calculates the sum of the products between each officer-unit matching. In the solution found in Table 3, the PP $= 1 * 2 + 1 * 1 + 1 * 2 = 5$. Total product preference (TPP) sums the officers matched preferences. In Table 3, TPP $= 1 + 1 + 1 = 3$. The overall product preference (OPP) is a metric specific to the current algorithm. The square root of TPP over the number of matches where officers and units preferred each other is calculated. Currently, officers and units do not necessarily need to state a preference for each opposing agent in the set, so OPP refers to the officers and units that did prefer each other.

PP and TPP will be calculated in Chapter IV; however, because OPP is specific to the current algorithm, a similar, but different metric will be used instead. The number of officers who receive a unit within their top three choices, which is also mentioned in the next section, will be used to replace OPP.

### 3.4.2 Other Metrics

In the scope of metrics, there are two main categories of interest: comparison metrics and quality metrics. The former will compare the current algorithm's solution to some baseline, while the latter focuses solely on the specific algorithm's solution. Comparison metrics:

- Number of officers receiving a different unit in current versus new model.

- Number of officers receiving a less preferred unit.
- Number of officers receiving a more preferred unit.
- Average change in different matchings.

Quality metrics:

- Officer cost: sum of matched preferences that officers receive. This is equivalent to TPP.
- Unit cost: sum of matched preferences that units receive.
- Number of officers receiving a first choice.
- Number of officers receiving a top three choice.
- Average preference of unit received by officer.
- Highest preference an officer receives, or $q$.
- Number of blocking pairs.

Table 5: Metrics for Table 2.

| | MIP $w_o = 1$, $w_u = 0$, $w_q = 1$ | MIP $w_o = 1$ $w_u = 0$ $w_q = 0$ | MIP $w_o = .5$ $w_u = .5$ $w_q = 1$ | MIP $w_o = .5$ $w_u = .5$ $w_q = 0$ | MIP $w_o = 0$ $w_u = 1$ $w_q = 1$ | MIP $w_o = 0$ $w_u = 1$ $w_q = 0$ | SMT |
|---|---|---|---|---|---|---|---|
| Officer Cost | 3 | 3 | 3 | 3 | 3 | 5 | 3 |
| Unit Cost | 5 | 5 | 5 | 5 | 3 | 3 | 5 |
| PP | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Num Off Rec'd 1st | 3 | 3 | 3 | 3 | 3 | 1 | 3 |
| Num Officers Rec'd Top 3 | 3 | 3 | 1 | 1 | 1 | 3 | 3 |
| Off Avg Pref Rec'd | 1 | 1 | 1 | 1 | 1 | 1.67 | 1 |
| q | 1 | 1 | 1 | 1 | 1 | 2 | 1 |
| Blocking Pairs | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### 3.5 Summary

Quantifying these metrics to the mixed integer program and stable marriage algorithms gives HRC insight on how valuable modifying their current algorithm may be. It provides clear differences, making it easy to see the impact of this research. In the next section, these models and metrics are applied to generated data sets in order to analyze the impact of current and new models.

# IV. Results and Analysis

## 4.1 Introduction

This chapter provides and discusses results for MIP models using randomly generated officer and unit preference data. Each data set tested uses 100 officers and 100 units. Section 4.2 and Section 4.3 each test the data using a model varied by twenty-five different sets of weights. The weights associated with the three components of the MIP objective function: officer preference, unit preference, and maximum officer disappointment (minimax). Section 4.2 uses a data set with strict ordinal preferences for both officers and units. Section 4.3 uses a data set permitting officers to be indifferent but still requiring units to produce strict preferences.

Section 4.4 contains additional results and statistical inference on metrics for select models, chosen based on the results from Section 4.2 and Section 4.3, obtained using 25 replications of randomly-generated data. In all cases, a baseline SMA model is created from the same data as used by the MIP models for use as a baseline for comparison metrics.

The SMA and SMT algorithms produced the same results whether the input preference data contained ties or not. Without ties, this result is not a surprise. When considering ties, it is less intuitive, but still explainable. Because units must input preferences in a strict ranking, the successive officers in a unit's preference list are deleted in the same manner for both algorithms. This means that each time a new match is made in each algorithm, the same steps are taken to ensure less preferred officers are not matched with the matched unit in future iterations. For this reason, the choice of SMA or SMT as the baseline model has no effect on results or conclusions.

## 4.2 Without Ties

This study's first research objective is accomplished by replicating the SMA baseline with a MIP. Figure 3 shows the results from different weights applied in the MIP. The figure has five main subplots being tested against different weights on $q$. Each subplot tests different weights on officer and unit preferences. Proceeding from left to right along the x-axis, weight on officer preferences decreases while weight on unit preferences increases. The dark blue and light blue bars depict the summed officer and unit matching preference cost, respectively. The brown bars depict the baseline for officer cost and total cost. As a reminder, lower costs are preferred.

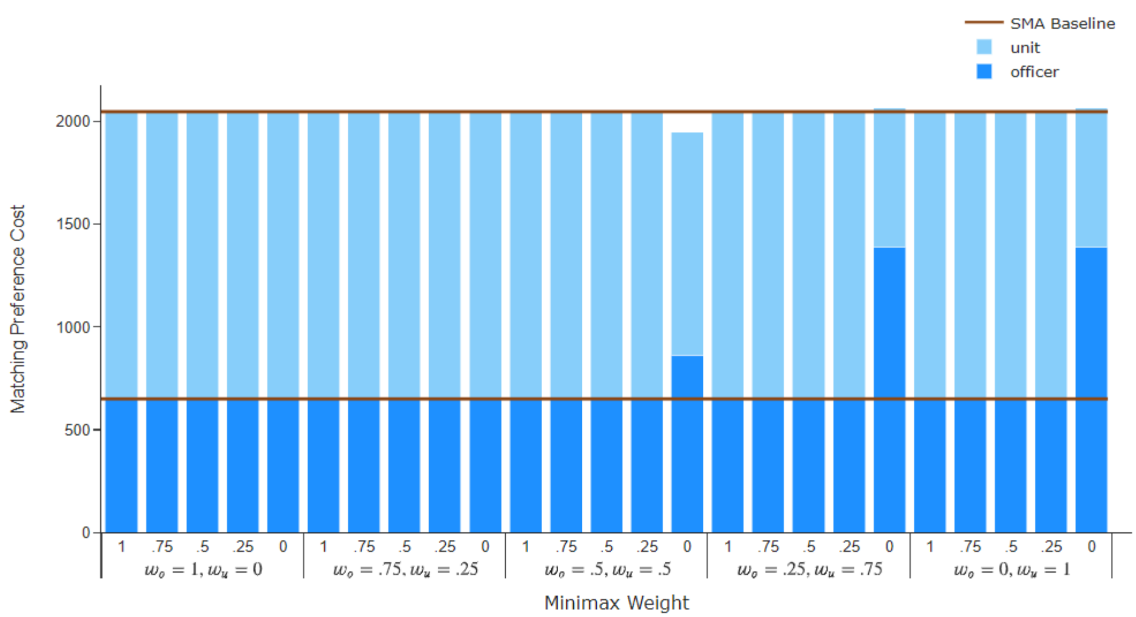


Figure 3: SMA solutions are replicated by the majority of MIP solutions in the no-ties data set.

As seen in Table 6, weights of $w_o = 1, w_u = 0, w_q = 0$ result in an exact replication of the baseline, given the mechanics of SMA. Additionally, Figure 3 shows that a majority of the MIP instances return the same results as the base case. The only deviation from the SMA is when $w_q = 0$ in each of the last three subplots. When there is zero value placed on the minimax constraint and the weight placed on the officers'

Table 6: Metrics for the no-ties data set.

| | SMA Baseline | MIP $w_o = .5$, $w_u = .5$, $w_q = 0$ | MIP $w_o = .25$ $w_u = .75$ $w_q = 0$ | MIP $w_o = 0$ $w_u = 1$ $w_q = 0$ |
|---|---|---|---|---|
| Officer Cost (TPP) | 650 | 863 | 1387 | 1387 |
| Unit Cost | 1398 | 1084 | 679 | 679 |
| PP | 9193 | 9897 | 10022 | 10022 |
| Num Officers Rec'd 1st | 16 | 13 | 7 | 7 |
| Num Officers Rec'd Top 3 | 42 | 35 | 25 | 25 |
| Officer Avg Pref Rec'd | 6.5 | 8.63 | 13.87 | 13.87 |
| q | 30 | 33 | 79 | 79 |
| Blocking Pairs | 0 | 0 | 0 | 0 |

preferences does not exceed 0.5, a different solution is produced. In these solutions, officers are generally matched with less preferred units, but units are matched with more preferred officers.

An additional model with weights of $w_o = 0, w_u = 0, w_q = 1$ determines what happens when weight is placed only on the minimax constraint. It is found that the results of this model is the exact same as the baseline and, consequently, the majority of the MIP models.

In order to assess the quality of matchings, Figure 4 plots product preference (PP) against each set of weights. The three sets of weights that deviate from the baseline in Figure 4 are the same as those that deviate in Figure 3. PP is noticeably higher in those three cases, indicating that the overall officer-unit matchings are higher than those in the baseline, where higher preferences indicate less-desired destinations. Even if unit satisfaction is higher, it does not offset the officer dissatisfaction in the PP calculations.

Table 6 displays the specific differences between the SMA baseline and the deviated MIP instances. The SMA baseline column represents all MIP instances where the solutions are equivalent, and it is compared against all deviated MIP instances. A higher PP from Figure 4 implies the other metrics tested are worse. Fewer officers have

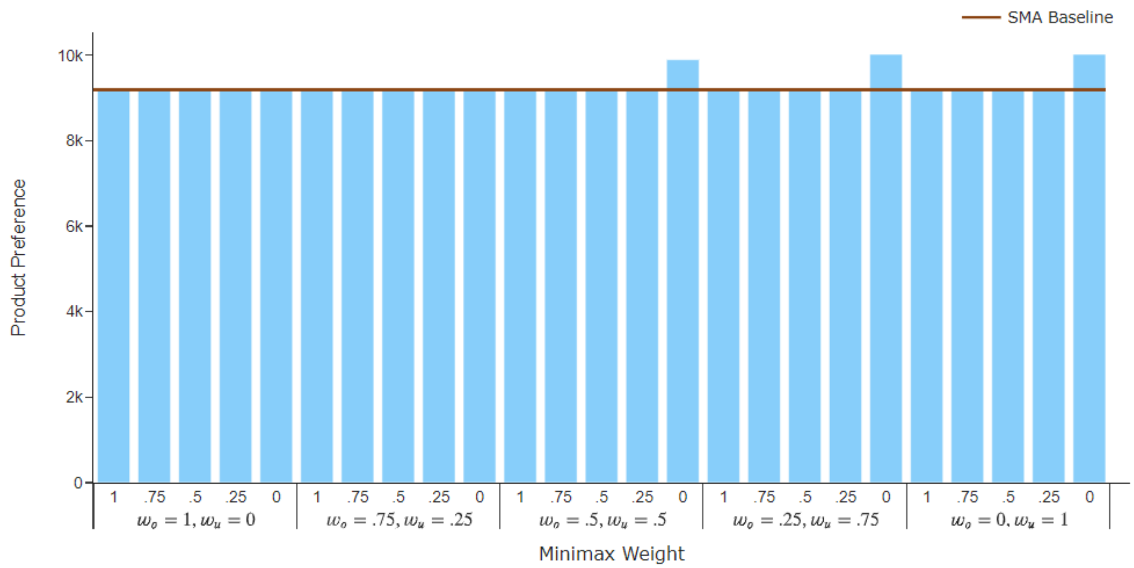


Figure 4: Stacked bar plot for product preference versus variable weights.

a first choice or top three match in the instances where PP is higher. The average officer preference match and the $q$ constraint also output higher results, meaning officers will generally be less satisfied with the matched units. The number of officers receiving first or top three choice are lower, which also shows inferior metrics to the SMA. The highest preference an officer receives is higher in the MIP instances, which is expected because the weight of the minimax constraint is zero. Lastly, the number of blocking pairs is zero in each scenario, indicating the solutions produced in each algorithm are stable.

Table 7 contains comparison metrics for the different scenarios. Like in Table 6, SMA and all equivalent solutions are represented in the first column. Table 7 shows that any solution not equivalent to the baseline produces a worse outcome for the officers. The deviated MIP instances assign officers to less preferred units, and no officers get a more preferred unit, which also correlates to the average officer preference increase.

Table 7: MIP solutions different than the baseline are unfavorable to officers with the no-ties data set.

| (Officer Weight, Unit Weight, Minimax Weight) | (1, 0, 0) | (.5, .5, 0) | (.25, .75, 0) | (0, 1, 0) |
|---|---|---|---|---|
| Officers Rec Different Unit | 0 | 21 | 47 | 47 |
| Officers Rec Less Pref'd Unit | 0 | 21 | 47 | 47 |
| Officers Rec More Pref'd Unit | 0 | 0 | 0 | 0 |
| Officer Pref Avg Change | 0 | +10.14 | +15.68 | +15.68 |

### 4.3 With Ties

In the no-ties data set, it is easy to replicate the SMA Baseline with a MIP; however, a data set with ties does not share that characteristic. Figure 5 shows the baseline solution with the dark horizontal lines and how it compares to the instances of the MIP, and Table 8 shows a more detailed insight on how the different algorithms perform.

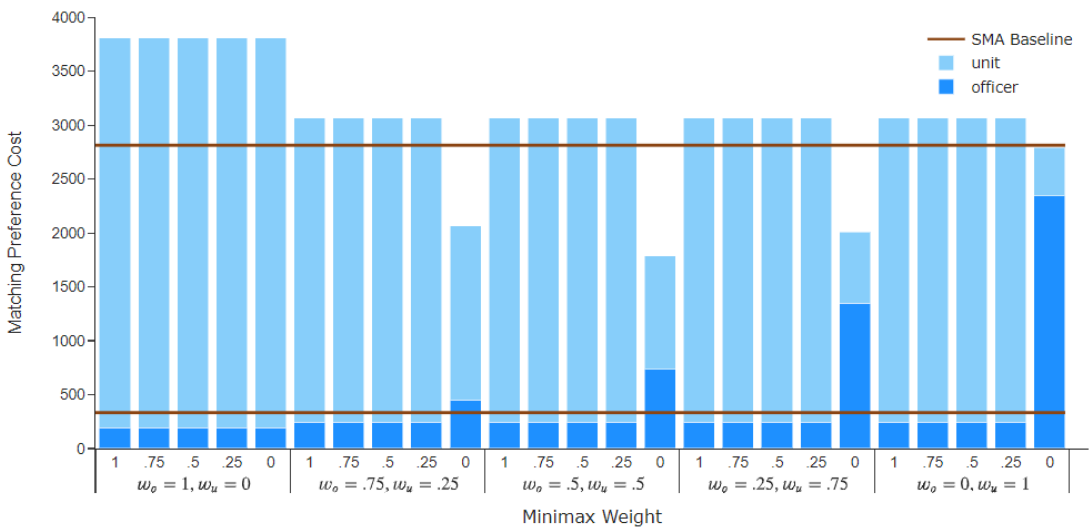


Figure 5: Minimax constraint is the only varying contributor to MIP solutions in each subplot.

The first observation in Figure 5 is that the officer cost is lower in the majority of the MIP instances compared to the baseline. The only instances where the MIP fails to perform better for officer cost is when $w_q = 0$. The first subplot gives complete weight to officer preferences and zero weight to unit preferences, and the effect is that minimax weight has no effect on either officer or unit cost. Observing from left to right along the x-axis, less weight is placed on officer preferences and more on unit preferences. In this regard, Figure 5 displays a similar trend as Figure 3 from the no-ties data set. Officer cost increases and unit cost decreases, specifically when $w_q = 0$. Otherwise, the majority of solutions are equivalent. Interestingly, the last four subplots produce exactly the same solutions, except when $w_q = 0$. This may imply that any weight on $w_q$ forces a specific solution; however, future work is necessary to substantiate this claim.

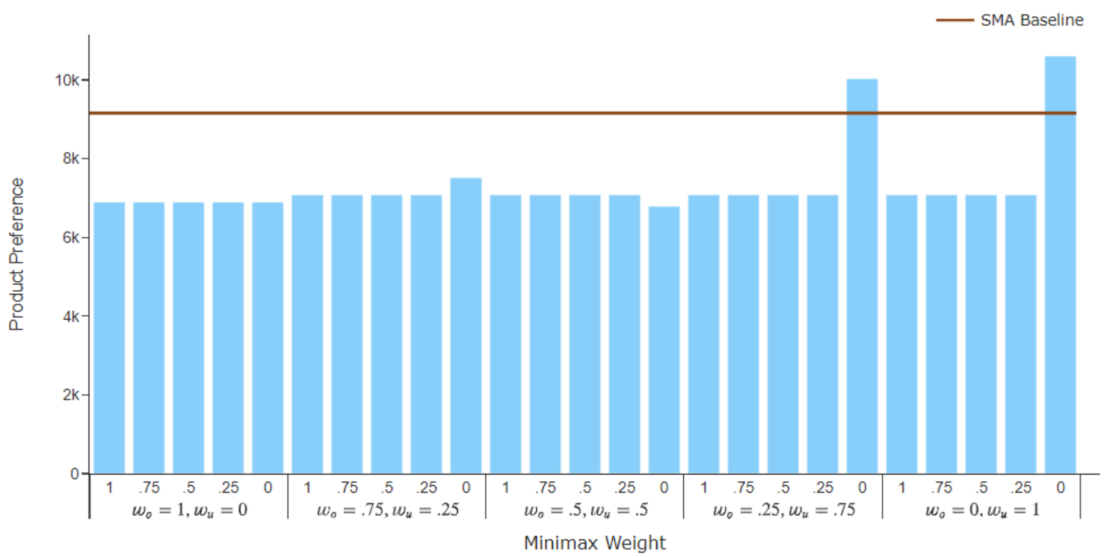


Figure 6: Majority of MIP PP Perform Better than SMA

Figure 6 plots PP against different weights for the ties data set. Again, the SMA baseline is shown with the brown bar, which is slightly lower relative to most of the MIP instances. Because many of the 25 models produce the same solution, the set of models with identical PP scores in Figure 6 is identical to the set of models with identical officer or unit preferences in Figure 5. The only deviations from each subplot occur when $w_q = 0$. Specifically when $w_q = 0$, the PP is slightly lower in the third subplot but higher in all others. In the second subplot, the model with $w_q = 0$ has a higher PP than the rest of its subplot, but a lower PP than the baseline. The third subplot is interesting in that it has a smaller PP than the rest of its subplot where $w_q = 0$. The implication is that, by equally weighting officer and unit preferences, the improvement in the outcome from the unit's perspective slightly outweighs the degradation in the outcome from the officer's perspective where compared to the baseline in which only officer preferences are considered. The fourth and fifth subplots where $w_q = 0$ both carry a higher officer cost and higher PP compared to the baseline. The implication is that, when unit preferences are weighted more heavily than officer preferences, the ensuing trade-off is such that the reduction in officer satisfaction is greater, in terms of overall PP, than the increase in unit satisfaction. However, this is only the case with $w_q = 0$. Applying any nonzero weight to $w_q$ eliminates this phenomenon. More specific differences in this particular data set against different models are shown in Table 8.

Table 8 includes a variation not seen in the previous two figures. The MIP variation where $w_o = 0$, $w_u = 0$, and $w_q = 1$ is shown in the right-most column. This is added in effort to quantify the effect of what placing weight only on $w_q$ does to the solution. This variation does not produce the best metrics for officers. In fact, the second column, $(1, 0, 1)$ has the best outcome for officers regarding cost, first preference, top three choices, and average preference received. While $(0, 0, 1)$ does have relatively better metrics than all other variations, it is inferior to $(1, 0, 1)$ because it is only concerned with the greatest preference an officer receives while $(1, 0, 1)$ is doing that in addition to minimizing officer cost. This also implies that $(0, 0, 1)$ has alternative optimal solutions; every solution where $q = 7$ is an alternative optimal solution for this data set. An advantage to running the variation $(0, 0, 1)$ identifies what the best value of $q$ a solution can produce.

For the remainder of this section, MIP instances where solutions are equivalent will be represented with one instance. Specifically, the instance where $(w_o = 1, w_u =$

Table 8: Metrics for the ties data set.

| | SMA Baseline | MIP $w_o = 1$ $w_u = 0$ $w_q = 1$ | MIP $w_o = .75$ $w_u = .25$ $w_q = 1$ | MIP $w_o = .75$ $w_u = .25$ $w_q = 0$ | MIP $w_o = .5$ $w_u = .5$ $w_q = 0$ | MIP $w_o = .25$ $w_u = .75$ $w_q = 0$ | MIP $w_o = 0$ $w_u = 1$ $w_q = 0$ | MIP $w_o = 0$ $w_u = 0$ $w_q = 1$ |
|---|---|---|---|---|---|---|---|---|
| Officer Cost | 333 | 191 | 246 | 450 | 733 | 1350 | 2346 | 204 |
| Unit Cost | 2481 | 3718 | 2823 | 1616 | 1054 | 661 | 447 | 3633 |
| PP | 9153 | 7220 | 7082 | 7519 | 6789 | 10033 | 10605 | 7623 |
| Officers Rec'd 1st | 41 | 62 | 47 | 32 | 22 | 12 | 7 | 56 |
| Officers Rec'd Top 3 | 68 | 87 | 76 | 55 | 39 | 22 | 12 | 85 |
| Officer Avg Pref | 3.33 | 1.91 | 2.46 | 4.5 | 7.33 | 13.5 | 23.46 | 2.04 |
| q | 17 | 7 | 7 | 22 | 37 | 67 | 86 | 7 |
| Blocking Pairs | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$0, w_q = 1)$ will represent all the solutions where $w_o = 1$ and $w_u = 0$ because all of these solutions are equivalent. Instance $(w_o = .75, w_u = .25, w_q = 1)$ will represent the remainder of solutions where $w_q \neq 0$. The remainder of the solutions are unique and will be represented individually.

Table 9 gives comparison metrics for the different solutions produced in the data set with ties. The first two columns, corresponding to weights of $(1, 0, 1)$ and $(.75, .25, 1)$, show general improvement from the SMA to MIP, with more officers receiving more favorable units, and the average preference matching decreasing. Although $(.75, .25, 1)$ assigns 11 officers to less favorable units, 22 are assigned more favorably, and the average preference matching still decreases. Overall, the first two columns are superior to the SMA from an officer satisfaction perspective. The rest of the instances show an increase in average preference, and few officers are assigned a more desired unit while many are assigned less favorably.

Table 9: Each column shows the officers' changes from the SMA baseline to the MIP instance. The majority of MIP instances produce more favorable officer preferences. The green cells represent more favorable outcomes for officers. The red shows less favorable outcomes.

| | (1, 0, 1) | (.75, .25, 1) | (.75, .25, 0) | (.5, .5, 0) | (.25, .75, 0) | (0, 1, 0) |
|---|---|---|---|---|---|---|
| Different Unit | 48 | 44 | 37 | 59 | 80 | 88 |
| Less Pref'd Unit | 0 | 11 | 28 | 51 | 75 | 86 |
| More Pref'd Unit | 34 | 22 | 3 | 1 | 0 | 0 |
| Pref Avg Change | -2.95 | -1.97 | +3.16 | +6.78 | +12.71 | +22.87 |

In order to show a pattern of trends similar to Figure 5 and Figure 6, Section 4.3

and Section 4.3 are shown. Section 4.3 follows same trends as Figure 5. The models where $w_q = 0$ do not follow the rest of its respective subplot. Another shared trait is that the first subplot, where $w_o = 1$ and $w_u = 0$, has a higher overall cost to the majority of solutions, but the lowest officer cost. PP in the MIP is higher than the baseline for all 25 models tested, which is different than Figure 6. However, the following section will show that PP is not consistently significant.

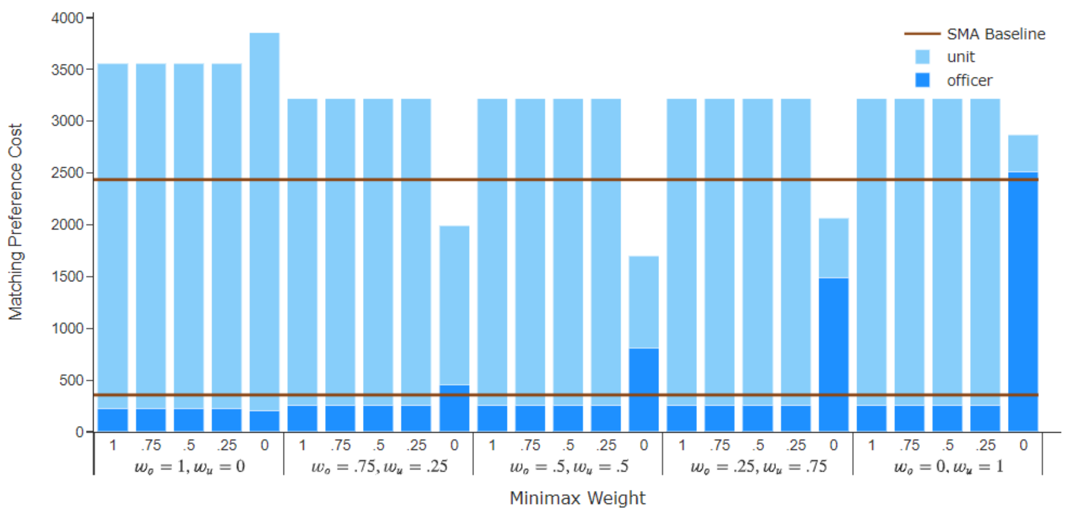


Figure 7: Plot of weights versus matching preference cost. Each subplot has identical solutions where $w_q \neq 0$.

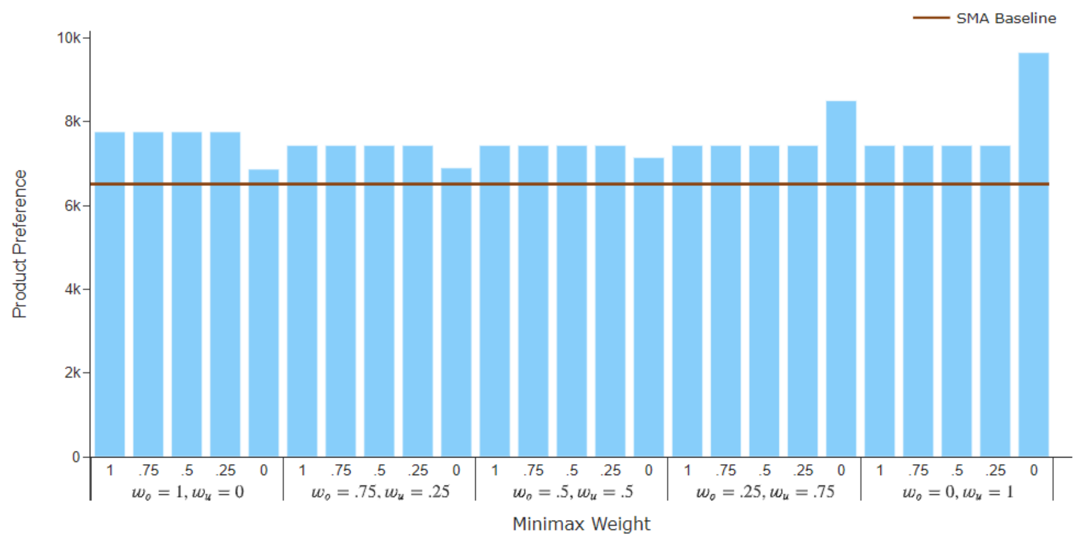


Figure 8: Plot of weights versus PP. Each subplot has the same PP where $q \neq 0$.

## 4.4 Statistical Inference

In the preceding sections, select models produced different values for the set of metrics relative to the base case. The purpose of this section is to examine whether those differences are statistically significant by performing a series of hypothesis tests. The following metrics are tested for significance: officer cost, product preference, number of officers who receive first choice, number of officers who receive top three, average officer preference matching, and $q$.

MIP instances $(1, 0, 0)$, $(.75, .25, 1)$, and $(.5, .5, 0)$ each produce solutions with calculated metrics whose difference from the baseline merits checking for statistical significance. Each model is run using 15 different sets of randomly-generated data with ties. With limited time to run the different data sets, 15 is an appropriate number given each metric is normally distributed for that weight. The null hypothesis for all of the hypotheis tests is that the SMA baseline and MIP metric values are equal. There are 18 total tests being conducted at the significance level, $\alpha$, of

0.05. Table 10, Table 11, and Table 12 show results for two-tailed two-sample t-tests assuming unknown but unequal variances for objective function weights of $(1, 0, 1)$, $(.7, .25, 1)$, and $(.5, .5, 0)$, respectively.

Asterisks in the rightmost column indicate that the p-value for the test is less than the significance level, controlling for Type I error using the Bonferroni adjustment with $c = 6$ hypothesis tests per model. In these cases, the null hypothesis is rejected and the conclusion is that the metric in the MIP model is statistically lower (or higher) than the metric in the SMA model. Almost all comparisons are significant except PP in Table 10 and Table 12. Therefore, in all comparisons except the two mentioned, the conclusion is that the perturbations of objective function weights has a significant effect on the model's performance metrics.

Table 10: Two-sample t-test assuming unequal variances run on a sample size of 15 data sets with ties for weights $(1, 0, 1)$.

| | Mean Baseline | Mean $(1, 0, 0)$ | Variance Baseline | Variance $(1, 0, 0)$ | P(T ≤ t) two-tail |
|---|---|---|---|---|---|
| Officer Cost (TPP) | 412.6 | 285.33 | 2725.97 | 3588.38 | 1.24E-06 * |
| PP | 9253.2 | 8398.73 | 1660786.7 | 1428621 | 0.070 |
| Num Officers Rec'd 1st | 32.8 | 43.46 | 32.74 | 90.41 | 0.0011 * |
| Num Officers Rec'd Top 3 | 57.6 | 71.4 | 32.11 | 84.54 | 2.80E-06 * |
| Officer Avg Pref Rec'd | 4.13 | 2.75 | 0.27 | 0.33 | 1.84E-07 * |
| q | 20.2 | 11.67 | 31.46 | 13.52 | 4.992E-05 * |

Table 11: Two-sample t-test assuming unequal variances run on a sample size of 15 data sets with ties for weights $(.75, .25, 1)$.

| | Mean Baseline | Mean $(.75, .25, 1)$ | Variance Baseline | Variance $(.75, .25, 1)$ | P(T ≤ t) two-tail |
|---|---|---|---|---|---|
| Officer Cost (TPP) | 412.6 | 285.33 | 2725.97 | 3101.7 | 9.77E-05 * |
| PP | 9253.2 | 8138.27 | 1660786.7 | 1189011 | 0.016 * |
| Num Officers Rec'd 1st | 32.8 | 37.8 | 32.74 | 38.6 | 0.029 * |
| Num Officers Rec'd Top 3 | 57.6 | 64.4 | 32.11 | 50.69 | 0.0074 * |
| Officer Avg Pref Rec'd | 4.13 | 3.23 | 0.2726 | 0.31 | 9.77E-05 * |
| q | 20.2 | 11.67 | 31.46 | 13.52 | 4.992E-05 * |

Table 12: Two-sample t-test assuming unequal variances run on a sample size of 15 data sets with ties for weights $(.5, .5, 0)$.

| | Mean Baseline | Mean $(.5, .5, 0)$ | Variance Baseline | Variance $(.5, .5, 0)$ | P(T $\leq$ t) two-tail |
|---|---|---|---|---|---|
| Officer Cost (TPP) | 412.6 | 885.93 | 2725.97 | 18446.4 | 2.29E-10 * |
| PP | 9253.2 | 8385.47 | 1660786.7 | 1190406 | 0.057 |
| Num Officers Rec'd 1st | 32.8 | 16.47 | 32.74 | 31.12 | 1.27E-08 * |
| Num Officers Rec'd Top 3 | 57.6 | 31.67 | 32.11 | 51.52 | 1.83E-11 * |
| Officer Avg Pref Rec'd | 4.13 | 8.86 | 0.27 | 1.84 | 2.29E-10 * |
| q | 20.2 | 43.73 | 31.46 | 49.21 | 1.03E-10 * |

### 4.5 Time Processing

The processing times from the data set from Section 4.3 and one of the data sets from Section 4.4 are used to determine what the average processing time is for each model. The mean for all data sets is 88.31 seconds with a standard deviation of 5.56 seconds. Larger scale problems will exponentially take longer, and is the reason each data set was limited to 100 officers and 100 units. The stable marriage algorithm took 2.08 seconds in the data set used in Section 4.3, and 1.61 seconds in the data set from Section 4.4. The computer used to run these models has a 11th Gen Intel Core i7-1165G7 processor with a 16 GB RAM in a Python 3.9.7 environment.

# V. Conclusions

## 5.1 Research and Impact

This research accomplishes three objectives. The first is to replicate results of the current SMA-based model with a MIP. The objective allows an analyst to conduct tailored impact analysis of proposed policy changes that are not as flexibly incorporated into the SMA. The remaining two objectives consist of specific potential adjustments to the current process.

The second objective analyzes the effects on officer assignment when incorporating priorities other than officer satisfaction into the objective function. Specifically, unit satisfaction and maximum disappointment experienced by an individual officer are quantified and then weighted with 25 different combinations of weights that represent the varied emphasis that might be given to the three priorities. The conclusion is that any weight placed on the maximum disappointment, $q$, pushes the solution in favor of officer preferences. Additionally, the different weighted models produced significant differences to the baseline. In the majority of solutions, officer satisfaction generally improved.

The third objective analyzes the effect of allowing officers to assign equal preference to more than one prospective destination. Allowing ties gives the algorithm more flexibility to give an officer a desired unit, which in turn improves officer satisfaction.

## 5.2 Future Research

This research allowed officers to be indifferent, but not units. No testing was done to determine if feasible solutions are produced when units are also able to submit ties in SMA or MIP. Testing these models, or perhaps developing new models, to allow both sides to submit ties in preferences would be a value added extension to this research.

Additionally, the MIP could be tailored to the specialized needs of select units. Adding constraints that consider different skills or requirements would appeal to technical branches. It would also reduce the number of manual changes HRC has to make to the pure-market solution to satisfy unique requirements such as the Exceptional Family Member Program (EFMP) or Married Army Couples Program (MACP). Currently, these programs require manual review and adjustment to the model output followed by a rerunning of the model to reassign any broken matches.

# Bibliography


1. Active duty military personnel by rank/grade and service. Technical report, Department of Defense, July 2022.

2. Mokhtar S Bazaraa, John J Jarvis, and Hanif D Sherali. *Linear programming and network flows*. John Wiley & Sons, 2011.

3. Zubeyir Cimen. A multi-objective decision support model for the turkish armed forces personnel assignment system. Master's thesis, Air Force Institute of Technology, 2001.

4. Robert J Dakin. A tree-search algorithm for mixed integer programming problems. *The computer journal*, 8(3):250–255, 1965.

5. Alexandra K. DeAngelis and Christopher B. Fisher. Manpower Analyst, Officer Personnel Management Directorate Army Human Resources Command. 26 April 2022.

6. Matthew D Ferguson, Raymond Hill, and Brian Lunday. A scenario-based parametric analysis of the army personnel-to-assignment matching problem. *Journal of Defense Analytics and Logistics*, 4(1):89–106, 2020.

7. David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

8. Martin Grötschel and George L Nemhauser. George dantzig's contributions to integer programming. *Discrete Optimization*, 5(2):168–173, 2008.

9. Robert W Irving. Stable marriage and indifference. *Discrete Applied Mathematics*, 48(3):261–272, 1994.

10. Ailsa H Land and Alison G Doig. An automatic method for solving discrete programming problems. 2010.

11. David F Manlove. Hospitals/residents problem. In Ming-Yang Kao, editor, *Encyclopedia of Algorithms*, chapter 180, pages 390–394. Springer, New York, 2008.

12. David F Manlove, Robert W Irving, Kazuo Iwama, Shuichi Miyazaki, and Yasufumi Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261–279, 2002.

13. Alan S Manne. A target-assignment problem. *Operations research*, 6(3):346–351, 1958.

14. Association of the United States Army. Most officers get top choice in new assignment process. https://www.ausa.org/news/most-officers-get-top-choice-new-assignment-process, January 2020. Accessed: 10.04.2022.

15. Alvin E 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.

# REPORT DOCUMENTATION PAGE

*Form Approved OMB No. 0704–0188*



| 1. REPORT DATE *(DD–MM–YYYY)* | 2. REPORT TYPE | 3. DATES COVERED *(From — To)* |
|---|---|---|
| 23–03–2023 | Master's Thesis | September 2021 — March 2023 |

**4. TITLE AND SUBTITLE**

The U.S. Army Officer-to-Unit Assignment Problem

**5a. CONTRACT NUMBER**

**5b. GRANT NUMBER**

**5c. PROGRAM ELEMENT NUMBER**

**6. AUTHOR(S)**

Phillips, Andrea, 2d Lt, USAF

**5d. PROJECT NUMBER**

**5e. TASK NUMBER**

**5f. WORK UNIT NUMBER**

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

**8. PERFORMING ORGANIZATION REPORT NUMBER**

AFIT-ENS-MS-23-M-151

**9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

U.S. Army Human Resources Command
Major Alexandra DeAngelis
1600 Spearhead Division Ave, Room 2-2-013
Fort Knox, KY 40122
Email: Alexandra.k.deangelis.mil@army.mil

**10. SPONSOR/MONITOR'S ACRONYM(S)**

HRC

**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

**12. DISTRIBUTION / AVAILABILITY STATEMENT**

Distribution Statement A.
Approved for Public Release; Distribution Unlimited.

**13. SUPPLEMENTARY NOTES**



**14. ABSTRACT**

Every two to three years, U.S. Army officers must change duty stations, which entails a selection process based on preferences. Currently, officers are assigned to units using a stable-marriage algorithm. Two impracticalities occur within this process. First, officers are required to submit strictly ranked preferences, not allowing indifference among units. Second, the stable-marriage algorithm does not give flexibility to alternative priorities. This research focuses on two modifications to the current model. First, a mixed integer program is created that allows the user, U.S. Army Human Resources Command, to consider other priorities: unit preferences and maximum officer disappointment. Second, generated data allowing officer indifference is tested and compared to the stable-marriage baseline. These solutions are tested using 25 models, each created by perturbing various parameters. Most models produce equivalent solutions, with differences stemming from placing no priority on maximum officer disappointment. These results are quantified and tested using metrics measuring officer satisfaction, providing insightful knowledge to decision makers weighing policy changes in this process.

**15. SUBJECT TERMS**

optimization, stable marriage, mixed integer programming, assignment problem, deferred acceptance

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Dr. Phillip LaCasse, AFIT/ENS |
| | | | | | 19b. TELEPHONE NUMBER *(include area code)* |
| U | U | U | UU | 55 | (937) 255-3636 x4318 phillip.lacasse@afit.edu |