NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

SIMULATION OF ENLISTED SAILOR ASSIGNMENT PROCESS TO EXPLORE THE COST OF NON-MONETARY INCENTIVES

by

Andreas Buelow

March 2010

Thesis Advisors:	William R. Gates Rachel T. Johnson

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** March 2010	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Simulation of Enlisted Sailor Assignment Process to Explore the Cost of Non-Monetary Incentives			**5. FUNDING NUMBERS**
6. AUTHOR(S) Andreas Buelow			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number _____________.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Distribution Statement			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

The attractiveness of an employer became more important for employees over time. It influences the decision to reject a job or to stay in the job. Incentives are useful to motivate employees to do their job and to stay with their employer. The U.S. Navy uses many different incentives to motivate sailors to stay in the Navy.

The Navy spends the biggest amount of money to motivate enlisted sailors to stay in the Navy for the reenlistment bonuses. The budget for this reenlistment bonus increased over time, but it is still difficulty to meet the end-strength goals. The use of Non-Monetary Incentives (NMIs) becomes more interesting to stop the increasing in budget of reenlistment bonuses.

The use of NMIs only makes sense if the sailor values these incentives higher than it costs the Navy. Therefore, the Navy has to know how much an NMI cost.

To identify the cost of NMIs that are related to the assignment process, this research uses a simulation model. The simulation model simulates the assignment process and is used to identify differences in assignment related costs. In different scenarios, the single and combined use of NMIs was tested.

14. SUBJECT TERMS Assignment Simulation Model, Non-Monetary Incentives, Reenlistment,			**15. NUMBER OF PAGES** 115
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 8-98) Prescribed by ANSI Std. Z39.18

THIS PAGE INTENTIONALLY LEFT BLANK

Appropriate for public release; distribution unlimited

SIMULATION OF ENLISTED SAILOR ASSIGNMENT PROCESS TO EXPLORE THE COST OF NON-MONETARY INCENTIVES

Andreas Buelow
Commander Junior Grade, German Navy
Graduate Engineer, Military Academy Hamburg, 1996

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
March 2010

Author: Andreas Buelow

Approved by: William R. Gates
Thesis Advisor

Rachel T. Johnson
Co-Advisor

William R. Gates
Dean, Graduate School of Graduate School of Business and Public Policy

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The attractiveness of an employer became more important for employees over time. It influences the decision to reject a job or to stay in the job. Incentives are useful to motivate employees to do their job and to stay with their employer. The U.S. Navy uses many different incentives to motivate sailors to stay in the Navy.

The Navy spends the biggest amount of money to motivate enlisted sailors to stay in the Navy for the reenlistment bonuses. The budget for this reenlistment bonus increased over time, but it is still difficulty to meet the end-strength goals. The use of Non-Monetary Incentives (NMIs) becomes more interesting to stop the increasing in budget of reenlistment bonuses.

The use of NMIs only makes sense if the sailor values these incentives higher than it costs the Navy. Therefore, the Navy has to know how much an NMI cost.

To identify the cost of NMIs that are related to the assignment process, this research uses a simulation model. The simulation model simulates the assignment process and is used to identify differences in assignment-related costs. In different scenarios, the single and combined use of NMIs was tested.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank my advisors, Dr. William R. Gates and Dr. Rachel T. Johnson. Your guidance and insights motivated me. Thanks for your patience.

Thanks to all friends, who made my time in the United States such a wonderful experience.

Many thanks to my parents. You supported me wherever you could. Mum and Dad, it is good to know you are behind me. I always can count on you. Thanks for all the time you spent with my kids while I was so far away.

Thanks to my son, Arni, and my daughters, Anna and Julia. Two years of studies in the USA, without being with you is a long and hard time. You have been always in my mind. Your smiles and your love was the best support I could get. I certainly could not have done this without your understanding.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

On December 31, 2009, 329,518 sailors were serving in the U.S. Navy. Of this number, 273,448 sailors were enlisted. With this number of employees, the Navy is a big competitor on the labor market. To gain and to hold sailors, the Navy invests a lot on sailor recruiting and reenlistment. In the fiscal year (FY) 2009, the Navy spent $333 million on reenlistment bonuses (Department of the Navy, Fiscal year 2011, Budgets Estimates, 2010). In FY 1999, it spent $169 million. (Department of the Navy, Fiscal Year 2001, Budgets Estimates, 2000) Within ten years, the budget for reenlistment bonuses has almost doubled. In times of economic crises, further increases in budgets will be very limited.

Avoiding the excessive costs of bonuses while still meeting retention goals is a current focus area for the Navy. Non-monetary incentives can help to increase motivation and a sailor's willingness to stay in the Navy. A survey has shown that non-monetary incentives, such as homeport of choice, billet of choice or regional stability, as alternatives to or in combination with a bonus check might have a higher value to the sailor than they directly cost the U.S. Navy.

The mechanism used to offer the non-monetary incentives (NMIs) is the Combinatorial Retention Auction Mechanism (CRAM) (Zimmermann, December 2008). CRAM minimizes cost for the Navy in the reenlistment process. In this process, reenlistment bonuses, sailors' values for NMIs and cost for NMIs are included. The CRAM will be explained in more detail in Chapter II. For the right use of this mechanism, it is necessary to know the cost of the non-monetary incentives offered. This thesis evaluates the cost of non-monetary incentives that can be used in the CRAM and that have an impact on the assignment process.

B. GOALS

The primary goals of this thesis are to assess the cost incurred by the U.S. Navy in offering an NMI. Additionally the thesis evaluates the constraints placed on the detailer offering NMIs. The cost and detailer constraints are estimated by simulating the assignment process of E4 to E6 sailors.

C. SCOPE AND LIMITATIONS

The scope will include: (1) a review of current non-monetary incentives, (2) a description of the assignment process, (3) the creation of a simulation model that mimics the assignment process over time, and (4) a calculation of the cost of the assignment process with and without non-monetary incentives.

This thesis is limited to measurable effects of NMIs that are related to the assignment process. Furthermore, this thesis focuses on an enlisted environment. Individual sailor's values of NMIs and benefits such as increased motivation are not included in the simulation model.

D. METHODOLOGY

Costs are analyzed by simulating the detailing process. The simulation model will include a community of sailors and their assignments over a period of time. All necessary information about the sailors will be discussed. The simulation will consider promotion, attrition rates, retention rates and sea-shore rotation. The model will estimate assignment costs over time. This cost includes moving costs as well as pay grade mismatch cost. Both costs will be explained in Chapter II.

The calculated assignment costs will be compared before and after adding the possibility to use non-monetary incentives into the model. The results should indicate the effect of non-monetary incentives on the assignment process and whether they have a significant impact or not.

E. ORGANIZATION OF STUDY

The next chapter provides the reader with background information on non-monetary incentives. The chapter also introduces the ideas behind the CRAM and describes the detailing process.

In the third chapter, the simulation model is described. This chapter should give the reader a detailed impression how the results in Chapter IV are generated. Chapter V answers the research questions, summarizes the research and makes recommendations.

Finally, the appendix shows detailed summary statistics and a short explanation of the simulation model procedures with the written code.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

A. INCENTIVES

The main goal of this thesis is to identify the cost of Non-Monetary-Incentives (NMIs) that are related to the assignment process. Before focusing on the specific set of NMIs that are the focus of this research, incentives in general will be explained.

Incentives should motivate people to do something. Incentives are an instrument to create a motivational environment for employees. The next paragraph presents a motivation theory before specific incentives are introduced later.

1. The Two-Factor Theory

Answering the question what motivates employees to work or even stay with their employer is as complex as the human being itself. To understand how incentives work, the motivation theory of Frederick Herzberg is explained next. In 1959, Frederick Herzberg, an American psychologist, published his two-factor theory, also known as Herzberg’s motivation-hygiene theory. (Exams_tutor_business) Herzberg categorized work related factors into two parts. The first category is the hygiene factors. These factors do not motivate employees, but if they are missing, the absence of these factors could reduce motivation. The second category is the motivators. If these factors exist employees are motivated to do their job, they are satisfied. If not, employees are not only not satisfied, but not unmotivated. The following table shows the two factors and gives examples for each category.

Hygiene Factors	Motivators
Salary	Nature of Work
Job Security	Sense of Achievement
Working Conditions	Recognition
Level and Quality of Supervision	Responsibility
Company Policy and Administration	Personal Growth and Advancement
Interpersonal Relationships	

Table 1. Herzbergs Two-factor Theory (Exams_tutor_business)

The hygiene factors are seen as extrinsic factors and they represent the complete job environment. The motivators are intrinsic factors; factors related to the job content. According to Herzberg, both factors are equally important. If an incentive should encourage somebody to do something, the incentive is a motivator. Money is generally seen by Herzberg as a force that makes somebody to do something, it is not a motivator to do something well.

2. Incentives

Monetary incentives are problematic. Offering a bonus for the employee of the month can motivate employees to do their work well to earn this bonus; on the other hand, this bonus could transform co-workers to competitors, which can disturb the interpersonal relationship. This can reduce one hygiene factor, which would lead to dissatisfaction and lack of motivation (Kohn, 1993). A possible monetary incentive could be stock options. This kind of incentive can motivate all employees to do well.

3. Non-Monetary-Incentives

NMIs can have advantages for both employees and employers. Increasing employee motivation might increase the benefits for the employer. If the increase in benefits is bigger than the costs of NMIs given to the employees, this will be a win-win situation.

Examples of NMIs are retirement planning, job training, sabbaticals, professional development, feedback and flexible work schedules. A survey by the American Association of Retired Persons has shown that different generations desire slightly different NMIs. “Baby Boomers” (born between 1946 and 1964) have job training on the list of desired NMIs, while Generation X’ers (born between 1964 and 1981) do not. Instead, this generation desires professional development, which was not a priority in the former time (Nelson, 1999). Offering incentives became more important for an organization over time to be competitive on the labor market.

4. NMIs in the Navy

The U.S. Navy, as every other military branch, is a big competitor on the labor market. To be attractive, the Navy offers their sailors many NMIs. This section only will provide the reader with a small overview of exemplary NMIs that the Navy offers. The Navy presents itself as an attentive employer, provides sailors with free health care, and supports them with military housing and childcare. The Navy also uses a couple of feedback systems, such as exams and performance marks. Further, the Navy offers many different training and education programs, which help sailors develop their Navy careers or prepares sailors for a career in the civil labor market.

NMIs that are related to the assignment process will be introduced later in this chapter. The next section focuses on reenlistment and the use of the Combinatorial Retention Auction Mechanism.

B. REENLISTMENT

The U.S. Navy recruits enlisted sailors for a first four-year period of obligation. During these times, the Navy looses approximately half of the enlisted sailors (55% of the 1994 cohort). After this four-year obligation, the U.S. Navy offers reenlistment bonuses to encourage sailors to stay in the Navy. In 2009, the Navy spent $333 million on these reenlistment bonuses.(Department of the Navy, Fiscal year 2011, Budgets Estimates, 2010) As mentioned in Chapter I, this bonus money increased dramatically over the last several years. A further increase is not an attractive option at this time, which makes the Navy think about alternatives to motivate sailors to stay in the Navy. Increasing the attractiveness of the Navy in general is one option. Another option is offering NMIs in the reenlistment process. The last option leads to the idea of selling NMIs in an auction process. These NMIs, sold in combination with monetary bonuses, can be a combinatorial incentive package that makes sailors stay in the Navy. The idea behind the Combinatorial Retention Auction Mechanism (CRAM) is explained in the next section.

C. CRAM

Zimmermann (2008) did a survey to estimate sailor's values for various NMIs. Then she combined these NMIs with reenlistment bonuses and estimated the Navy's savings based on sailor values for NMIs in the reenlistment process. The mechanism she used is the Combinatorial Retention Auction Mechanism. "Under the CRAM, a retained Sailor receives a particular NMI only if he values the incentive more than it cost the Navy to provide. This eliminates the need to determine which incentives to offer. All incentives are offered to all Sailors and allocated to those whose value exceeds cost."(Zimmermann, December 2008)

Some of the NMIs used in CRAM are related to the sailor assignment process. These NMIs are described in the next section.

D. NMIS IN THE ASSIGNMENT PROCESS

This section introduces the NMIs that will be simulated in this research and explains the assignment process related costs.

1. Types of NMIs

Three different types of NMIs and possible combinations of these are related to the assignment process: Homeport of Choice, Platform of Choice, and finally Regional Stability.

a) Homeport of Choice

Homeport of Choice is the opportunity for a sailor to choose a homeport in which he would like to serve. If he chooses this opportunity as a reenlistment NMI, he gets the guarantee to stay the complete time of his next tour in this region. If the sailor needs a new billet during this tour, he will be assigned to a new billet at the same place.

b) Platform of Choice

With Platform of Choice, the sailor is guaranteed to serve on the type of platform he has chosen. It does not matter where the platform is located. If a type of platform is available in different homeports, it is up to the detailer to choose the sailor's homeport.

c) Regional Stability

The last option is the Regional Stability. Taking this NMI allows the sailor to stay in a region for two tours. There is also a possibility to increase the number of tours, but this research concentrates on the opportunity to stay for two tours.

d) Tested Types and Combinations of Nmis

The simulation model designed to identify costs of NMIs estimated the costs of six different NMI choices and combinations. These types are based on the described NMIs above. Table 2. provides an overview of the NMIs that are simulated in this research.

Tested Type	NMI/NMI combination
Type 1	Homeport of Choice
Type 2	Platform of Choice
Type 3	Homeport of Choice and Platform of Choice
Type 4	Regional Stability
Type 5	Homeport of Choice and Regional Stability
Type 6	Homeport of Choice and Platform of Choice and Regional Stability

Table 2. Overview of NMIs used in simulation model

2. Assignment Related Costs

The goal of this thesis is to identify cost of NMIs that can be offered for reenlistment and have an impact on the assignment process. Explanations for the costs that are measured with the simulation model follow.

a) PCS Costs

The PCS cost depends on the distance the sailor has to move (distance between old and new billet) and the weight he is allowed to move, which depends on the pay grade.

b) Mismatch Cost

Under perfect conditions, every sailor in the Navy organization would have a billet that fits the sailor's rank. If the sailor's rank is higher or lower than the required rank, or if no sailor is available for a billet, there is a mismatch in the system. This mismatch creates an opportunity cost, which could be defined as a difference of the pay grade of the sailor and the pay grade that is related to the billet's required rank.

To capture changes in these costs while offering sailors NMIs, a simulation model is designed. This simulation model is explained in the next chapter.

THIS PAGE INTENTIONALLY LEFT BLANK

III. SIMULATION MODEL

A. INTRODUCTION

The goal of this research is to simulate the assignment process. This is done by describing a community of sailors over time. The process is presented in Figure 1.

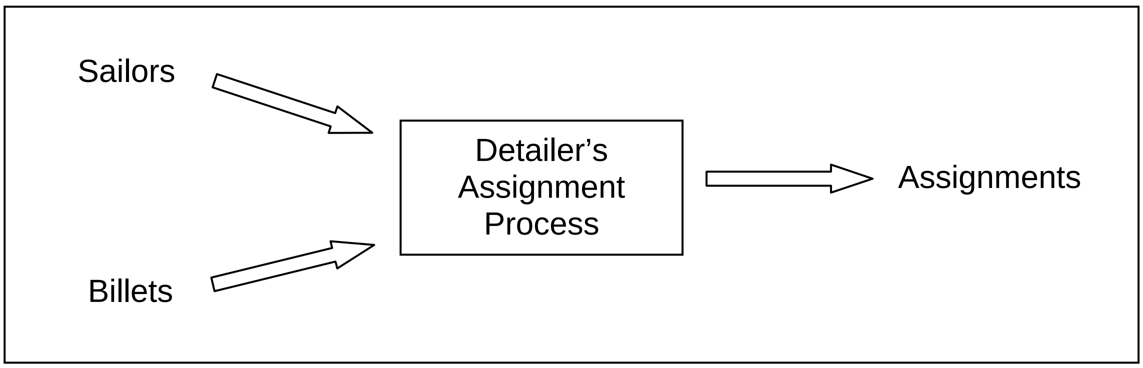


Figure 1. Overview on the assignment process

This assignment process can be seen in three parts. First, on the input side, there is the community of sailors and a set of billets. The second part is the assignment process and the third part is the output. The output of the detailer's work is the sailors' orders, the assignments. This assignments lead to moving sailors, which produces costs. The simulation model determines changes in costs while changing constraints in the assignment process.

In the next sections of this chapter, present the inputs and the assignment process. The last section describes the simulation model.

B. INPUTS

The detailer input will be the group of sailors who need new billets. To get this information requires a sailor community as well as a set of billets. The assignment process is a monthly recurring action. Therefore, the simulation sailor community has to behave like a real community with monthly changes caused by promotion, attrition and aging. The simulation sailor community and the organization of billets are described in the next paragraphs.

1. The Sailor Community

To assign an individual sailor in the assignment process to different billets, requires simulating a community of sailors with individual characteristics. This section explains the individual characteristics as well as the sailor community's behavior that are used in the simulation.

a) Sailor Community Structure

The Aviation Support (AS) community was chosen as the Navy community to model sailors. The size and structure of this community was used to generate the simulated sailor community.(Bureau of Navy Personnel, 2009) The AS community includes different Navy Enlisted Codes (NEC). Each NEC requires different skills. The sailor's skills have to match the billet requirements. Therefore, it is very unlikely that a detailer would take a sailor of one NEC to assign him to a billet that requires another NEC. For this reason, the simulation concentrates only on a part of the whole AS community, which reduces the size of the simulation community. Another limitation is a focus on the ranks E4 to E6. With these two constraints, the simulated sailor community by rank, E4-E6, is depicted in Figure 2.

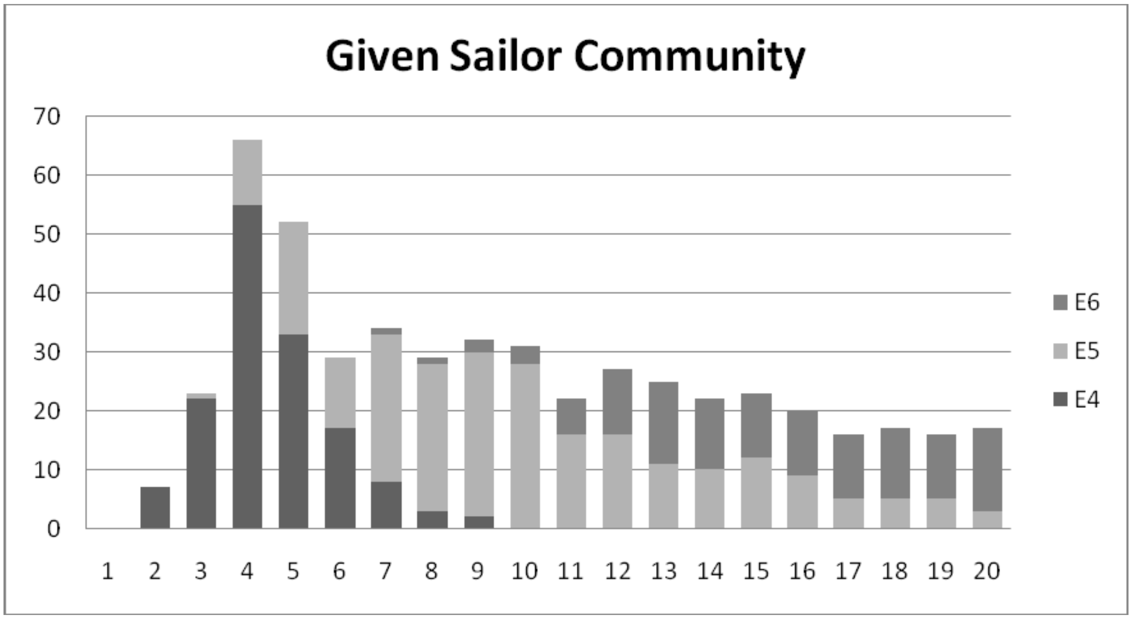


Figure 2. Distribution of E4 to E6 over Years of Service

Based on this sailor community structure each sailor in the community has his or her own characteristics.

b) Individual Sailor Characteristics

The individual sailor characteristics that are used in the simulation model are Time in Service (TIS), pay grade, Time in Rate (TIR), point of leaving the simulation (loss date), the next promotion date and individual NMI preferences.

TIS, pay grade and TIR are related to the community structure shown in Figure 2. The loss date, the promotion date and individual preferences are simulated using a random number generator. To generate the loss date, the U.S. Navy was used as a model. Because of a high attrition rate within the first 10 years of service, and nearly no attrition between 15 and 20 years of service, the 1994 cohort data set was taken to capture the attrition rate of the first years of service. The generic empirical attrition distribution of the U.S. Navy 1994 enlisted cohort is used in a Monte Carlo simulation to generate the loss date as an individual sailor characteristic in the simulation model (See Figure 3.).

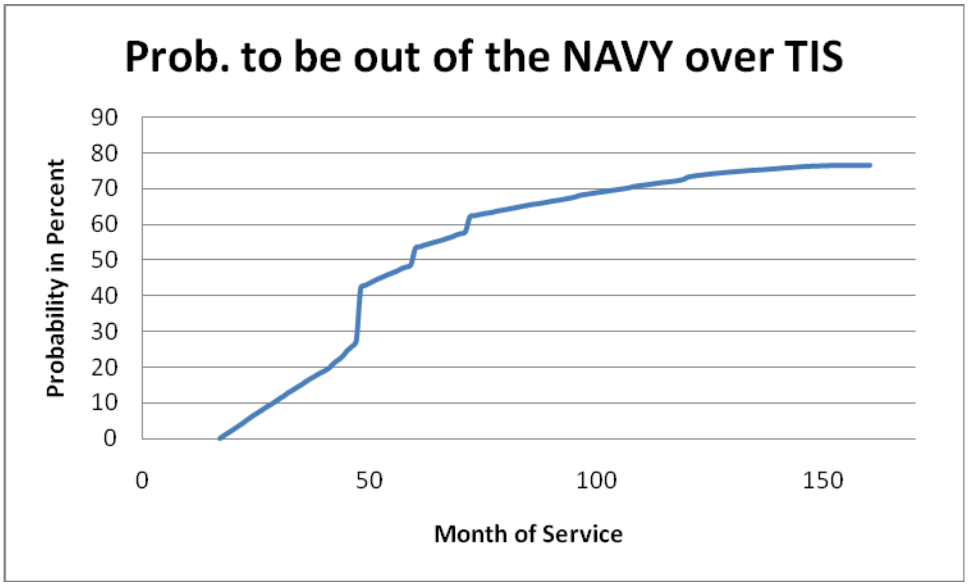


Figure 3. Attrition distribution for Sailor finished 16 Months of Service

The process of promotion is simplified for the simulation. Basic eligibilities like the required time in service (TIS) and time in rate (TIR) (Table 3.) are included in the simulation, while other factors that influence promotion in actuality, such as exam scores and performance points, are excluded. To capture all these individual qualities in the simulation, a random time is added to the earliest possible time for promotion. This random time is normally distributed and determined by the Monte Carlo simulation. In the simulation model, a promotion will be executed after the TIS and TIR requirements are fulfilled and a random time has passed.

Promotion to	Time in Service	Time in Rate
E4	2 years	6 months
E5	3 years	12 months
E6	7 years	36 months
E7	11 years	36 months

Table 3. Time Requirements for Promotions (Military.com, 2009)

The delay time distribution for a promotion to E6 is shown in Figure 4. This delay time simulates the individual sailor's performance as well as the

Navy's financial capability to promote sailors. It also incorporates a possibility to not be promoted. The simulation possibility for promotion is chosen with 95% confidence.

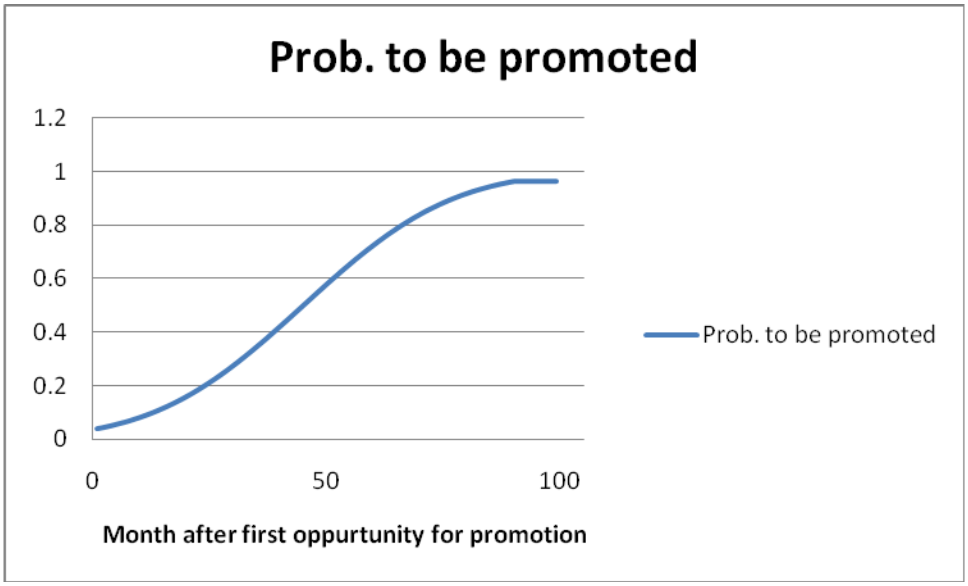


Figure 4. Probability to be promoted over time

c) Sailor Community Behavior

An initial simulation was run to validate the behavior of the sailor community and ensure a steady state of sailors after some period of time. This verifies the proper sailor population dynamics. This initial simulation includes growth and aging. To simulate growth, every sailor leaving the simulation is replaced by a new E4 with 16 months of TIS. Sailors leave the simulation community when they reach the attrition date or a High Year Tenure (HYT) limit (Table 4.).

Rank	HYT
E4	10 years
E5	20 years
E6	20 years

Table 4. HYT limits by rank (From Navy.mil, 2002)

After aging the sailor community, the shape and structure should be similar to the structure of the given community. Several simulation runs over 30 years have given the following average sailor community structure:

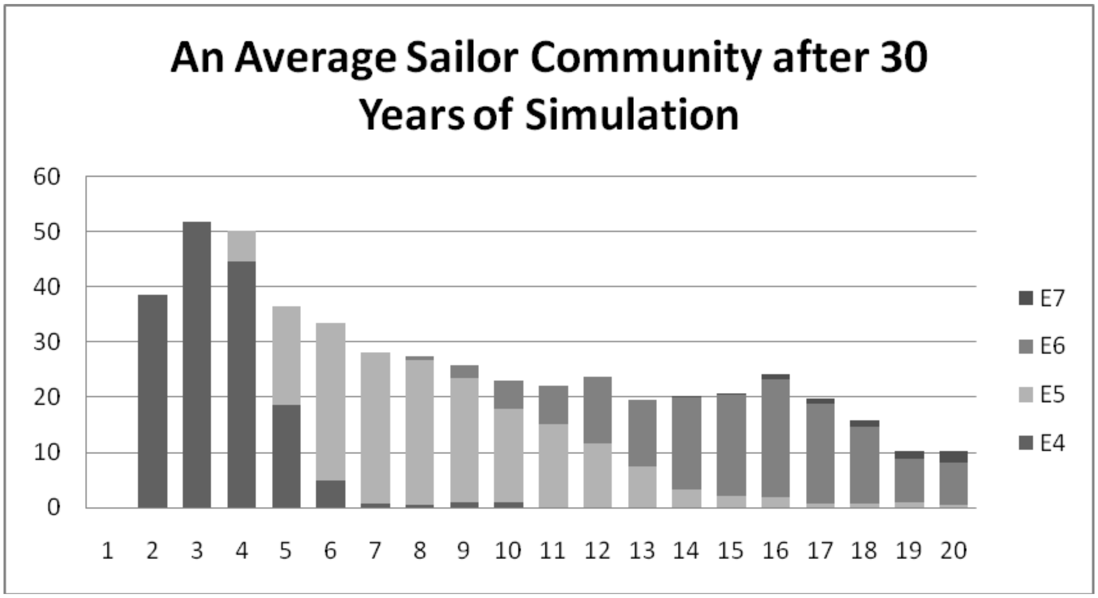


Figure 5. Diagram of an average sailor community structure after 30 years of simulation

A comparison with the actual AS sailor community shows that the simulation meets the goal to simulate growing and aging of a sailor community. The structure and rank distribution fits a real sailor community, with a smooth transition from rank to rank.

A plot of the average number of new sailors each month of simulation has shown that the simulation has a warm up period of 235 months before reaching a nearly steady state. Using this warm up time for the simulation guarantees an unbiased sailor community for different simulation runs. The unbiased simulation sailor community after a warm up period refelcts a complete change of all sailors in the simulation model after 224 months of simulation and randomly chosen individual sailor characteristics for every new sailor.

2. The Billets

The simulated billets have to fulfill some basic requirements to reach the research goal. The billets have to be distributed over different regions to make sailors move to get there. Each billet has to require a particular rank of a sailor. Further, the billet has to be on a type of platform to give the possibility to offer platform of choice as an NMI. With all these requirements, a set of billets was created that fits the given sailor community (Figure 2.). The used regions are:

Region	Description
CEC	This billet is located in the Central East Coast
CGC	This billet is located in the Central Gulf Coast
CNW	This billet is located in the Central North West coast
CSW	This billet is located in the Central South West coast

Table 5. Regions used for billets

This region setup is identical to the one used in Stitt's optimization model. (Stitt, December 2009). This simulation model does not differentiate between region and homeport. This level of detail is hidden for simplification reasons. The distances used to calculate moving costs between these regions are the distances between the main Navy ports within the region. These are Norfolk, VA; Mayport, FL; Everett, WA; and San Diego, CA. The platform types that used in the simulation model are:

Platform type	Description
CVN	Carrier Vessel, Nuclear (sea)
LCS	Littoral Combat ship (sea)
LHD	Amphibious assault ship (sea)
LPD	Amphibious transport docks (sea)
NAS	Naval Aviation Station (shore)
Other	Other type of shore billet (school…)

Table 6. Type of Platforms

With both the regions and the platform types, a simulated Navy was designed for the simulated sailor community. This simulated Navy should not be a copy of a part of the U.S. Navy. The goal is to simulate an assignment process that faces similar constraints to that of the U.S. Navy's detailer assignment process. The billet structure is presented in the next figure.

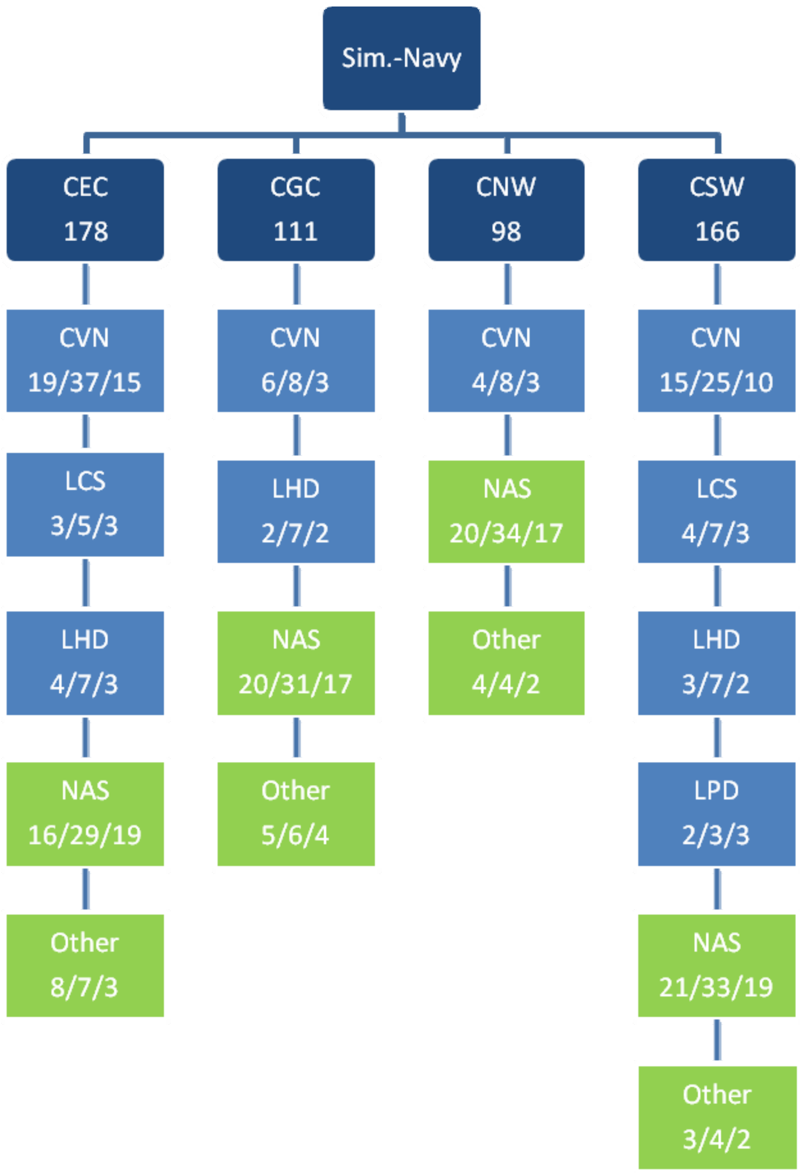


Figure 6. Diagram of the organization of billets used in the simulation

The numbers under the regions are the number of billets within the region. The numbers under a type of platform are the numbers of billets that require an E4, E5 or E6, respectively. Both the simulated sailor community and the simulated Navy are the inputs for the assignment process described in the next section.

C. ASSIGNMENT PROCESS

The assignment process is a monthly recurring process in the Navy as well as in the simulation. First, the detailer has to identify which sailors require reassignment. There are different reasons to assign a sailor to a new billet. The main reason to assign sailors to a new billet is the sea-shore cycle. After a set time period (called a tour) the sailor needs to be moved to another billet. The simulation model uses the tour times for the AS community. (see Table 7.)

Tour	Sea	Shore
1	36	48
2	36	48
3	36	48
4	36	36
5	36	36
6	36	36

Table 7. Sea-Shore-Flow in months, AS Community(From Bureau of Navy Personnel, 2009)

If a new assignment is due, the detailer starts nine months in advance to find a new billet for the sailor. Other reasons to assign a sailor to a new billet could be that a sailor finished his A-School and needs his first billet or a sailor was promoted and the rank does not match with the required rank of the billet to which he is assigned. When the list of sailors available to move is completed, the detailer can match these sailors to the available billets.

This matching process can be described as a set of rules the detailer follows. These rules can be:

- New sailors in the community who are not assigned to a billet should get a vacant billet before other sailors who already have an assignment.
- Sailors who became available to move earlier should be assigned before sailors who became available later (first-come, first-serve).
- The sea-shore indicator has to match with the type of billet to which the sailor will be assigned.
- A sailor only can be assigned to a billet that is free at the time it is needed.
- The required rank of the billet should fit with the rank of the sailor to avoid a mismatch cost. Law allows a mismatch of one pay grade step up or down.
- Finding a new billet within the region where the sailor is located avoids moving costs for the Navy.

These are the rules the simulation model uses to find an assignment. If an NMI is awarded to a sailor, additional match criteria, such as a platform choice, have to be fulfilled. If a match is found, the detailer can write the sailor's orders.

D. USE OF THE SIMULATION MODEL

As soon as a sailor gets his new orders, he can move from one duty station to another. This sailor movement causes moving costs, which are calculated and recorded in the simulation model. Additionally, depending on the fit between the billet and assigned sailor, there can be a mismatch cost. To identify differences in costs, the assignment process simulation presented in Figure 1. has to run under different scenarios. These scenarios are the inputs to the simulation model. Costs are part of the simulation model output. Figure 7. outlines the assignment process in the simulation model. The following two paragraphs explain the simulation model's inputs and outputs.

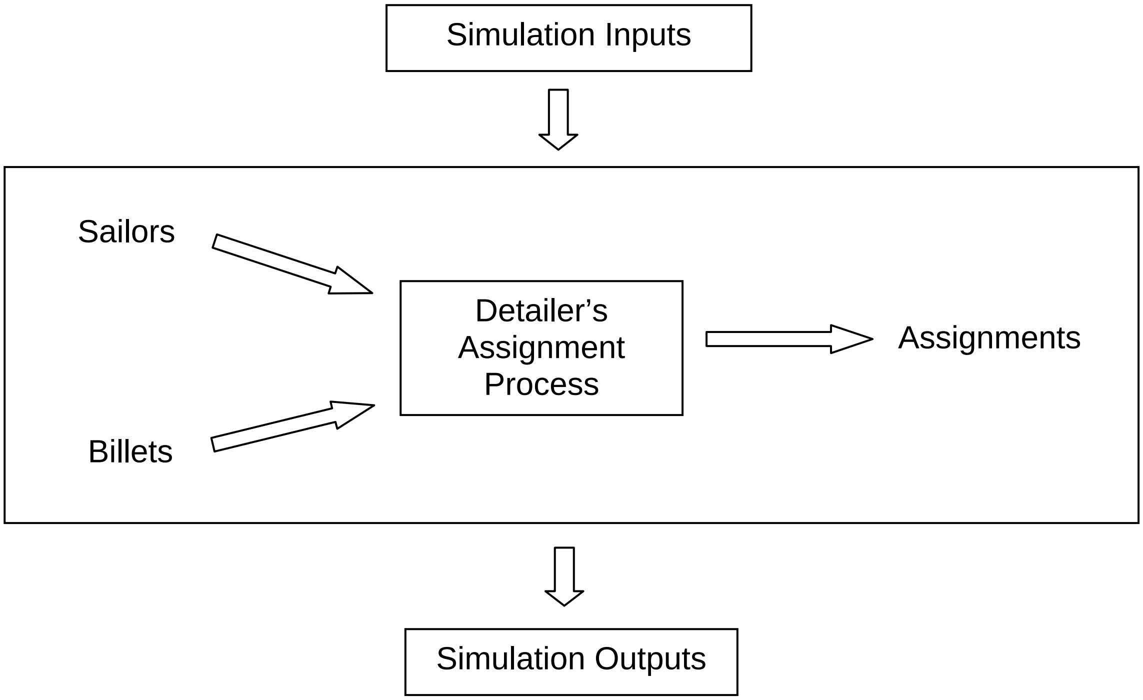


Figure 7. The assignment process in the simulation model.

1. Inputs to the Simulation Model

The simulation model counts costs per assignment and other measured values over a specified period. To run a scenario with the simulation model, scenario inputs are required. These inputs include:

- Simulation time and beginning of documentation: The simulation model identifies cost over time. The length of time these costs will recorded is given by the number of months for the simulation (end of experiment) and the starting point for documentation. Because of the simulation model's warm-up period, the starting point for documentation should be greater than 240 months.
- Number of scenarios: The simulation model allows the user to run several scenarios with an identical set-up. This number of scenarios that should be simulated is an input too.

- Type of NMI offered: The type of NMI that a sailor can chose after his first service obligation in a scenario is another input to the simulation model.

- Percentage of sailors awarded an NMI: To test the influence of increasing NMIs given to the sailors, the percentage of NMIs awarded has to be specified in the simulation model.

2. Outputs from the Simulation Model

The simulation model is designed to identify the costs of NMIs that are related to the assignment process and the constraints placed when offering these NMIs. To answer the research question in this thesis, the assignment costs relevant to the simulation results have to be calculated. To calculate the cost of assignments, this thesis uses Stitt's cost elements (Stitt, December 2009). In addition to the total costs, the simulation model documents several different output values. These values include:

- Average PCS Cost: An average moving cost per assignment.

- Number of assignments: The number of all assignments between the beginning and end of the documentation period in an experiment.

- Average waiting time: The average delay time per assignment. The delay time is months it takes the detailer to assign a sailor after the sailor is scheduled to fill a new billet.

- Number of NMIs awarded: The number of NMIs awarded within the sailor community in an experiment.

- Number of assignments in the presence of NMIs: Number of assignments satisfying the additional matching criteria imposed by the NMIs offered.

- Average waiting time of NMI assignments: Average delay time for assignments constrained by the additional NMI matching criteria.
- Mismatch Costs: Costs that measure the misfit between billets and the assigned sailors.

The costs of the different types of NMIs explained in Chapter II are presented in the next chapter. Each NMI was simulated at different levels of NMI use.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. RESULTS

A. INTRODUCTION

The simulation model described in Chapter III was used to run several scenarios. The simulation model set up and methods for calculating results are presented in section B. Results are presented in sections C. and D.

B. EXPERIMENTAL OVERVIEW

1. Set Up

As shown in Chapter III, the simulation model needs a warm up period of 240 months. After simulating 240 months, the data can be collected. The length of simulation period is related to the variance of the results. The average cost the system identifies becomes more constant over time. A limitation in the simulation duration is the computer time required to run a simulation. This time varies by the type of NMI offered and the percentage of NMI use.

All experiments used in this thesis have a simulation time of 360 months. The starting point to collecting data was fixed at 240 months. This leads to a 120 month (10 year) period for data collection. A set of scenarios in this thesis include 25 simulation runs.

2. Output Measures

The output measures are the outputs of the simulation model as described in Chapter III.4.B.. The units of measurement are presented in Table 8.

Output measure	Unit of Measurement
Average PCS Cost	Average $ per Assignment
Waiting Time	Month
NMI Assignment Waiting Time	Month

Output measure	Unit of Measurement
Average Mismatch Cost	Average $ per Billet and Month
Number of Assignments	Absolute Number of Assignments in Period of Interest
Number of NMIs	Absolute Number of NMI given to Sailors in Period of Interest

Table 8. Units of Measurement of Output measures

3. Scenario Description

The simulation model explored different scenarios. The first simulation runs evaluated the baseline scenario. Then, the different NMIs (see Section II, Paragraph C.1 of Chapter II.) were added as constraints to the simulation model. Each of these NMIs was tested at different levels of NMI use. The levels of NMI use were 5%, 10%, 20%, 30%, 40% and 50%. Section C of this chapter presents the baseline results. The different scenario results are presented in Section D.

4. Statistical Tests

A multiple comparison method was used to test if there are statistically significant changes in means between different scenarios and between different levels of NMI use. The chosen test is the Tukey's test. Tukey's test is very similar to the t-test. The difference is that the Tukey's test corrects for the probability of increasing type I errors if multiple comparisons are made. The test statistic used is given by the following equation

$$q = \frac{\bar{x}_{max} - \bar{x}_{min}}{s/\sqrt{n}}$$

Tukey's test "determines a critical number such that, if any pair of sample means has a difference greater than this number, we conclude that the pair's two corresponding population means are different."(Keller, 2005) The critical number is described by:

$$\omega = q_{\alpha}(k, v)\sqrt{\frac{MSE}{n_g}}$$

Where n is the number of observation, v is the degrees of freedom associated with MSE, n_g is the number of observations in each of k samples, α is level of significance and q_{α} is the critical value for the given degrees of freedom.

All statistical test results are presented in Appendix A.

C. BASELINE SCENARIO TESTING

To form a system performance baseline, the first simulation runs excluded all NMIs. The results are presented in Table 9.

Values	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
Average Number of Assignments per Simulation	2229.08	36.92	7.30	2214.0	2244.2
Average PCS Cost per Assignment	2367.18	63.26	12.65	2341.1	2393.3
Average Waiting Time per Assignment	4.89	0.23	0.05	4.79	4.98
Average Mismatch Cost per Billet and Month	388.27	8.10	1.62	384.93	391.61

Table 9. Results of baseline testing

The number of assignments was as expected and matches with the number of assignments a detailer makes in a ten years period for this community size, according to Richard Schlegel.(Schlegel, 2000) Because of the simplification in the simulated Navy, there are no validation data for the PCS Cost and Mismatch Cost. Both depend on the simulation model's sailor-billet fit.

Further, the simulation model has an average waiting time of 4.89 months for a sailor assignment. The reason for this relatively high value is related to the sailor - billet fit in the simulation model. New sailors in the simulation model have to fill a sea billet as their first assignment. Because there are fewer sea billets than sailors entering the simulation, most of the waiting time reflects new sailors waiting in the queue. Assigning sailors to billets that are in use, as happens in the U.S. Navy, is precluded in the simulation model. Therefore, the waiting time is higher and not comparable to real data, however, it will be relevant for comparison with other scenarios. The waiting time is a measure of the sailor-billet fit and shows how constraints will affect the assignment process itself. Changes in waiting time from the baseline scenario will reflect longer wait times in reality. Increasing waiting time can also decrease the sailor's motivation.

Because of the restriction, that the number of sailors is always smaller than the number of billets, any waiting time for a new sailor implies an empty billet elsewhere in the system. Empty billets, as well as billets filled by sailors with mismatching ranks, create a mismatch cost (see Section C, Paragraph 2.b. in Chapter II.)

This mismatch cost is an average mismatch cost per billet per month. The total mismatch cost equals the average mismatch cost times the months of simulation times the number of billets (there are 553 billets in this simulation model).

D. RESULTS OF SCENARIO TESTING

In further scenarios, the simulation model measured changes in behavior and costs after offering different NMIs.

1. Overview

The following paragraphs present the results from offering a single or combination of NMIs where the simulation model identifies statistically significant changes in mean values for the output measures of interests. The NMIs include Homeport of Choice (HP), Platform of Choice (PF), Regional Stability (RS) and combinations of these, as introduced in Table 2. The next paragraph concentrates on presenting total PCS Cost and PCS Cost per NMI. The following sections presents the Mismatch Cost.

2. PCS Cost

a) Total PCS Costs

Offering Homeport of Choice has a significant impact on the PCS costs. The mean comparison test shows that the average PCS Costs change significantly if more than 10% of the sailors choose this option. This leads to an increase in the total PCS cost. Total PCS costs are calculated by the following equation:

$$Total\ PCS\ Cost = Average\ PCS\ Cost * Average\ Number\ of\ Assignments$$

The Total PCS Cost for Homeport of Choice are presented in Figure 8. These results reflect the detailers' assignment priorities. Detailers are modeled as economizing on moving costs. If there is a suitable assignment in the same or a neighboring region, that assignment receives priority. With Homeport of Choice, detailers have be constrained to moving sailors to a more distant location when a closer suitable assignment is available; it might also require moving another sailor who could have been assigned locally, because his possible billet s given to a sailor who has chosen the NMI Homeport of Choice.

The Regional Stability NMI guarantee a sailor will stay in the assigned region for two tours. Not moving sailors should reduce the PCS Cost, though it could have the secondary effect of increasing PCS costs for other sailors. The results indeed show a decrease in PCS Cost if the detailer is constrained to retain sailors within a region for two tours. If 20 percent or more sailors use this NMI, there is a statistically significant decrease in average PCS cost. (See Figure 8.)

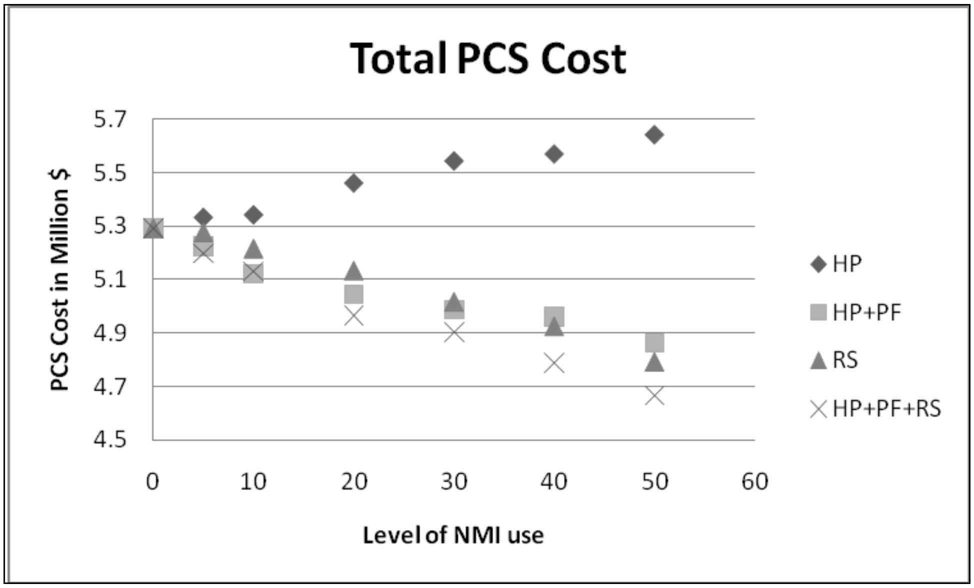


Figure 8. Total PCS Cost in millions $ for a 10 year period

As Figure 8. and Figure 9. show, offering Regional Stability can save PCS Costs. These cost savings are possible, because the detailer's rules to find a match have a higher priority for matching ranks match than to economize on PCS costs. This rule was implemented into the simulation model as a constraint to minimize mismatch costs. When sailors chose the Regional Stability NMI, the detailer has to prioritize the regional match more often.

Offering Platform of Choice leads to no statistically significant changes in the means of average PCS Cost. This is the reason, why the total PCS Cost values for this NMI are omitted from Figure 8.

If Platform of Choice and Homeport of Choice are offered simultaneously, the sailor will chose a specific platform in the preferred region. The PCS costs in the simulation model with these combined constraints do not match the results when Homeport of Choice is the only NMI offered. We see an increase in average PCS cost while offering Homeport of choice, there is a decrease in PCS cost when we offer Homeport of Choice in conjunction with Platform of Choice.

This savings result from changing the detailing priorities. These savings are possible, because of the given misfit between the sailor community and the set of billets. Normally sailors are assigned to a sea billet for their first tour and a shore billet for the second tour. Because of the limited number of shore billets, many sailors have to move for their second tour. Some sailors will choose a sea billet for their second tour if given the choice of platform. This allows the detailer to assign more sailors within the region they are stationed. Assignments within a region reduce total PCS Costs.

Offering all NMIs at the same time means the sailor can decide where he wants to serve, on what platform and will remain in this location for the next two tours. The reduction in PCS Costs from staying in the same region for two tours occurs here again. The total PCS costs decrease because fewer sailors have to move. The total PCS cost for the community of sailors for the 10 year simulation decreases as the NMI use increases. The results are depicted in Figure 8.

b) PCS Cost per NMI

The change of total PCS Costs for increasing NMI use can be related to the NMIs offered in these experiments, because the NMIs offered is the only change. The average cost per NMI is calculated by relating the change in total PCS Cost to the number of NMIs offered, as calculated by the equation:

$$NMIcost_{(\%\,of\,use)} = \frac{Change\ in\ total\ PCS\ Cost_{(\%\,of\,use)}}{Number\ of\ NMI\ offered_{(\%\,of\,use)}}$$

Based on this equation the average PCS Cost per NMI were calculated and are presented in Figure 9.

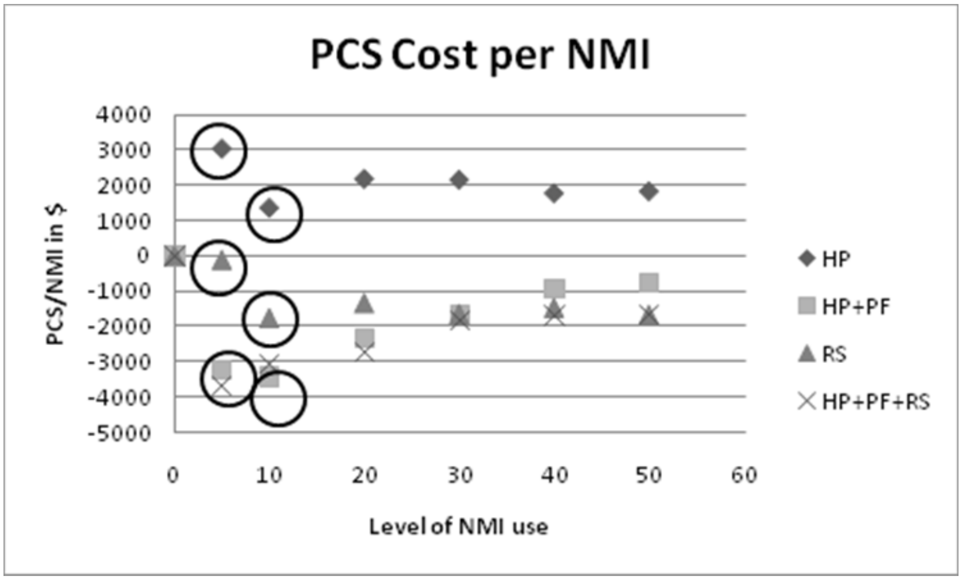


Figure 9. PCS Cost per NMI at different level of NMI use (Circled data points are based on means that are not statistically significantly different then the base line means)

The PCS Cost per NMI for the combination of Homeport, Platform and Regional Stability in Figure 9. levels out after 30% of NMI use. This depends on the decrease of assignments per month depending on an increasing waiting time and an independent linear increase of waiting time for NMIs that are used.

3. Mismatch Cost

a) Waiting Time

The simulation model identifies statistically significantly fewer assignments for higher levels of NMI use for the NMI Platform of Choice and NMI combinations including Platform of Choice. The decrease in the number of assignments reflects an increasing waiting time, the time a sailor has to wait after he should have been assigned a new billet. Offering Platform of Choice appears to make it harder for the detailer to find assignments for all sailors. The average waiting time is presented in Figure 10.

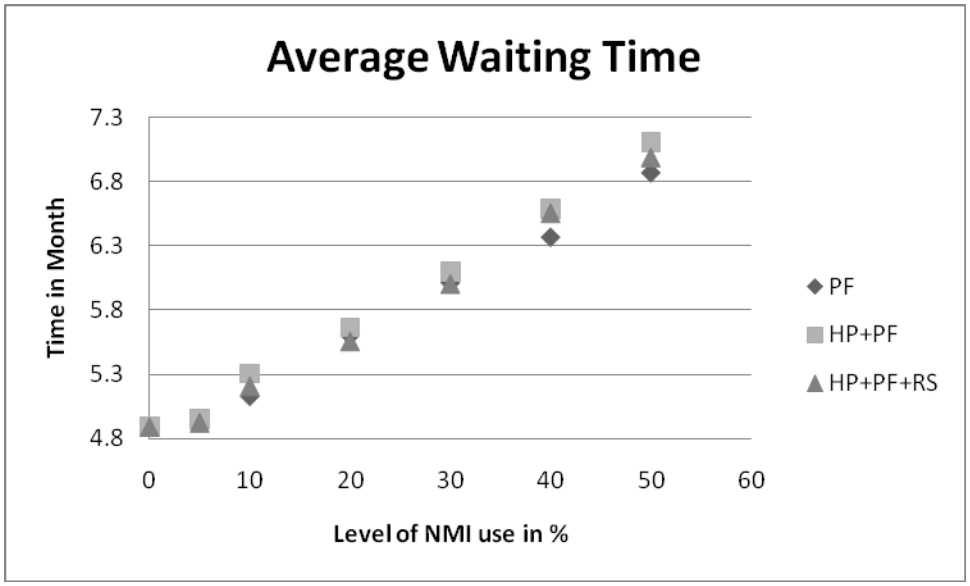


Figure 10. Average Waiting time at different levels of NMI use

This increase in waiting time also leads to an increase in mismatch cost.

b) Total Mismatch Cost

The Total Mismatch Cost are calculate by the following equation

$$\textit{Total Mismatch Cost} = \textit{Average Mismatch Cost per billet} * \textit{Number of Billets} * \textit{month of simulation}$$

As opposed to PCS Costs, Mismatch Costs are not real costs. The mismatch cost reflects the sailor-billet-fit. The mismatch cost occurs when sailors and billets have mismatching ranks, either filling a billet with a presumably overqualified sailor, which wastes some of the sailor's skills, or filling a billet with an underqualified sailor, which presumably compromises job performance. These costs are measured by the pay difference between the billet's required pay graqde and the salior's actual pay grade. The sailor community structure does not depend on the assignment fit or the waiting time (e.g., increasing mismatches and waiting time do not decrease retention). However, a longer waiting times and more frequent sailor-billet-misfits, will likely decrease motivation and job satisfaction. The Total Mismatch Costs are presented in Figure 11.

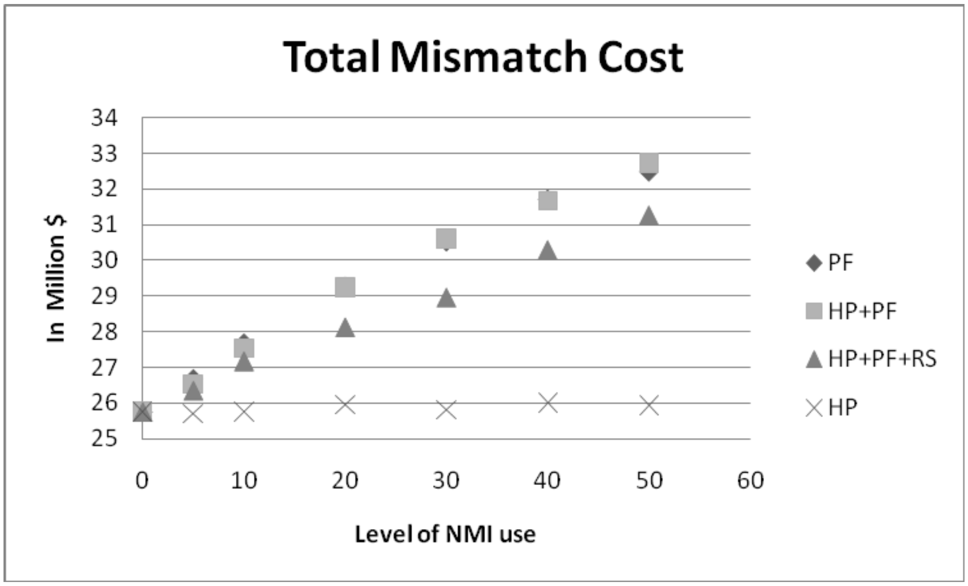


Figure 11. Total Mismatch Cost in a ten year period

Assignment quality (sailor-billet-fit) is almost the same when offering Platform of Choice and the combination of Platform and Homeport of Choice. All of the means of Mismatch Costs are statistically significant compared to the baseline value.

Offering the NMI Homeport of Choice or the NMI Regional Stability leads to no statistically significant changes in average mismatch cost values.

c) Mismatch Cost Per NMI

The increase in the average mismatch costs per billet depends on the NMIs used. Dividing the increase in total mismatch cost for a given NMI use by the number of NMIs awarded, provides the cost per NMI.

The Mismatch Cost per NMI over the different levels of NMI use for the NMIs that include Platform of Choice are presented below:

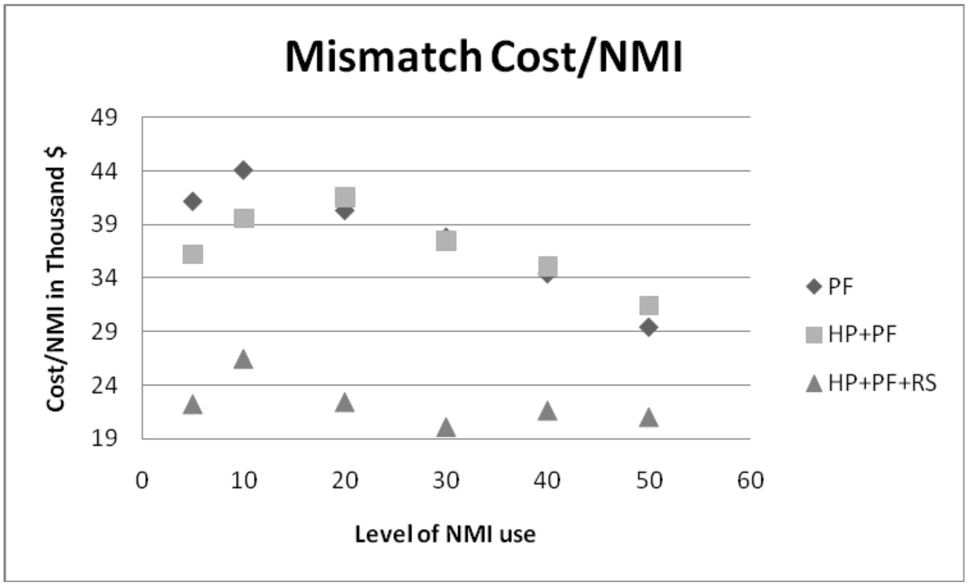


Figure 12. Mismatch Cost per NMI over levels of NMI use

Holding the sailors a second tour in the region on the same type of platform shows a lower Mismatch Cost per NMI, than turning them back after one tour into the regular sea-shore-rotation.

E. CONCLUDING REMARKS

The simulation model appears to successfully measure changes in different outcome indicators in the complex assignment process. The simulation model behaves in an expected way it confronts increasing constraints (here NMIs) that influence the assignment process. An example of expected behavior is the decreasing average in mean of PCS Cost if more sailors have to stay for a longer time in the same location.

The final chapter will summarize the results, discuss limitations of the results and provide recommendations.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS

Offering sailors NMIs that affect the assignment process will have varied effects on the Navy's complex organization. It is almost impossible to identify one price that could be cited for an NMI. The results give an impression about the complexity of the considerations facing a detailer. This thesis reports results from a simulation model that counted steady-state costs over 10 years for a stylized Navy community. Very often, the changes in measurable values for small levels of NMIs use were not statistically significant. The simulation model identified the cost of different NMIs by adding them one by one as input constraints to the simulation model. Table 10. summarizes the results of Chapter IV.

It should be noted that the costs per NMI reported here are average costs for that NMI or combination of NMIs over the entire sample for the entire time period. The costs for any individual sailor will likely be higher or lower than this average cost. Thus, some sailors will over pay for the NMI and some will under pay. Furthermore, some may have values that exceed the specific cost to provide them the NMI but are less than the average cost. Using average cost in the CRAM model will preclude those sailors from receiving the NMI. The difficulty is that the cost to provide one sailor an NMI can't be separated from the overall system costs as the costs are interrelated. Therefore, the only alternative is to infer the average cost of the NMI on every sailor.

A. SUMMARY

The simulation model designed to identify the costs of NMI worked well. The chosen scenario definitions are a compromise between lower variances and higher statistical significances in changes of measurable values on the one hand, and much longer simulation times on the other hand. This compromise leads to a problem with small numbers in the lower levels of NMI use. The cost per NMI is calculated by using the number of NMIs awarded. For the 5% level of NMI use these are 22.5 NMIs with a standard deviation of 5.2 (Platform of Choice).

Because of this high standard deviation, the lower levels of NMI use are often not statistically significant in the changes in means. An overview on the costs and impacts side of the NMIs that were tested by the simulation model are presented in Table 10.

NMI Type	PCS Cost/ NMI (level of use)	Mismatch Cost/ NMI (level of use)	Impact on Motivation for Sailor Community
Homeport of Choice	~$2000 (>=10%)	0	0
Platform of Choice	0	$ 45k (10%) down to $ 30k (50%)	negative
Homeport and Platform of Choice	-$2300 (20%)up to -$800 (50%)	$40k (10 %)down to $31k (50%)	negative
Regional Stability	~-$1600 (>=10%)	0	0
Regional Stability in a Region of Choice	~$800 (>=40%)	0	0
Stability in a Region and on a Platform of Choice	-$3100 (10%)up to -$1600 (30-50%)	~$22k (all)	negative

Table 10. Cost and Impacts of NMIs to the Navy and the sailor community

The experiments have shown values for the different NMI types. With these values in mind, the CRAM can be implemented with NMI costs predetermined. But as shown in the example above, it will be up to the detailer to classify the sailors' preferences before value them into a category of NMI type.

B. RECOMMENDATIONS

The simulation model concentrates on measurable values in the assignment process. This research gives no insight into the possible increase in motivation of sailors and resulting benefits of increased motivation. Here, further research has to be done.

Offering NMIs in an auction is like selling them. If these incentives are sold to sailors, there is a risk of losing the motivational benefits of NMIs. Even worse, this could lead to a competition between sailors, i.e. to get a popular job. This competition might have an impact on interpersonal relationships, which can lead to dissatisfaction in the sailor community (Kohn, 1993). On the other hand, it is also possible that giving sailors a greater choice in determining their individualized retention incentive packages will increase morale and intrinsic motivation.

Beside the calculated costs, the simulation model gives insights into the organizational behavior if constraints are added. The simulation model uses a different set of rules to find assignments for sailors. Even without offering NMIs, this model can be used to vary these rules. This might be helpful to provide detailers with policies to use their resources efficiently.

Offering the labor force NMIs is essential to be an attractive organization in the labor market. Recognizing regional preferences of the employees in a large organization, such as the Navy, and realizing them will increase the organization's attractiveness.

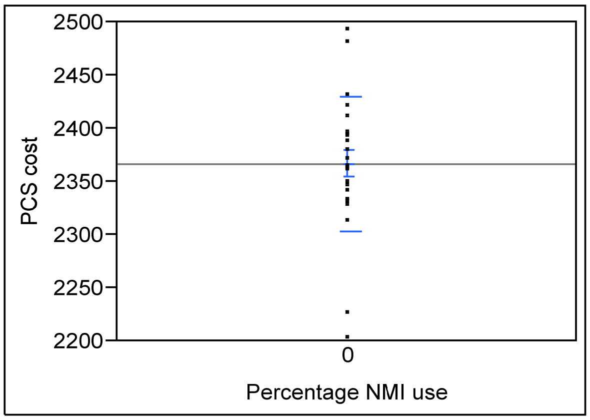
THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A

A. STATISTICAL RESULTS

1. Baseline Results

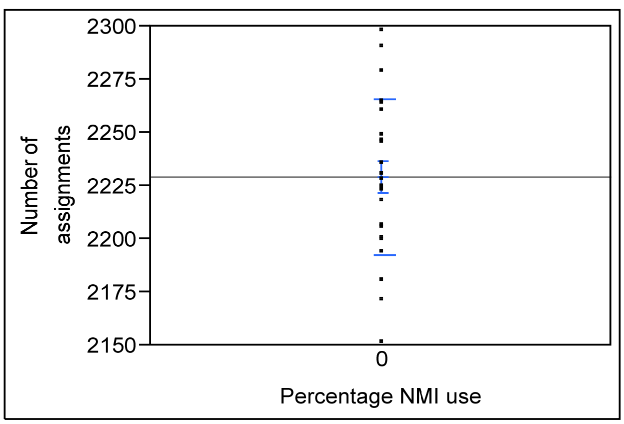
Oneway Analysis of PCS cost By Percentage NMI use NMI Type=0



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3

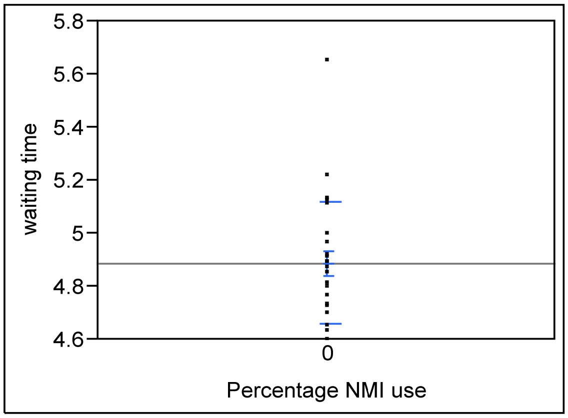
Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=0



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2

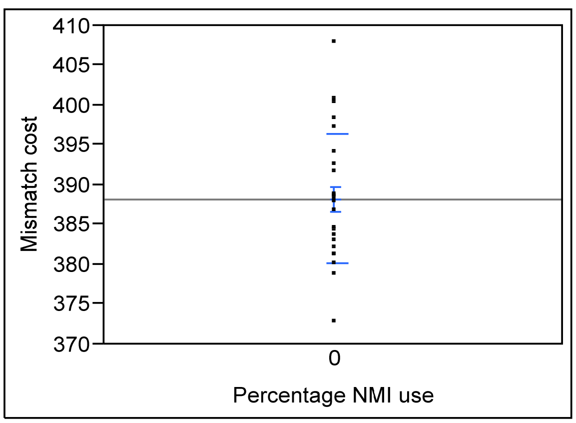
Oneway Analysis of waiting time By Percentage NMI use NMI Type=0



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=0



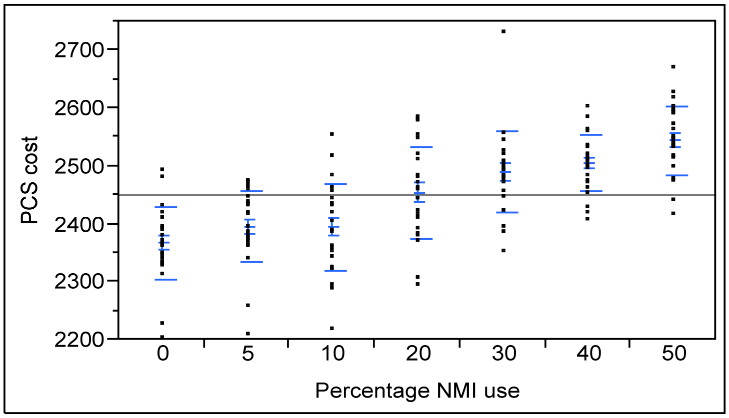
Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.09728	1.6195	384.93	391.61

2. Scenario Test Results

a) Statistical Results of Testing Type 1

Oneway Analysis of PCS cost By Percentage NMI use NMI Type=1



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3
5	25	2395.39	61.6685	12.334	2369.9	2420.8
10	25	2394.60	75.5024	15.100	2363.4	2425.8
20	25	2454.54	79.8423	15.968	2421.6	2487.5
30	25	2490.28	71.1343	14.227	2460.9	2519.6
40	25	2505.05	48.1324	9.626	2485.2	2524.9
50	25	2544.39	58.9551	11.791	2520.1	2568.7

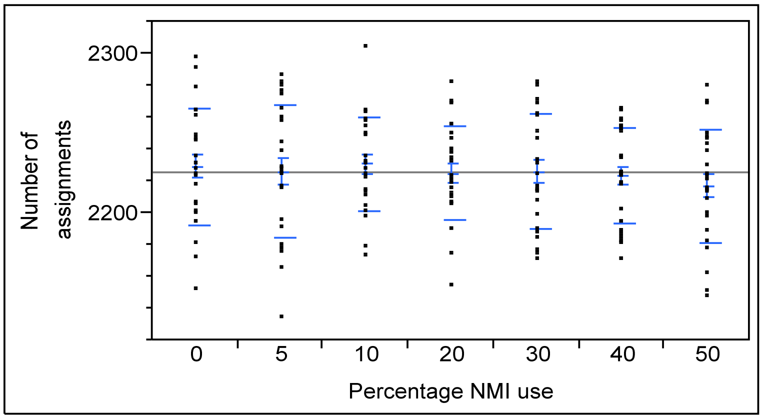
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	5	10	0
50	-55.9314	-16.5957	-1.82126	33.9185	93.07147	93.86153	121.278
40	-16.5957	-55.9314	-41.1569	-5.41716	53.73581	54.52587	81.94231
30	-1.82126	-41.1569	-55.9314	-20.1916	38.96133	39.75139	67.16783
20	33.9185	-5.41716	-20.1916	-55.9314	3.221571	4.011635	31.42807
5	93.07147	53.73581	38.96133	3.221571	-55.9314	-55.1413	-27.7249
10	93.86153	54.52587	39.75139	4.011635	-55.1413	-55.9314	-28.515
0	121.278	81.94231	67.16783	31.42807	-27.7249	-28.515	-55.9314

Positive values show pairs of means that are significantly different.

Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=1



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2
5	25	2226.08	41.4788	8.2958	2209.0	2243.2
10	25	2230.84	29.4218	5.8844	2218.7	2243.0
20	25	2224.80	29.3598	5.8720	2212.7	2236.9
30	25	2226.00	35.9108	7.1822	2211.2	2240.8
40	25	2223.20	29.8315	5.9663	2210.9	2235.5
50	25	2217.20	35.5622	7.1124	2202.5	2231.9

Means Comparisons

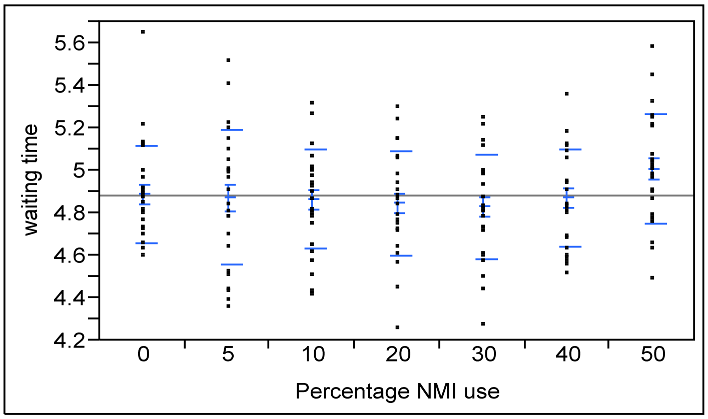
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	10	0	5	30	20	40	50
10	-28.9365	-27.1765	-24.1765	-24.0965	-22.8965	-21.2965	-15.2965
0	-27.1765	-28.9365	-25.9365	-25.8565	-24.6565	-23.0565	-17.0565
5	-24.1765	-25.9365	-28.9365	-28.8565	-27.6565	-26.0565	-20.0565
30	-24.0965	-25.8565	-28.8565	-28.9365	-27.7365	-26.1365	-20.1365
20	-22.8965	-24.6565	-27.6565	-27.7365	-28.9365	-27.3365	-21.3365
40	-21.2965	-23.0565	-26.0565	-26.1365	-27.3365	-28.9365	-22.9365
50	-15.2965	-17.0565	-20.0565	-20.1365	-21.3365	-22.9365	-28.9365

Positive values show pairs of means that are significantly different.

Oneway Analysis of waiting time By Percentage NMI use NMI Type=1



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836
5	25	4.87181	0.316349	0.06327	4.7412	5.0024
10	25	4.86452	0.235106	0.04702	4.7675	4.9616
20	25	4.84603	0.243897	0.04878	4.7454	4.9467
30	25	4.82950	0.243576	0.04872	4.7290	4.9300
40	25	4.87447	0.229397	0.04588	4.7798	4.9692
50	25	5.00809	0.257078	0.05142	4.9020	5.1142

Means Comparisons

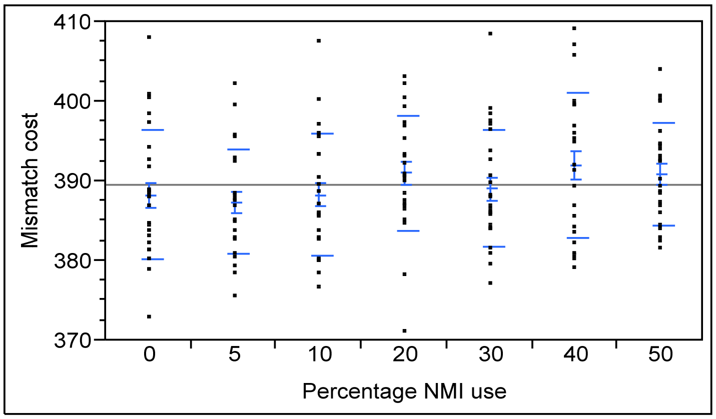
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	0	40	5	10	20	30
50	-0.21305	-0.0935	-0.07944	-0.07678	-0.06948	-0.051	-0.03447
0	-0.0935	-0.21305	-0.199	-0.19634	-0.18904	-0.17056	-0.15403
40	-0.07944	-0.199	-0.21305	-0.21039	-0.2031	-0.18462	-0.16809
5	-0.07678	-0.19634	-0.21039	-0.21305	-0.20576	-0.18728	-0.17074
10	-0.06948	-0.18904	-0.2031	-0.20576	-0.21305	-0.19457	-0.17804
20	-0.051	-0.17056	-0.18462	-0.18728	-0.19457	-0.21305	-0.19652
30	-0.03447	-0.15403	-0.16809	-0.17074	-0.17804	-0.19652	-0.21305

Positive values show pairs of means that are significantly different.

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=1



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.09728	1.6195	384.93	391.61
5	25	387.395	6.58264	1.3165	384.68	390.11
10	25	388.319	7.58366	1.5167	385.19	391.45
20	25	391.076	7.24736	1.4495	388.08	394.07
30	25	389.057	7.32629	1.4653	386.03	392.08
40	25	392.017	9.03597	1.8072	388.29	395.75
50	25	390.834	6.40531	1.2811	388.19	393.48

Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

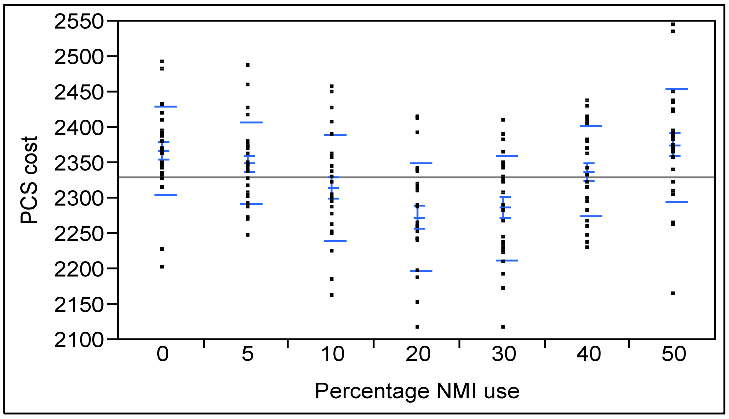
q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	40	20	50	30	10	0	5
40	-6.3432	-5.40222	-5.15993	-3.38354	-2.64513	-2.5953	-1.72121
20	-5.40222	-6.3432	-6.10091	-4.32451	-3.58611	-3.53628	-2.66218
50	-5.15993	-6.10091	-6.3432	-4.5668	-3.8284	-3.77857	-2.90447
30	-3.38354	-4.32451	-4.5668	-6.3432	-5.6048	-5.55496	-4.68087
10	-2.64513	-3.58611	-3.8284	-5.6048	-6.3432	-6.29337	-5.41927
0	-2.5953	-3.53628	-3.77857	-5.55496	-6.29337	-6.3432	-5.4691
5	-1.72121	-2.66218	-2.90447	-4.68087	-5.41927	-5.4691	-6.3432

Positive values show pairs of means that are significantly different.

b) Statistical Results of Testing Type 2

Oneway Analysis of PCS cost By Percentage NMI use NMI Type=2



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3
5	25	2348.94	57.5989	11.520	2325.2	2372.7
10	25	2315.32	74.9336	14.987	2284.4	2346.2
20	25	2273.71	75.8448	15.169	2242.4	2305.0
30	25	2286.60	73.9191	14.784	2256.1	2317.1
40	25	2338.47	62.8071	12.561	2312.5	2364.4
50	25	2375.28	80.6727	16.135	2342.0	2408.6

Means Comparisons

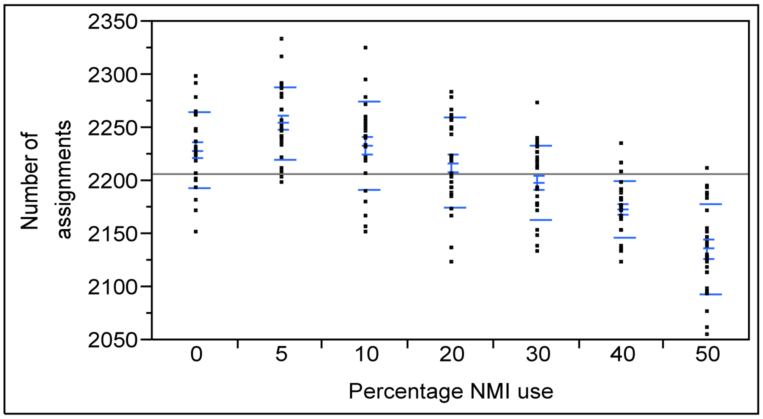
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	0	5	40	10	30	20
50	-59.3494	-51.2527	-33.011	-22.5402	0.611942	29.32358	42.2176
0	-51.2527	-59.3494	-41.1077	-30.6369	-7.48479	21.22685	34.12087
5	-33.011	-41.1077	-59.3494	-48.8786	-25.7265	2.985155	15.87917
40	-22.5402	-30.6369	-48.8786	-59.3494	-36.1973	-7.48567	5.408344
10	0.611942	-7.48479	-25.7265	-36.1973	-59.3494	-30.6378	-17.7438
30	29.32358	21.22685	2.985155	-7.48567	-30.6378	-59.3494	-46.4554
20	42.2176	34.12087	15.87917	5.408344	-17.7438	-46.4554	-59.3494

Positive values show pairs of means that are significantly different.

Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=2



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2
5	25	2254.44	34.0894	6.8179	2240.4	2268.5
10	25	2233.44	41.4699	8.2940	2216.3	2250.6
20	25	2217.28	42.2853	8.4571	2199.8	2234.7
30	25	2198.56	35.0168	7.0034	2184.1	2213.0
40	25	2173.08	26.1565	5.2313	2162.3	2183.9
50	25	2135.88	42.9741	8.5948	2118.1	2153.6

Means Comparisons

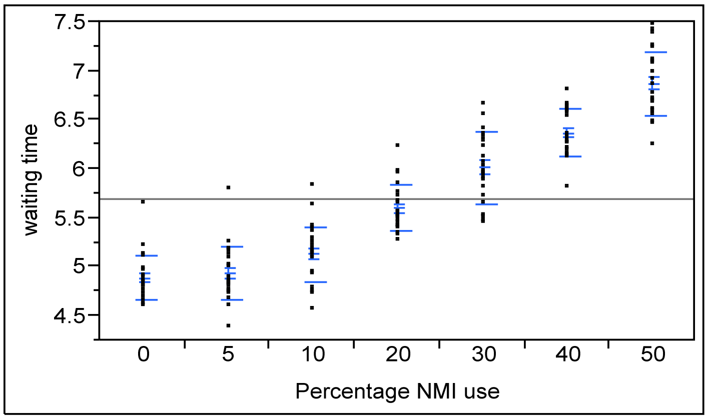
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	5	10	0	20	30	40	50
5	-31.5205	-10.5205	-6.16049	5.639509	24.35951	49.83951	87.03951
10	-10.5205	-31.5205	-27.1605	-15.3605	3.359509	28.83951	66.03951
0	-6.16049	-27.1605	-31.5205	-19.7205	-1.00049	24.47951	61.67951
20	5.639509	-15.3605	-19.7205	-31.5205	-12.8005	12.67951	49.87951
30	24.35951	3.359509	-1.00049	-12.8005	-31.5205	-6.04049	31.15951
40	49.83951	28.83951	24.47951	12.67951	-6.04049	-31.5205	5.679509
50	87.03951	66.03951	61.67951	49.87951	31.15951	5.679509	-31.5205

Positive values show pairs of means that are significantly different.

Oneway Analysis of waiting time By Percentage NMI use NMI Type=2



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836
5	25	4.94318	0.271888	0.05438	4.8310	5.0554
10	25	5.12869	0.284747	0.05695	5.0112	5.2462
20	25	5.60309	0.229224	0.04584	5.5085	5.6977
30	25	6.01391	0.366370	0.07327	5.8627	6.1651
40	25	6.36939	0.239341	0.04787	6.2706	6.4682
50	25	6.87134	0.329929	0.06599	6.7351	7.0075

Means Comparisons

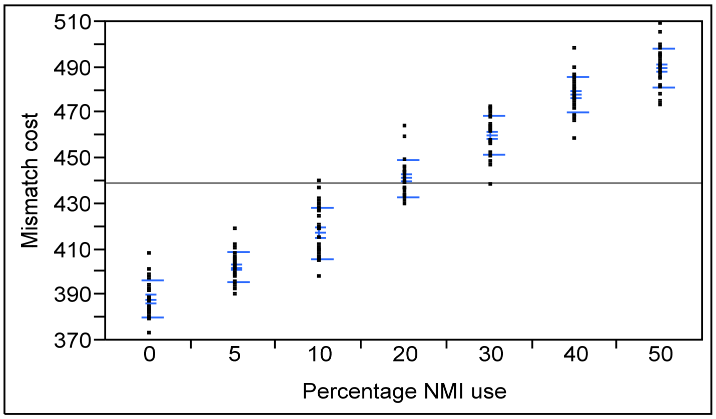
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	10	5	0
50	-0.23896	0.26299	0.618468	1.029283	1.503689	1.689197	1.743849
40	0.26299	-0.23896	0.11652	0.527335	1.001741	1.187249	1.241901
30	0.618468	0.11652	-0.23896	0.171858	0.646264	0.831771	0.886423
20	1.029283	0.527335	0.171858	-0.23896	0.235448	0.420955	0.475608
10	1.503689	1.001741	0.646264	0.235448	-0.23896	-0.05345	0.001202
5	1.689197	1.187249	0.831771	0.420955	-0.05345	-0.23896	-0.18431
0	1.743849	1.241901	0.886423	0.475608	0.001202	-0.18431	-0.23896

Positive values show pairs of means that are significantly different.

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=2



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.0973	1.6195	384.93	391.61
5	25	402.233	6.4073	1.2815	399.59	404.88
10	25	417.301	11.3638	2.2728	412.61	421.99
20	25	441.290	8.0479	1.6096	437.97	444.61
30	25	460.217	8.5449	1.7090	456.69	463.74
40	25	478.098	8.1437	1.6287	474.74	481.46
50	25	489.724	8.6301	1.7260	486.16	493.29

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

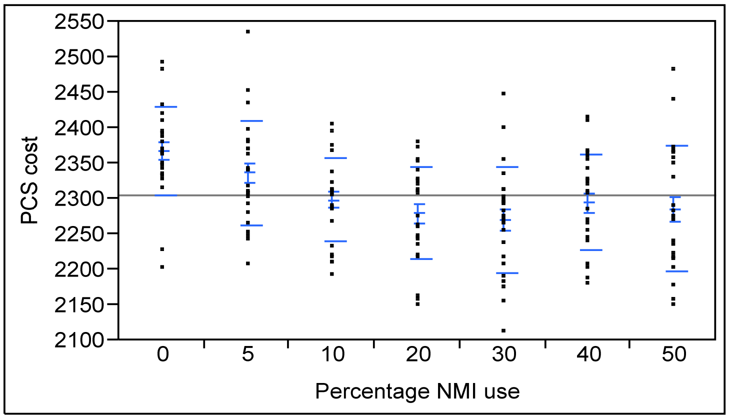
q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	10	5	0
50	-7.23564	4.390571	22.27167	41.19817	65.18804	80.25566	94.21965
40	4.390571	-7.23564	10.64547	29.57196	53.56183	68.62945	82.59345
30	22.27167	10.64547	-7.23564	11.69086	35.68073	50.74835	64.71235
20	41.19817	29.57196	11.69086	-7.23564	16.75423	31.82185	45.78585
10	65.18804	53.56183	35.68073	16.75423	-7.23564	7.831982	21.79598
5	80.25566	68.62945	50.74835	31.82185	7.831982	-7.23564	6.72836
0	94.21965	82.59345	64.71235	45.78585	21.79598	6.72836	-7.23564

Positive values show pairs of means that are significantly different.

c) Statistical Results of Testing Type 3

Oneway Analysis of PCS cost By Percentage NMI use NMI Type=3



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3
5	25	2336.33	73.3801	14.676	2306.0	2366.6
10	25	2298.16	59.0625	11.812	2273.8	2322.5
20	25	2279.22	65.4666	13.093	2252.2	2306.2
30	25	2269.75	75.5613	15.112	2238.6	2300.9
40	25	2294.12	67.2696	13.454	2266.4	2321.9
50	25	2285.49	88.5512	17.710	2248.9	2322.0

Means Comparisons

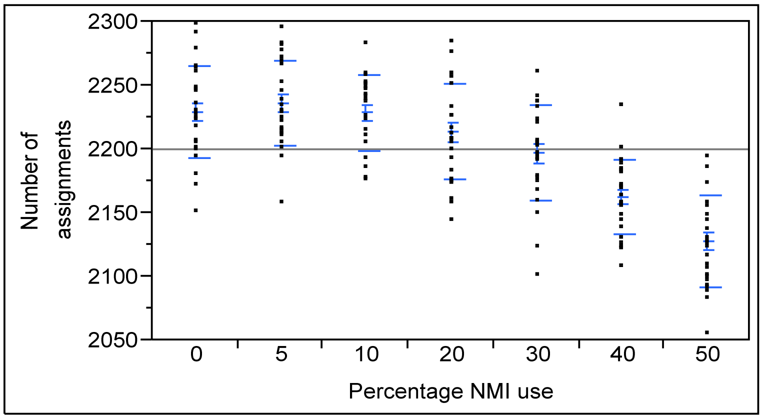
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	0	5	10	40	50	20	30
0	-59.8898	-29.0373	9.12551	13.16763	21.80104	28.06892	37.54515
5	-29.0373	-59.8898	-21.727	-17.6849	-9.05147	-2.78359	6.692643
10	9.12551	-21.727	-59.8898	-55.8477	-47.2142	-40.9464	-31.4701
40	13.16763	-17.6849	-55.8477	-59.8898	-51.2564	-44.9885	-35.5123
50	21.80104	-9.05147	-47.2142	-51.2564	-59.8898	-53.6219	-44.1457
20	28.06892	-2.78359	-40.9464	-44.9885	-53.6219	-59.8898	-50.4135
30	37.54515	6.692643	-31.4701	-35.5123	-44.1457	-50.4135	-59.8898

Positive values show pairs of means that are significantly different.

Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=3



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2
5	25	2236.44	33.2855	6.6571	2222.7	2250.2
10	25	2228.64	30.0720	6.0144	2216.2	2241.1
20	25	2213.60	37.8275	7.5655	2198.0	2229.2
30	25	2197.00	37.5433	7.5087	2181.5	2212.5
40	25	2162.88	29.3148	5.8630	2150.8	2175.0
50	25	2127.64	35.8932	7.1786	2112.8	2142.5

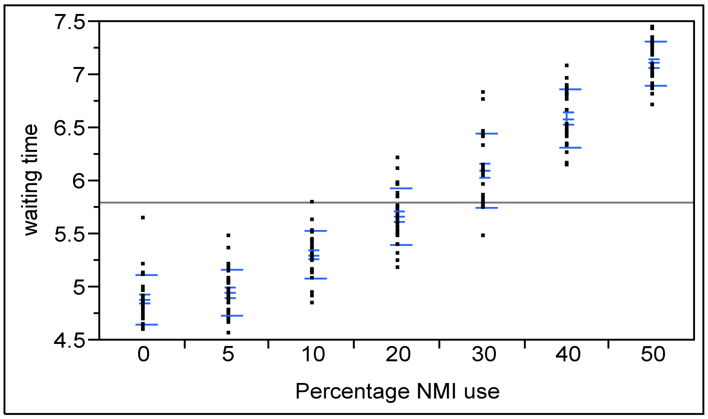
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	5	0	10	20	30	40	50
5	-29.126	-21.766	-21.326	-6.28596	10.31404	44.43404	79.67404
0	-21.766	-29.126	-28.686	-13.646	2.954036	37.07404	72.31404
10	-21.326	-28.686	-29.126	-14.086	2.514036	36.63404	71.87404
20	-6.28596	-13.646	-14.086	-29.126	-12.526	21.59404	56.83404
30	10.31404	2.954036	2.514036	-12.526	-29.126	4.994036	40.23404
40	44.43404	37.07404	36.63404	21.59404	4.994036	-29.126	6.114036
50	79.67404	72.31404	71.87404	56.83404	40.23404	6.114036	-29.126

Positive values show pairs of means that are significantly different.

Oneway Analysis of waiting time By Percentage NMI use NMI Type=3



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836
5	25	4.95102	0.222821	0.04456	4.8590	5.0430
10	25	5.30658	0.222849	0.04457	5.2146	5.3986
20	25	5.66651	0.263607	0.05272	5.5577	5.7753
30	25	6.10061	0.342898	0.06858	5.9591	6.2422
40	25	6.58977	0.269051	0.05381	6.4787	6.7008
50	25	7.10913	0.214329	0.04287	7.0207	7.1976

Means Comparisons

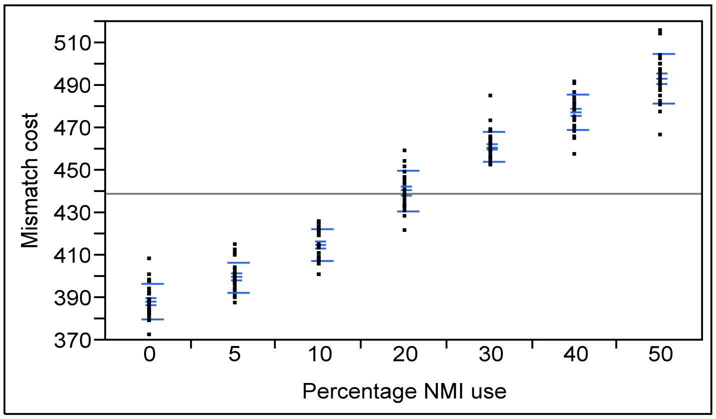
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	10	5	0
50	-0.21586	0.303498	0.792654	1.226761	1.586691	1.942251	2.00474
40	0.303498	-0.21586	0.273297	0.707403	1.067333	1.422893	1.485382
30	0.792654	0.273297	-0.21586	0.218247	0.578177	0.933737	0.996226
20	1.226761	0.707403	0.218247	-0.21586	0.14407	0.49963	0.562119
10	1.586691	1.067333	0.578177	0.14407	-0.21586	0.1397	0.202189
5	1.942251	1.422893	0.933737	0.49963	0.1397	-0.21586	-0.15337
0	2.00474	1.485382	0.996226	0.562119	0.202189	-0.15337	-0.21586

Positive values show pairs of means that are significantly different.

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=3



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.0973	1.6195	384.93	391.61
5	25	399.857	6.9805	1.3961	396.98	402.74
10	25	415.010	7.2835	1.4567	412.00	418.02
20	25	440.557	9.4119	1.8824	436.67	444.44
30	25	461.223	7.3008	1.4602	458.21	464.24
40	25	477.281	8.3334	1.6667	473.84	480.72
50	25	493.374	11.4787	2.2957	488.64	498.11

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

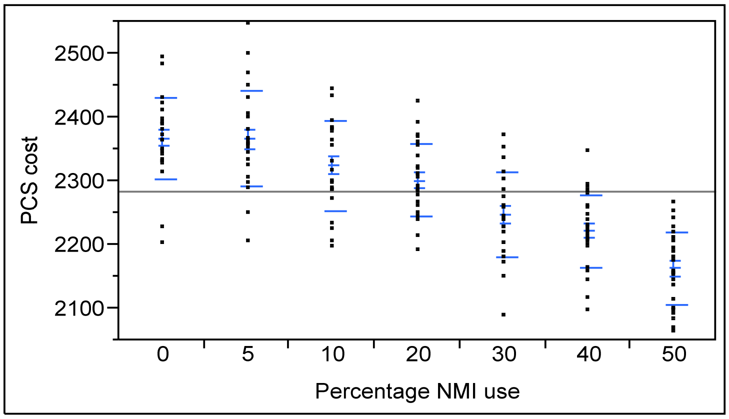
q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	10	5	0
50	-7.20813	8.885598	24.94317	45.60905	71.15654	86.30931	97.89736
40	8.885598	-7.20813	8.849441	29.51532	55.06281	70.21557	81.80362
30	24.94317	8.849441	-7.20813	13.45775	39.00523	54.158	65.74605
20	45.60905	29.51532	13.45775	-7.20813	18.33935	33.49212	45.08017
10	71.15654	55.06281	39.00523	18.33935	-7.20813	7.944634	19.53268
5	86.30931	70.21557	54.158	33.49212	7.944634	-7.20813	4.379916
0	97.89736	81.80362	65.74605	45.08017	19.53268	4.379916	-7.20813

Positive values show pairs of means that are significantly different.

d) Statistical Results of Testing Type 4

Oneway Analysis of PCS cost By Percentage NMI use NMI Type=4



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3
5	25	2365.76	75.3156	15.063	2334.7	2396.9
10	25	2324.37	70.7939	14.159	2295.2	2353.6
20	25	2301.34	56.5256	11.305	2278.0	2324.7
30	25	2246.55	66.7266	13.345	2219.0	2274.1
40	25	2221.44	57.4528	11.491	2197.7	2245.2
50	25	2162.61	58.0005	11.600	2138.7	2186.6

Means Comparisons

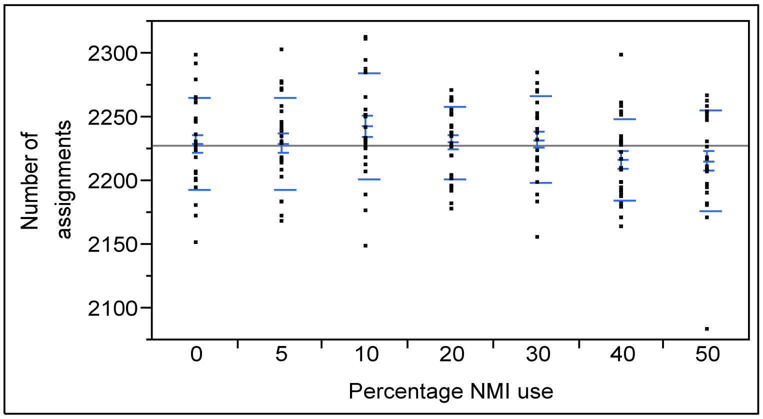
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	0	5	10	20	30	40	50
0	-54.3304	-52.9127	-11.5244	11.51199	66.30351	91.40903	150.2375
5	-52.9127	-54.3304	-12.9421	10.09425	64.88578	89.99129	148.8197
10	-11.5244	-12.9421	-54.3304	-31.294	23.49752	48.60304	107.4315
20	11.51199	10.09425	-31.294	-54.3304	0.461129	25.56665	84.39507
30	66.30351	64.88578	23.49752	0.461129	-54.3304	-29.2249	29.60355
40	91.40903	89.99129	48.60304	25.56665	-29.2249	-54.3304	4.498027
50	150.2375	148.8197	107.4315	84.39507	29.60355	4.498027	-54.3304

Positive values show pairs of means that are significantly different.

Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=4



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2
5	25	2229.68	35.9881	7.1976	2214.8	2244.5
10	25	2243.24	41.6196	8.3239	2226.1	2260.4
20	25	2230.04	27.9918	5.5984	2218.5	2241.6
30	25	2232.60	33.5857	6.7171	2218.7	2246.5
40	25	2216.84	32.2899	6.4580	2203.5	2230.2
50	25	2215.92	39.0554	7.8111	2199.8	2232.0

Means Comparisons

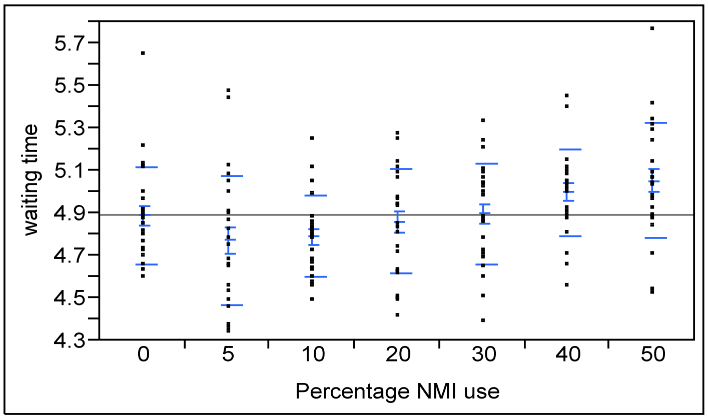
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	10	30	20	5	0	40	50
10	-29.998	-19.358	-16.798	-16.438	-15.838	-3.59797	-2.67797
30	-19.358	-29.998	-27.438	-27.078	-26.478	-14.238	-13.318
20	-16.798	-27.438	-29.998	-29.638	-29.038	-16.798	-15.878
5	-16.438	-27.078	-29.638	-29.998	-29.398	-17.158	-16.238
0	-15.838	-26.478	-29.038	-29.398	-29.998	-17.758	-16.838
40	-3.59797	-14.238	-16.798	-17.158	-17.758	-29.998	-29.078
50	-2.67797	-13.318	-15.878	-16.238	-16.838	-29.078	-29.998

Positive values show pairs of means that are significantly different.

Oneway Analysis of waiting time By Percentage NMI use NMI Type=4



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836
5	25	4.77211	0.304370	0.06087	4.6465	4.8977
10	25	4.78894	0.191456	0.03829	4.7099	4.8680
20	25	4.85999	0.246995	0.04940	4.7580	4.9619
30	25	4.89713	0.236504	0.04730	4.7995	4.9948
40	25	4.99719	0.202917	0.04058	4.9134	5.0810
50	25	5.05373	0.272610	0.05452	4.9412	5.1663

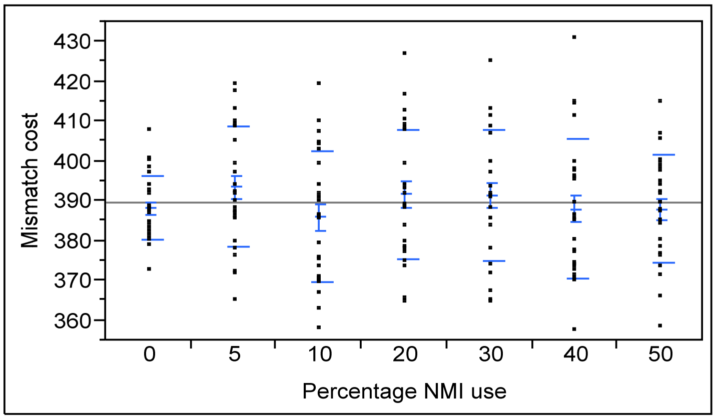
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	0	20	10	5
50	-0.20548	-0.14894	-0.04888	-0.04028	-0.01174	0.059315	0.076139
40	-0.14894	-0.20548	-0.10542	-0.09681	-0.06828	0.00278	0.019604
30	-0.04888	-0.10542	-0.20548	-0.19687	-0.16834	-0.09728	-0.08046
0	-0.04028	-0.09681	-0.19687	-0.20548	-0.17694	-0.10588	-0.08906
20	-0.01174	-0.06828	-0.16834	-0.17694	-0.20548	-0.13442	-0.11759
10	0.059315	0.00278	-0.09728	-0.10588	-0.13442	-0.20548	-0.18865
5	0.076139	0.019604	-0.08046	-0.08906	-0.11759	-0.18865	-0.20548

Positive values show pairs of means that are significantly different.

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=4



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.0973	1.6195	384.93	391.61
5	25	393.494	15.1262	3.0252	387.25	399.74
10	25	385.950	16.4776	3.2955	379.15	392.75
20	25	391.720	16.3389	3.2678	384.98	398.46
30	25	391.473	16.4089	3.2818	384.70	398.25
40	25	388.098	17.3785	3.4757	380.92	395.27
50	25	388.105	13.3603	2.6721	382.59	393.62

Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

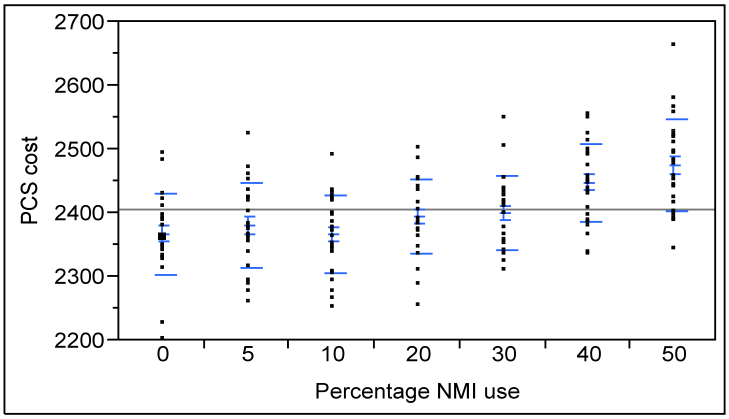
q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	5	20	30	0	50	40	10
5	-12.6923	-10.919	-10.6713	-7.46755	-7.30345	-7.29696	-5.14881
20	-10.919	-12.6923	-12.4446	-9.24081	-9.07671	-9.07022	-6.92207
30	-10.6713	-12.4446	-12.6923	-9.4885	-9.3244	-9.31791	-7.16976
0	-7.46755	-9.24081	-9.4885	-12.6923	-12.5282	-12.5217	-10.3735
50	-7.30345	-9.07671	-9.3244	-12.5282	-12.6923	-12.6858	-10.5376
40	-7.29696	-9.07022	-9.31791	-12.5217	-12.6858	-12.6923	-10.5441
10	-5.14881	-6.92207	-7.16976	-10.3735	-10.5376	-10.5441	-12.6923

Positive values show pairs of means that are significantly different.

e) Statistical Results of Testing Type 5

Oneway Analysis of PCS cost By Percentage NMI use NMI Type=5



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3
5	25	2380.17	66.1035	13.221	2352.9	2407.5
10	25	2366.55	59.8695	11.974	2341.8	2391.3
20	25	2394.05	59.1258	11.825	2369.6	2418.5
30	25	2400.00	58.5660	11.713	2375.8	2424.2
40	25	2447.30	62.3646	12.473	2421.6	2473.0
50	25	2475.08	72.7663	14.553	2445.0	2505.1

Means Comparisons

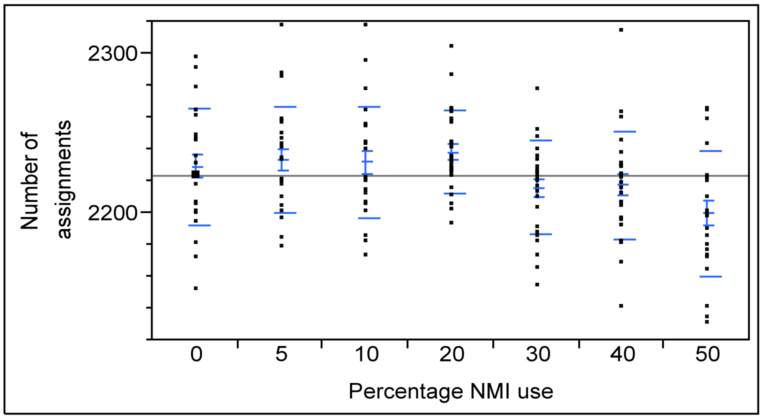
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	5	0	10
50	-53.4497	-25.6705	21.62887	27.58684	41.46464	54.45318	55.07858
40	-25.6705	-53.4497	-6.15027	-0.19229	13.6855	26.67404	27.29944
30	21.62887	-6.15027	-53.4497	-47.4917	-33.6139	-20.6254	-20
20	27.58684	-0.19229	-47.4917	-53.4497	-39.5719	-26.5834	-25.958
5	41.46464	13.6855	-33.6139	-39.5719	-53.4497	-40.4611	-39.8357
0	54.45318	26.67404	-20.6254	-26.5834	-40.4611	-53.4497	-52.8243
10	55.07858	27.29944	-20	-25.958	-39.8357	-52.8243	-53.4497

Positive values show pairs of means that are significantly different.

Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=5



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2
5	25	2233.32	33.1244	6.6249	2219.6	2247.0
10	25	2231.80	35.0678	7.0136	2217.3	2246.3
20	25	2238.24	25.8912	5.1782	2227.6	2248.9
30	25	2215.68	29.5293	5.9059	2203.5	2227.9
40	25	2217.40	34.0478	6.8096	2203.3	2231.5
50	25	2199.72	39.2529	7.8506	2183.5	2215.9

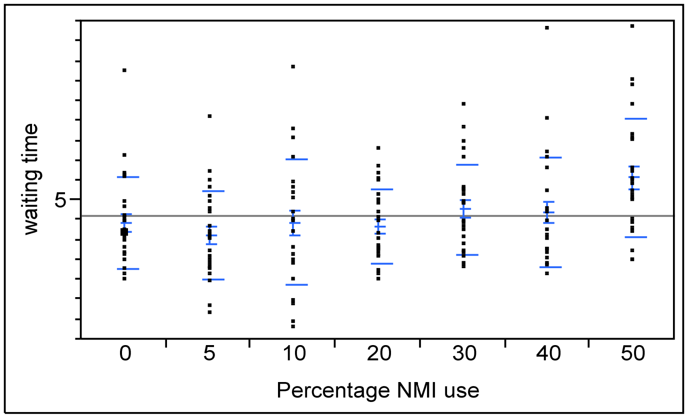
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	20	5	10	0	40	30	50
20	-28.3628	-23.4428	-21.9228	-19.2028	-7.52284	-5.80284	10.15716
5	-23.4428	-28.3628	-26.8428	-24.1228	-12.4428	-10.7228	5.237159
10	-21.9228	-26.8428	-28.3628	-25.6428	-13.9628	-12.2428	3.717159
0	-19.2028	-24.1228	-25.6428	-28.3628	-16.6828	-14.9628	0.997159
40	-7.52284	-12.4428	-13.9628	-16.6828	-28.3628	-26.6428	-10.6828
30	-5.80284	-10.7228	-12.2428	-14.9628	-26.6428	-28.3628	-12.4028
50	10.15716	5.237159	3.717159	0.997159	-10.6828	-12.4028	-28.3628

Positive values show pairs of means that are significantly different.

Oneway Analysis of waiting time By Percentage NMI use NMI Type=5



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836
5	25	4.82457	0.224280	0.04486	4.7320	4.9171
10	25	4.89035	0.313283	0.06266	4.7610	5.0197
20	25	4.86994	0.185355	0.03707	4.7934	4.9465
30	25	4.95504	0.228365	0.04567	4.8608	5.0493
40	25	4.94224	0.274045	0.05481	4.8291	5.0554
50	25	5.11479	0.296833	0.05937	4.9923	5.2373

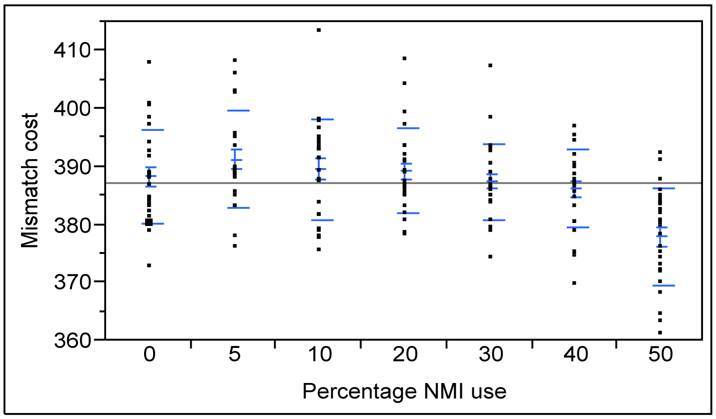
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	30	40	10	0	20	5
50	-0.21431	-0.05457	-0.04176	0.010124	0.011946	0.03053	0.075906
30	-0.05457	-0.21431	-0.20151	-0.14962	-0.1478	-0.12921	-0.08384
40	-0.04176	-0.20151	-0.21431	-0.16243	-0.1606	-0.14202	-0.09664
10	0.010124	-0.14962	-0.16243	-0.21431	-0.21249	-0.19391	-0.14853
0	0.011946	-0.1478	-0.1606	-0.21249	-0.21431	-0.19573	-0.15035
20	0.03053	-0.12921	-0.14202	-0.19391	-0.19573	-0.21431	-0.16894
5	0.075906	-0.08384	-0.09664	-0.14853	-0.15035	-0.16894	-0.21431

Positive values show pairs of means that are significantly different.

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=5



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.09728	1.6195	384.93	391.61
5	25	391.262	8.32753	1.6655	387.82	394.70
10	25	389.608	8.70395	1.7408	386.02	393.20
20	25	389.242	7.31713	1.4634	386.22	392.26
30	25	387.454	6.57182	1.3144	384.74	390.17
40	25	386.168	6.69041	1.3381	383.41	388.93
50	25	377.946	8.44460	1.6889	374.46	381.43

Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

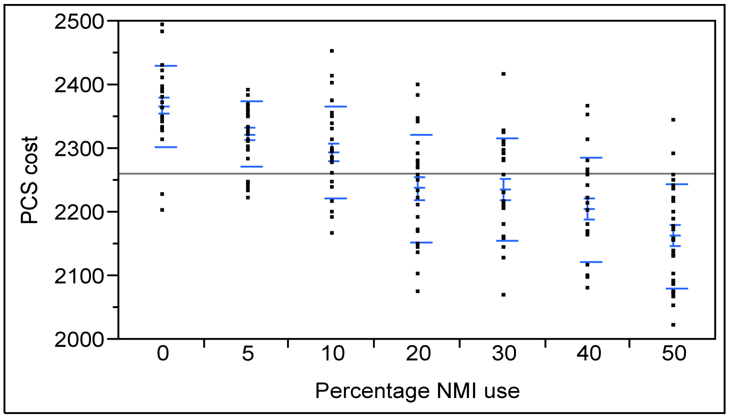
q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	5	10	20	0	30	40	50
5	-6.56549	-4.91192	-4.54624	-3.57272	-2.75737	-1.47144	6.750645
10	-4.91192	-6.56549	-6.19981	-5.2263	-4.41094	-3.12502	5.097073
20	-4.54624	-6.19981	-6.56549	-5.59198	-4.77662	-3.4907	4.731392
0	-3.57272	-5.2263	-5.59198	-6.56549	-5.75013	-4.46421	3.757881
30	-2.75737	-4.41094	-4.77662	-5.75013	-6.56549	-5.27956	2.942525
40	-1.47144	-3.12502	-3.4907	-4.46421	-5.27956	-6.56549	1.656601
50	6.750645	5.097073	4.731392	3.757881	2.942525	1.656601	-6.56549

Positive values show pairs of means that are significantly different.

f)   Statistical Results of Testing Type 6

Oneway Analysis of PCS cost By Percentage NMI use NMI Type=6



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2367.18	63.2618	12.652	2341.1	2393.3
5	25	2323.33	52.4812	10.496	2301.7	2345.0
10	25	2293.80	72.5163	14.503	2263.9	2323.7
20	25	2237.78	85.5593	17.112	2202.5	2273.1
30	25	2235.46	79.9040	15.981	2202.5	2268.4
40	25	2204.20	83.2951	16.659	2169.8	2238.6
50	25	2163.43	81.7766	16.355	2129.7	2197.2

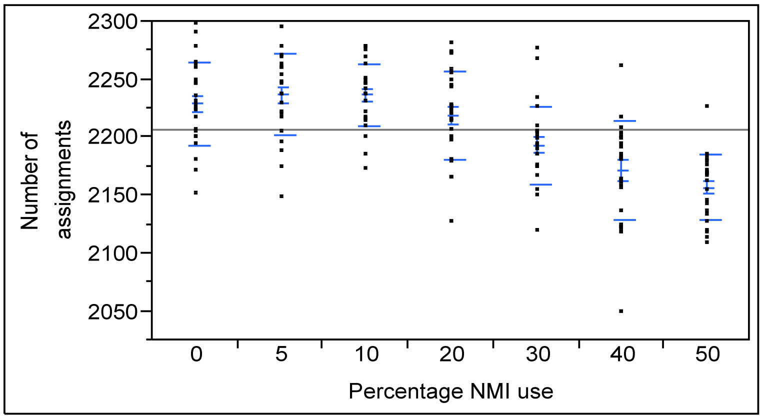
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	**0**	**5**	**10**	**20**	**30**	**40**	**50**
0	-63.285	-19.4301	10.09194	66.11109	68.43418	99.69429	140.4688
5	-19.4301	-63.285	-33.7629	22.25623	24.57932	55.83943	96.61395
10	10.09194	-33.7629	-63.285	-7.26585	-4.94276	26.31735	67.09187
20	66.11109	22.25623	-7.26585	-63.285	-60.9619	-29.7018	11.07273
30	68.43418	24.57932	-4.94276	-60.9619	-63.285	-32.0249	8.749634
40	99.69429	55.83943	26.31735	-29.7018	-32.0249	-63.285	-22.5105
50	140.4688	96.61395	67.09187	11.07273	8.749634	-22.5105	-63.285

Positive values show pairs of means that are significantly different.

Oneway Analysis of Number of assignments By Percentage NMI use NMI Type=6



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	2229.08	36.5193	7.3039	2214.0	2244.2
5	25	2236.88	35.2507	7.0501	2222.3	2251.4
10	25	2236.64	27.2594	5.4519	2225.4	2247.9
20	25	2218.56	37.9353	7.5871	2202.9	2234.2
30	25	2193.16	33.9039	6.7808	2179.2	2207.2
40	25	2171.80	42.9942	8.5988	2154.1	2189.5
50	25	2156.84	27.8219	5.5644	2145.4	2168.3

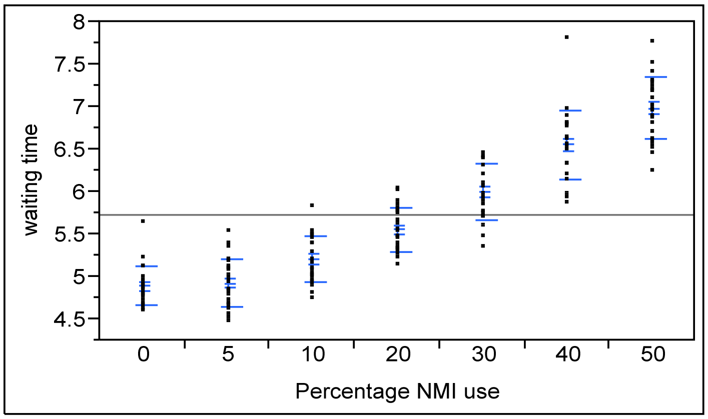
Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	5	10	0	20	30	40	50
5	-29.4673	-29.2273	-21.6673	-11.1473	14.25274	35.61274	50.57274
10	-29.2273	-29.4673	-21.9073	-11.3873	14.01274	35.37274	50.33274
0	-21.6673	-21.9073	-29.4673	-18.9473	6.452744	27.81274	42.77274
20	-11.1473	-11.3873	-18.9473	-29.4673	-4.06726	17.29274	32.25274
30	14.25274	14.01274	6.452744	-4.06726	-29.4673	-8.10726	6.852744
40	35.61274	35.37274	27.81274	17.29274	-8.10726	-29.4673	-14.5073
50	50.57274	50.33274	42.77274	32.25274	6.852744	-14.5073	-29.4673

Oneway Analysis of waiting time By Percentage NMI use NMI Type=6



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	4.88853	0.230337	0.04607	4.7935	4.9836
5	25	4.92245	0.282558	0.05651	4.8058	5.0391
10	25	5.20623	0.275336	0.05507	5.0926	5.3199
20	25	5.55535	0.255381	0.05108	5.4499	5.6608
30	25	6.00220	0.327785	0.06556	5.8669	6.1375
40	25	6.55258	0.404751	0.08095	6.3855	6.7197
50	25	6.98538	0.364631	0.07293	6.8349	7.1359

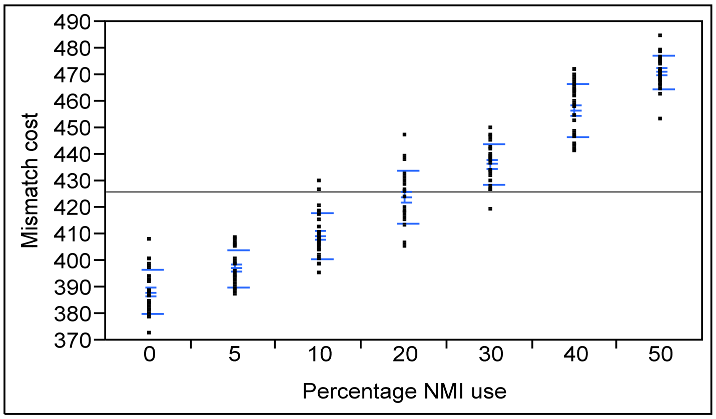
Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	10	5	0
50	-0.26273	0.17007	0.720452	1.167302	1.516415	1.800198	1.834121
40	0.17007	-0.26273	0.287649	0.734499	1.083612	1.367395	1.401317
30	0.720452	0.287649	-0.26273	0.184117	0.53323	0.817013	0.850935
20	1.167302	0.734499	0.184117	-0.26273	0.08638	0.370163	0.404085
10	1.516415	1.083612	0.53323	0.08638	-0.26273	0.02105	0.054972
5	1.800198	1.367395	0.817013	0.370163	0.02105	-0.26273	-0.22881
0	1.834121	1.401317	0.850935	0.404085	0.054972	-0.22881	-0.26273

Positive values show pairs of means that are significantly different.

Oneway Analysis of Mismatch cost By Percentage NMI use NMI Type=6



Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
0	25	388.269	8.09728	1.6195	384.93	391.61
5	25	397.184	7.06936	1.4139	394.27	400.10
10	25	409.599	8.72038	1.7441	406.00	413.20
20	25	423.930	9.78118	1.9562	419.89	427.97
30	25	436.520	7.78965	1.5579	433.30	439.74
40	25	456.512	9.89473	1.9789	452.43	460.60
50	25	471.148	6.22770	1.2455	468.58	473.72

Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.98440	0.05

Abs(Dif)-LSD	50	40	30	20	10	5	0
50	-7.02356	7.613081	27.60458	40.1951	54.52559	66.94054	75.85597
40	7.613081	-7.02356	12.96794	25.55846	39.88895	52.3039	61.21933
30	27.60458	12.96794	-7.02356	5.566959	19.89745	32.3124	41.22784
20	40.1951	25.55846	5.566959	-7.02356	7.306938	19.72189	28.63732
10	54.52559	39.88895	19.89745	7.306938	-7.02356	5.391391	14.30683
5	66.94054	52.3039	32.3124	19.72189	5.391391	-7.02356	1.891877
0	75.85597	61.21933	41.22784	28.63732	14.30683	1.891877	-7.02356

Positive values show pairs of means that are significantly different.

APPENDIX B

A. CODE

The simulation program is written in the MATLAB programming environment. MATLAB allows standard commands such as for, if, while and else. Variables do not have to be defined in advance, as in other programming languages. Procedures can be saved in “m-files”. The same applies to functions. Variables that are used in one procedure can be reused in other procedures, because they are global. The only exception is reusing variables within functions. If a function needs specific information to operate, this information has to be given to the function and the result will be given back by the function. In addition, MATLAB offers a set of functions that can be used in programs. Variables or sets of variables can be saved in mat-files.

The simulation is written in the MATLAB environment release 2009.
The following pages provide the simulation procedures. The main procedures are followed by the sub procedures. Finally functions written specifically for this model are shown. Before introducing the procedures, the most important variables are explained.

B. VARIABLES

Most of the variables are matrixes. Each row in these matrixes is a vector. Each vector contains a set of data. MATLAB uses data by calling the row and column of a variable. Example: In the billet matrix, the row number equals the billet identification number. The billet identification number is content of the first column. The platform type is written down in the fifth column. This means that billets(123,5) equals the platform type of the 123th billet (which is a type 5 billet: a NAS billet).
The complete “billets” vector is:

$$billet = \begin{bmatrix} billet\ ID \\ required\ rank \\ sea - shore\ indicator \\ region \\ plattform \\ sailorID \end{bmatrix}$$

The “sailor_com” vector includes:

$$Sailor = \begin{bmatrix} (1)\ sailor\ ID \\ (2)\ month\ of\ service\ (TIS) \\ (3)\ rank \\ (4)\ month\ in\ rank\ (TIR) \\ (5)\ sea(1) - shore(0) \\ (6)\ attrition\ date\ in\ month\ of\ service \\ (7)\ next\ promotion\ date\ in\ month\ of\ service \\ (8)\ nmi\ is\ indicated \\ (9)\ typ\ of\ NMI \\ (10)\ favorit\ region \\ (11) favorit\ platform \end{bmatrix}$$

The ”sailor_without_billet” file contains:

$$sailor\ without\ billet = \begin{bmatrix} sailor\ ID \\ time\ the\ billet\ is\ needed \\ priority\ bit\ (sailors\ without\ former\ billet) \end{bmatrix}$$

The “empty_billet” vector:

$$empty\ billet = \begin{bmatrix} billet\ ID \\ time\ \ billet\ is\ available \end{bmatrix}$$

The “assignments” matrix contains vectors of:

$$assignments = \begin{bmatrix} sailorID \\ old\ billetID \\ new\ billetID \\ time\ \ of\ assign. \\ time\ \ asst. is\ late \\ rank\ of\ sailor \\ NMI\ involved \end{bmatrix}$$

The “result” vector of an experiment:

$$result = \begin{bmatrix} (1)\ Average\ PCS\ cost \\ (2)\ Number\ of\ assignments \\ (3)\ Average\ waiting\ time \\ (4)\ Number\ of\ NMI \\ (5)\ Number\ of\ NMI\ assignments \\ (6)\ Average\ NMI\ assi.\ waiting\ t. \\ (7)\ Average\ mismatch\ cost\ per\ billet\ and\ month \\ (8)\ simulation\ time \\ (9)\ start\ point\ of\ documentation \\ (10)\ percentage\ NMI\ use \\ (11) NMI\ type \end{bmatrix}$$

C. MAIN PROCEDURES

Procedures are considered main procedures when they are used in the main monthly assignment loop or are used above this level. All procedures used within the main procedures, to make the code more readable, are called sub procedures.

1. Start

The start procedure has two functions. First, the user is asked for a (random) number, which leads to a use of this amount of random numbers to set the pattern for the other random numbers. The goal is to start the simulation with different random numbers to get different results. MATLAB will use the same random number if it is restarted, unless otherwise directed, which leads to identical results if all other parameters are the same too. Second the function calls the procedure main_prog. The procedure should be used the first time, the simulation should be executed.

```
% routine to set up the rand generator by using a different number of
% random numbers
x=input('put in a random integer between 100 and 10000 ...');
for i=1:x
    r=rand;
end
clear
main_prog
```

2. Main_prog

The main_prog is used to set up the simulation. Inputs such as time, number of experiments and type of experiment are required. Further the procedure runs the simulation as it is asked for.

```
%main routine
%clearing the memory
clear
% reading setting
%time of one simulation run
month=input('Month of simulation? ');
% start point of counting
start=input('Start point of taking assignments into statistic? ');
%number of experiments
sim_number=input('Number of experiments? ');
% percentage of nmi use
percent=input('Percentage of NMI use? ');
% input of nmi that should be simulated
nmi_typ=input('Which NMI typ should be simulated? ');
result=[];
zaehl=0;
for count=1:sim_number
    sim_run
end,
```

3. Sim_run

The sim_run procedure loads variables, further defines new variables and finally includes the simulation loop.

```
%simulation routine
clc
var_load
[last_sailor_id,l]=size(sailor_com);
%counter that plots the number of new sailors who enter the system
new_sailors_month=[];
%variable to take a look on the sailor distribution at different
%times
history=[];
sailor_stat;
%calculation of the attrition point and promotion point for the given
%sailors
attrition_time1
promotion_time1
%adding 6 more columns to the sailor matrix for later use
for i=6:11
    sailor_com(1,i)=0;
end,
%sailor distribution after first promotions
sailor_stat;
%definition/reset of variables used later in the simulation
```

```
empty_billets=[];
sailor_without_billet=[];
dropout=[];
assignments=[];
old_assignments=[];
mismatch_cost=0;
mismatch_cost_billet=0;
for time=1:month
    %first rows include a counter that shows the percentage of month
    %of simulation done
    clc
    percent_done=zaehl*100/(sim_number*month)
    zaehl=zaehl+1;
    %begin of simulation loop
    growth;
    attrition_proc;
    promotion;
    future_assi;
    find_assi;
    sailor_move;
    aging;
    %next sub procedure proofs if still all billets are in use
    billet_control;
    %sub procedure that sums up the mismatch cost for this month
    sum_mismatch_cost;
end
%statistic about the sailor community at the end of simulation time
sailor_stat;
cost_stat;
```

4. Growth

This procedure generates new sailors. The number of new sailors corresponds to the number of sailors who left one month earlier. Every new sailor gets a random calculated attrition date, promotion date and personal wishes for a NMI type, favorite homeport and platform. The used method to calculate these random dates is a Monte Carlo simulation.

```
%adding the number of new sailors to the system that left earlier
[k,l]=size(dropout);
%for loop creates k new sailors
for i=last_sailor_id+1:last_sailor_id+k
    r1=rand;
    r2=rand;
    % Monte Carlo simulation to calculate the attrition date
    if (r1>0.6070)
        attrition_date=240;
    else
        m=1;
```

```
        while (m<224)
            if (r1<attrition(m,2))
                attrition_date=attrition(m,1);
                m=224;
            else
                m=m+1;
            end,
        end,
    end,
    % Monte Carlo simulation to calculate the next promotion date
    m=1;
    while (m<38)
        if (r2<promo_distrib_5(m,2))
            promo_date=35+promo_distrib_5(m,1);
            m=38;
        else
            m=m+1;
        end,
    end,
    % random NMI use yes/no related to the percentage of NMI offered
    r=randi(18);
    r3=rand;
    if (r3>percent/100)
        nmi_use=0;
    else
        nmi_use=1;
    end,
    if (nmi_typ==7)
        nmit=randi(6);
    else
        nmit=nmi_typ;
    end
    new_sailor=[  i  16  4  2  1  attrition_date  promo_date  nmi_use  nmit
pos_pf(r,1) pos_pf(r,2)];
    sailor_com=[sailor_com ; new_sailor];
    sailor_without_billet=[sailor_without_billet ; i time 1];
end,
new_sailors_month=[new_sailors_month ; k];
dropout=[];
last_sailor_id=last_sailor_id+k;
clear new_sailor
clear promo_date
clear attrition_date
clear nmi_use
clear r
clear r1
clear r2
clear r3
clear i
clear k
clear l
clear m
```

5. Attrition

The attrition_proc takes care of all sailors who reach their attrition date or the high-year-tenure limitation of their rank. These sailors have to leave the system. If these sailors have a billet in use, this billet is considered free and can be used for another sailor later.

```
%procedure simulates attrition for sailor_com
[k,l]=size(sailor_com);
for n=k:-1:1
    % eliminating sailors who reach their attrition date
    if (sailor_com(n,2)==sailor_com(n,6))
        dropout=[dropout ; sailor_com(n,1)];
        b_id=billet_id(billets,sailor_com(n,1));
        % if the leaving sailor has a billet he is on, this billet is
to
        % called to be free
        if (b_id>0)
            empty_billets=add_billet_to_eb(empty_billets,   assignments,
b_id, time);
            billets(b_id,6)=0;
        end,

sailor_without_billet=del_sailor_from_swob(sailor_without_billet,sailor
_com(n,1));

empty_billets=set_future_billet_free(empty_billets,assignments,sailor_c
om(n,1));
        assignments=del_sailor_from_assi(assignments,sailor_com(n,1));
        sailor_com(n,:)=[];
    % eliminating sailors who reach the high year tenure limitation
    elseif ((sailor_com(n,2)==120) && (sailor_com(n,3)==4))
        dropout=[dropout ; sailor_com(n,1)];
        b_id=billet_id(billets,sailor_com(n,1));
        % if the leaving sailor has a billet he is on, this billet is
to
        % called to be free
        if (b_id>0)
            empty_billets=add_billet_to_eb(empty_billets,   assignments,
b_id, time);
            billets(b_id,6)=0;
        end,

sailor_without_billet=del_sailor_from_swob(sailor_without_billet,sailor
_com(n,1));

empty_billets=set_future_billet_free(empty_billets,assignments,sailor_c
om(n,1));
        assignments=del_sailor_from_assi(assignments,sailor_com(n,1));
        sailor_com(n,:)=[];
    end,
end
clear n
```

```
clear k
clear l
clear b_id
```

6. Promotion

In this procedure, sailors who reach their promotion date get promoted and a new promotion date will be calculated. If the sailor has more than 18 months to go in his tour, and the required rank of his billet is lower than his new rank, he will get a new billet. The same adjustment occurs if the required rank is more than one rank below his new rank.

```
% promotion with calculation of the next promotion date
[k,l]=size(sailor_com);
for n=k:-1:1
    %proof if a sailor can be promoted
    if (sailor_com(n,2)==sailor_com(n,7))
        %Promotion
        sailor_com(n,3)=sailor_com(n,3)+1;
        sailor_com(n,4)=0;
        %calculate time in tour (Sea-Shore)
        tit=sea_shore_date(sailor_com(n,2))-sailor_com(n,2);
        %required rank of sailors billet
        b_id=billet_id(billets,sailor_com(n,1));
        %if the sailor has a billet and the required rank is lower than
his
        %actual rank and he has more than 18 month of his tour he is in
        %left, he should get a new billet.
        if (b_id>0)
            rr=billets(b_id,2);
            if                        ((tit>=18                       &&
sailor_com(n,3)>rr)||(sailor_com(n,3)>rr+1))

sailor_without_billet=add_sailor_to_swob(sailor_without_billet,
assignments, sailor_com(n,1),time);
            end,
        end,
        % calculation of a new promotion date (Monte Carlo simulation)
        r=rand;
        m=1;
        while (m<l_prom_d+1)
            if (r<promo_distrib_6_7(m,2))
                promo_date=promo_distrib_6_7(m,1);
                m=l_prom_d+1;
            else
                m=m+1;
            end,
        end,
        if (sailor_com(n,3)==5)
            if (84>sailor_com(n,2)+36)
```

```
                sailor_com(n,7)=84+promo_date;
            else
                sailor_com(n,7)=sailor_com(n,2)+36+promo_date;
            end,
        end,
        if (sailor_com(n,3)==6)
            if (132>sailor_com(n,2)+36)
                sailor_com(n,7)=132+promo_date;
            else
                sailor_com(n,7)=sailor_com(n,2)+36+promo_date;
            end,
        end,
    end,
end,
clear k
clear l
clear m
clear n
clear r
clear promo_date
clear tit
clear rr
```

7. Future Assignments

This procedure writes billets that will be free in the future to the file of empty billets, and also, writes available sailors to the file of sailors_without_billet if their new sea-shore rotation is coming up. Both events happened 9 month in advance.

```
%billet and sailor search for future assignments
[k,l]=size(sailor_com);
for n=k:-1:1
    %sailor who have to leave the system for sure with TIS=240
    %will give there billet back
    if (sailor_com(n,2)==231)
        b_id=billet_id(billets,sailor_com(n,1));
        if (b_id>0)

empty_billets=add_billet_to_eb(empty_billets,assignments,b_id, time+9);
        end,
    end,
    %sailor who finish their tour in 9 month have to get a new billet
    if (sea_shore_date(sailor_com(n,2))-sailor_com(n,2)==9)
        if (sailor_com(n,2)==39 && sailor_com(n,8)==1)

sailor_without_billet=add_sailor_to_swob_prio(sailor_without_billet,
assignments, sailor_com(n,1), time+9);
        else
```

```
sailor_without_billet=add_sailor_to_swob(sailor_without_billet,
assignments, sailor_com(n,1), time+9);
        end
        %set change in needed billet type:
        if (sailor_com(n,5)==1)
            sailor_com(n,5)=0;
        else
            sailor_com(n,5)=1;
        end,
    end,
end,
clear k
clear l
clear n
```

8. Find Assignments

This procedure removes all E7s who are looking for a new billet from the system; the simulation is designed for E4 to E6 sailors and has no E7 billets. ..In a second step, the file with sailors looking for a billet is sorted. Sailors with priority come first, within the groups; sailors who needed the billet first, have first priority. Then, a decision about the matching procedure to use is made and the proper matching procedure is called. For each sailor, all available billets will be checked if they match with the given criteria.

```
%this procedure looks for possible assignments
nnm=1;%no_new_matches
sailor_move;
%removing E7 who are looking for a new billet out of the system
[sn,l]=size(sailor_without_billet);
for n=sn:-1:1
    if (rank(sailor_com, sailor_without_billet(n,1))==7)
        sailor_com=del_sailor_from_sc(sailor_com,
sailor_without_billet(n,1));
        dropout=[dropout ; sailor_without_billet(n,1)];
        b_id=billet_id(billets,sailor_without_billet(n,1));
        if (b_id>0)
            billets(b_id,6)=0;
            empty_billets=add_billet_to_eb(empty_billets,   assignments,
b_id, time);
        end,
        sailor_without_billet(n,:)=[];
    end,
end,
while (nnm<2)
    %match_criteria
    mc=4;
    while (mc>0)
        %sort sailor file first by priority bit and second by time
```

```
        sailor_without_billet=sortrows(sailor_without_billet,[-3 2]);
        [sn,l]=size(sailor_without_billet);
        n=1;
        %end whole procedure if no sailor is looking for a billet
        if (sn==0)
            nnm=2;
        end,
        [bn,p]=size(empty_billets);
        while (n<sn+1)
            m=1;
            ss=sea_shore_ident(sailor_com, sailor_without_billet(n,1));
            ra=rank(sailor_com, sailor_without_billet(n,1));
            re=region(billets, sailor_without_billet(n,1));
            while (m<bn+1)
                %proof  if  a  match  procedure  with  or  w/o  NMI  match
criteria
                %has to be used
                if (nmi_t(sailor_com,sailor_without_billet(n,1))>0)
                    if    (mos(sailor_com,sailor_without_billet(n,1))>38
&& nmi_i(sailor_com,sailor_without_billet(n,1))==1)
                        match_w_nmi
                    else
                        match_wo_nmi
                    end,
                else
                    match_wo_nmi
                end,
                if (mc<2)
                    m=bn+1;
                end,
            end,
            n=n+1;
        end,
        %if no new match was found, the match criteria gets lowered
        [cn,l]=size(sailor_without_billet);
        if (cn==sn)
            mc=mc-1;
        end,
    end,
    nnm=2;
end,
clear nnm
clear m
clear sn
clear l
clear bn
clear p
clear n
clear mc
clear cn
clear ss
clear ra
clear re
clear br
clear t
```

9. Sailor Move

If the time is right, this procedure moves the sailor from his old billet to the new billet and writes the old billet to the empty billet file.

```
%sailor movement
[k,l]=size(assignments);
for n=k:-1:1
    % time check
    if (time>=assignments(n,4))
        % check if the billet is free, so the sailor can move on
        if (billets(assignments(n,3),6)==0)
            billets(assignments(n,3),6)=assignments(n,1);
            %set old billet of sailor free for another sailor
            if (assignments(n,2)>0)
                billets(assignments(n,2),6)=0;
            end,
            %save assignment to file
            old_assignments=[old_assignments ; assignments(n,:)];
            %delete old executed assignment
            assignments(n,:)=[];
        end
    end,
end,
```

10. Aging

This procedure includes three steps. First, all sailors with TIS of 240 leave the simulation and the billet becomes available. Second, all remaining sailors become one month older. Third, sailors who chose to take a NMI offer get checked. If the window of NMI use is over, the NMI indicator in the sailor file is reset.

```
%aging sailor comunity by one month
[k,l]=size(sailor_com);
for n=k:-1:1
    %sailors who reach TIS=240 have to leave and the billet gets set
free
    if (sailor_com(n,2)==240)
        dropout=[dropout ; sailor_com(n,1)];
        b_id=billet_id(billets,sailor_com(n,1));
        billets(b_id,6)=0;
        if (time<=10 && b_id>0)
            empty_billets=add_billet_to_eb(empty_billets,   assignments,
b_id,time);
        end,

sailor_without_billet=del_sailor_from_swob(sailor_without_billet,sailor
_com(n,1));
```

```
empty_billets=set_future_billet_free(empty_billets,assignments,sailor_c
om(n,1));
        assignments=del_sailor_from_assi(assignments,sailor_com(n,1));
        sailor_com(n,:)=[];
    %all other sailors get older
    else
        sailor_com(n,2)=sailor_com(n,2)+1;
        sailor_com(n,4)=sailor_com(n,4)+1;
        %reset the NMI indicator when time for NMI use is over
        if      (sailor_com(n,8)==1      &&      sailor_com(n,9)<4      &&
sailor_com(n,2)==84)
            sailor_com(n,8)=0;
        elseif     (sailor_com(n,8)==1     &&     sailor_com(n,9)>3     &&
sailor_com(n,2)==120)
            sailor_com(n,8)=0;
        end,
    end,
end,
clear k
clear l
clear n
clear b_id
```

D. 1SUB PROCEDURES

The idea behind these procedures is, that the code should be better readable and steps that have to be used more than once in the program can use the same code easily.

1. Sailor Statistic

This procedure is not relevant for the cost simulation process. The results have been used to validate to model. Average data collected by this procedure is shown in Figure 5.

```
%sailor statistic
%procedure counts sailors with same rank and years of service
%result of this procedure can be plotted as excel bar graph
statistic=[];
[i,j]=size(sailor_com);
for year=1:20
    count4=0;
    count5=0;
    count6=0;
    count7=0;
    for z=1:i
        if   (sailor_com(z,2)<12*year   &&   sailor_com(z,2)>12*(year-1)&&
sailor_com(z,3)==4)
```

```
                count4=count4+1;
            end
        if  (sailor_com(z,2)<12*year  &&  sailor_com(z,2)>12*(year-1)&&
sailor_com(z,3)==5)
                count5=count5+1;
        end
        if  (sailor_com(z,2)<12*year  &&  sailor_com(z,2)>12*(year-1)&&
sailor_com(z,3)==6)
                count6=count6+1;
        end
        if  (sailor_com(z,2)<12*year  &&  sailor_com(z,2)>12*(year-1)&&
sailor_com(z,3)==7)
                count7=count7+1;
        end
    end
    statistic(1,year)=count4;
    statistic(2,year)=count5;
    statistic(3,year)=count6;
    statistic(4,year)=count7;
end
history=[history ; statistic];
clear count4
clear count5
clear count6
clear count7
clear i
clear j
clear z
clear year
```

2. Billet Control

This procedure proofs if all billets, that are empty, are still in use at the end of the month. If these billets are not in account, they will be written to the empty billet file.

```
% procedure put empty billets back into file with empty billets
for i=1:553
    if (billets(i,6)==0)
        empty_billets=add_billet_to_eb(empty_billets,  assignments,  i,
time);
    end,
end,
```

3. Sum Mismatch Cost

At the end of each simulation month, this procedure sums the mismatch related costs.

```
%routins sums up the billets mismatch cost, cost that are arise because
%there is not the right sailor available for a billet
rank_cost=[0 0 0 1986 2305 2801 3505];
c=0;
for i=1:553
    if (billets(i,6)>0)
        r=rank(sailor_com,billets(i,6));
        if (billets(i,2)==r)
            c=0;
        else
            c=abs(rank_cost(billets(i,2))-rank_cost(r));
        end,
    else
        c=rank_cost(billets(i,2));
    end
    mismatch_cost=mismatch_cost+c;
end,
clear i
clear c
clear r
```

4. Match without NMI

This procedure follows the match criteria in table XZ. No match with the given criteria switches the criteria to the use of the 1st alternative criteria and later to the 2nd alternative of criteria.

```
%matching without nmi
if (mc==4)
    %sea-shore-check and time check
    if                (ss==billets(empty_billets(m,1),3)              &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
        %rank check
        if (ra==billets(empty_billets(m,1),2))
            %region check
            if (re==billets(empty_billets(m,1),4))
                write_match
            else
                m=m+1;
            end
        else
            m=m+1;
        end
    else
        m=m+1;
    end,
end,
if (mc==3)
    %sea-shore-check and time check
```

```
    if              (ss==billets(empty_billets(m,1),3)              &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
        %rank check
        if (ra==billets(empty_billets(m,1),2))
            write_match
        else
            m=m+1;
        end
    else
        m=m+1;
    end,
end,
if (mc==2)
    %sea-shore-check and time check
    if              (ss==billets(empty_billets(m,1),3)              &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
        %rank check (rank=rank+1)
        br=billets(empty_billets(m,1),2)-1;
        if (ra==br)
            write_match
        else
            m=m+1;
        end
    else
        m=m+1;
    end,
end,
if (mc<2)
    m=bn+1;
end,
```

5. Match with NMI

Depending on the NMI type the sailor has chosen or is tested, this procedure proofs if a billet fits a sailor. The match criteria are shown in table XZ.

```
%matching with nmi
typ=nmi_t(sailor_com,sailor_without_billet(n,1));
%nmi typ 1 (region must match with wish of sailor)
if (typ==1 || typ==5)
    if (mc==4)
        %sea-shore-check and time check
        if             (ss==billets(empty_billets(m,1),3)             &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
            %rank check
            if (ra==billets(empty_billets(m,1),2))
                %region check
                if                                 (nmi_r(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),4))
                    write_match_nmi
```

```
                else
                    m=m+1;
                end
            else
                m=m+1;
            end
        else
            m=m+1;
        end,
    end,
    if (mc==3)
        %sea-shore-check and time check
        if              (ss==billets(empty_billets(m,1),3)             &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
            %rank check (rank=rank+1)
            br=billets(empty_billets(m,1),2)-1;
            if (ra==br)
                %region check
                if                                    (nmi_r(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),4))
                    write_match_nmi
                else
                    m=m+1;
                end
            else
                m=m+1;
            end
        else
            m=m+1;
        end,
    end,
    if (mc<3)
        m=bn+1;
    end,
end,
%nmi typ 2 (platform must match with wish of sailor)
if (typ==2)
    if (mc==4)
        %time check
        if
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2))
        %rank check
            if (ra==billets(empty_billets(m,1),2))
                %platform check
                if                                    (nmi_p(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),5))
                    write_match_nmi
                else
                    m=m+1;
                end
            else
                m=m+1;
            end
```

```
        else
             m=m+1;
        end,
    end,
    if (mc==3)
    %time check
    if
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2))
        %rank check (rank=rank+1)
        br=billets(empty_billets(m,1),2)-1;
        if (ra==br)
            %platform check
            if                                        (nmi_p(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),5))
                write_match_nmi
            else
                m=m+1;
            end
        else
            m=m+1;
        end
    else
         m=m+1;
    end,
    end,
    if (mc<3)
        m=bn+1;
    end,
end,
%nmi typ 3 (region andplatform must match with wish of sailor)
if (typ==3 || typ==6)
    if (mc==4)
        %time check
        if
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2))
            %rank check
            if (ra==billets(empty_billets(m,1),2))
                %platform and region check
                if                                    (nmi_p(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),5)            &&
nmi_r(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),4))
                    write_match_nmi
                else
                    m=m+1;
                end
            else
                m=m+1;
            end
        else
             m=m+1;
        end,
    end,
```

```
    if (mc==3)
    %time check
    if
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2))
        %rank check (rank=rank+1)
        br=billets(empty_billets(m,1),2)-1;
        if (ra==br)
            %platform and region check
            if                                        (nmi_p(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),5)            &&
nmi_r(sailor_com,
sailor_without_billet(n,1))==billets(empty_billets(m,1),4))
                write_match_nmi
            else
                m=m+1;
            end
        else
            m=m+1;
        end
    else
         m=m+1;
    end,
    end,
    if (mc<3)
        m=bn+1;
    end,
end,
% nmi typ 4, regional stability for two tours
if (typ==4)
    if (mc==4)
        %sea-shore-check and time check
        if              (ss==billets(empty_billets(m,1),3)             &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
            %rank check
            if (ra==billets(empty_billets(m,1),2))
                %region check
                b_id=billet_id(billets, sailor_without_billet(n,1));
                if (b_id>0)
                    re=billets(b_id,4);
                else
                    re=1;
                end,
                if (re==billets(empty_billets(m,1),4))
                    write_match_nmi
                else
                    m=m+1;
                end
            else
                m=m+1;
            end
    else
         m=m+1;
    end,
```

```
    end,
    if (mc==3)
    %sea-shore-check and time check
    if               (ss==billets(empty_billets(m,1),3)              &&
(sailor_without_billet(n,2)>=empty_billets(m,2)||time>=empty_billets(m,
2)))
        %rank check (rank=rank+1)
        br=billets(empty_billets(m,1),2)-1;
        if (ra==br)
            %region check
            b_id=billet_id(billets, sailor_without_billet(n,1));
            if (b_id>0)
                re=billets(b_id,4);
            else
                re=1;
            end,
            if (re==billets(empty_billets(m,1),4))
                write_match_nmi
            else
                m=m+1;
            end
        else
            m=m+1;
        end
    else
         m=m+1;
    end,
    end,
    if (mc<3)
        m=bn+1;
    end,
end,
clear typ
```

6. Write Match

This procedure adds the found match to the assignment file. In a second step the sailor's old billet is added to the file of empty billets.

```
%procedure writes found match in the list of assignments
%calculating the point of the assignment
if (sailor_without_billet(n,2)>=time)
    t=sailor_without_billet(n,2);
else
    t=time;
end,
new_assi=[sailor_without_billet(n,1)                  billet_id(billets,
sailor_without_billet(n,1))        empty_billets(m,1)        t        t-
sailor_without_billet(n,2) rank(sailor_com, sailor_without_billet(n,1))
0];
assignments=[assignments; new_assi];
%add sailors old billet to 'empty_billets'
b_id=billet_id(billets,sailor_without_billet(n,1));
```

```
if (b_id>0)
    empty_billets=add_billet_to_eb(empty_billets,   assignments,   b_id,
t);
end,
sailor_without_billet(n,:)=[];
empty_billets(m,:)=[];
clear new_assi
sailor_move;
%restart the matching process
n=sn+1;
m=bn+1;
```

7. Cost Statistic

This procedure writes all information collected from one experiment into a result file. To do so, the procedure does the necessary calculation first.

```
%procedure calculates costs and other statistics based on information
in
%the variable old-assignments
[r,p]=size(old_assignments);
%multiplier for sailors alowed pcs costs
multi=[0 0 0 14 14 15];
%calculation of the mismatch cost per month and billet
mismatch_cost_billet=mismatch_cost/(553*(month-start));
%deleting of older assignments that where executed before point of
interest
for i=r:-1:1
    if (start>old_assignments(i,4))
        old_assignments(i,:)=[];
    end,
end,
[r,p]=size(old_assignments);
%set numbers bak to zero/creating variables used later
total_pcs=0;
tot_wait=0;
number_nmi=0;
number_nmi_assi=0;
tot_wait_nmi=0;
%cost calculation for all assignments
for i=1:r
    %looking for the region the sailor started his movement
    if (old_assignments(i,2)==0)
        reg_old_billet=1;
    else
        reg_old_billet=billets(old_assignments(i,2),4);
    end,

total_pcs=total_pcs+pcs(reg_old_billet,billets(old_assignments(i,3),4))
*multi(old_assignments(i,6));
    tot_wait=tot_wait+old_assignments(i,5);
    %counting the number of assignment found by using NMI match
criteria
```

```
    %and counting the number of offered NMI
    if (old_assignments(i,7)==1)
        c=0;
        for j=1:i-1
            if      (old_assignments(j,1)==old_assignments(i,1)      &&
old_assignments(j,7)==1)
                c=1;
                j=i-1;
            end
        end
        if (c==0)
            number_nmi=number_nmi+1;
        end,
        number_nmi_assi=number_nmi_assi+1;
        tot_wait_nmi=tot_wait_nmi+old_assignments(i,5);
    end,
end,
%calculation of average values
ave_cost=total_pcs/r;
ave_wait=tot_wait/r;
ave_wait_nmi=tot_wait_nmi/number_nmi_assi;
%documentation of experiments results
result=[result;   ave_cost   r   ave_wait   number_nmi   number_nmi_assi
ave_wait_nmi mismatch_cost_billet month start percent nmi_typ];
```

8. Variable Load

At the beginning of each experiment, the variables needed are loaded from the hard drive. The variables are saved in different files to facilitate changes in the simulation set up.

```
%load variables for simulation
load sailor_com.mat
load billets.mat
%PCS cost file
load pcs.mat
%Attrition distribution
load attrition.mat
%Promotion distribution to E5
load promotion5.mat
%Promotion distribution to E6/E7
load promotion6_7.mat
%List of possible homeport-platform combinations
load pos_pf.mat
```

E. FUNCTIONS

1. Add Billet to Empty Billet

This function adds a billet to the matrix of empty billets. Before it does so, it checks if the billet is already in the file. If the billet is already in, the procedure only might change the time the billet is free.

```
%function that adds a billet to empty billets or change the time the billet
%will be free
function[eb]=add_billet_to_eb(eb,assi, billet_id, t)
[r,p]=size(eb);
[q,s]=size(assi);
m=1;
c=0;
%check if billet is already in the file
while (m<r+1)
    if (billet_id==eb(m,1))
        if (eb(m,2)>t)
            eb(m,2)=t;
        end
        c=1;
        m=r+1;
    else
        m=m+1;
    end,
end,
%check if the billet is given to another sailor but sailor has not moved to
%the billet yet.
for i=1:q
    if (billet_id==assi(i,3))
        c=1;
    end,
end,
if (c==0)
    eb=[eb; billet_id t];
end,
```

2. Add Sailor to Sailor without Billet File

Only one of these similar functions is described here. These functions add a sailor to the file "sailor_without_billet" if the sailor is not in included.

```
%function that adds a sailor to sailor_without_billet or change the
time the billet
%is needed
function[swob]=add_sailor_to_swob(swob, assi, sailor_id, t)
[r,p]=size(swob);
[q,s]=size(assi);
```

```
m=1;
c=0;
% check if sailor is already in the file
while (m<r+1)
    if (sailor_id==swob(m,1))
        if (t<swob(m,2))
            swob(m,2)=t;
        end
        c=1;
        m=r+1;
    else
        m=m+1;
    end,
end,
% check if sailor is already assigned
for i=1:q
    if (sailor_id==assi(i,1))
        c=1;
    end,
end,
if (c==0)
    swob=[swob; sailor_id t 0];
end,
```

3. Billet Identification Number

This small function records the billet ID of a billet the sailor uses at the time.

```
%function that responds the billet ID to a given sailor ID
function[bill_id]=billet_id(billets, sailor_id)
bill_id=0;
m=1;
while (m<554)
    if (sailor_id==billets(m,6))
        bill_id=billets(m,1);
        m=554;
    else
        m=m+1;
    end,
end,
```

4. Deleting a Sailor from the Files

Complete sailor information is saved in the sailor community file. If a sailor has to leave the experiment, this information has to be deleted. Because of the possibility that the sailor can look for a new billet at the time he has to leave, his ID has to be deleted from the "sailor_without_billet" file as well. The same applies to the assignment file. If a sailor is not completely deleted, he might use a billet forever and new sailors could not get this billet. The simulation program uses an own function to delete a sailor from each file. These functions are similar, only one (deleting a sailor from the “sailor_without_billet” file) is shown below:

```
%function that deletes a sailor from 'sailor_com'
function[swob]=del_sailor_from_swob(swob, sailor_id)
[r,p]=size(swob);
m=1;
while (m<r+1)
    if (sailor_id==swob(m,1))
        swob(m,:)=[];
        m=r+1;
    else
        m=m+1;
    end,
end,
```

5. Functions Looking for Sailor Information

The sailor identification number is not related to a position in the sailor community file. The sailor community file changes its size every month through attrition, growth and aging. It also changes the position of sailors within the file. Therefore, a tool is necessary to get sailor information out of the file for a given sailor identification number. The simulation uses an own function for all information. The information a function can look for are: TIS (Mos), Rank (Rank), NMI indicator(Nmi_i), Platform wish (Nmi_p), Homeport wish (Nmi_r) ,NMI type wish (Nmi_t) and Sea-Shore indicator (Sea_shore_ident). All these functions use a similar routine. The procedure for the rank is shown here:

```
%function that responds the rank from sailor_com for a
%given sailor_id
function[r]=rank(sailors, sailor_id)
r=0;
m=1;
while (m<554)
    if (sailor_id==sailors(m,1))
        r=sailors(m,3);
        m=554;
    else
        m=m+1;
    end,
end,
```

6. Sea-shore Date

This function calculates the next date for a sea-shore rotation in month of service (TIS).

```
%function that responds the next point in time for a sea shore rotation
function[date]=sea_shore_date(mos)
rot_date=[48;96;132;180;216;264];
m=1;
while (m<7)
    if (mos<rot_date(m))
        date=rot_date(m);
        m=7;
    else
        m=m+1;
    end,
end
```

7. Set Billet Free

If a sailor has to leave the system and is assigned to a new billet, but has not yet moved, this function writes the billet back to the file of empty billets.

```
%functions that put a billet that should be used in the future by a
leaving
%sailor back to empty_billets
function[eb]=set_future_billet_free(eb, assi, sailor_id)
[r,p]=size(assi);
m=1;
while (m<r+1)
    if (sailor_id==assi(m,1))
        eb=add_billet_to_eb(eb, assi, assi(m,3),assi(m,4));
        m=r+1;
    else
        m=m+1;
    end,
end,
```

LIST OF REFERENCES

Bureau of Navy Personnel, B. (2009, September). *Aviation Support Equipment Technician.* Retrieved October 25, 2009, from AS Community Overview: http://www.npc.navy.mil/Enlisted/CommunityManagers/Aviation/AS.htm

Department of the Navy, D. (2000, February). *Fiscal Year 2001, Budgets Estimates.* Retrieved March 05, 2010, from Military Personnel, Navy: http://www.finance.hq.navy.mil/FMB/01pres/PERS/MPN_Book.pdf

Department of the Navy, D. (2010, February). *Fiscal year 2011, Budgets Estimates.* Retrieved March 05, 2010, from Military Personnel, Navy: http://www.finance.hq.navy.mil/FMB/11pres/MPN_Book.pdf

Exams_tutor_business. (n.d.). *Study Room*. Retrieved March 12, 2010, from Motivation - 4. Herzbergs Two Factor Theory: http://www.examstutor.com/business/resources/studyroom/people_and_organisations/motivation_theory/4-herzbergstwofactortheory.php

GAO. (2002, November 25). *Military Personnel: Management and Oversight of Selective Reenlistment Bonus Program Needs Improvement.* Retrieved February 15, 2010, from GAO U.S. Government Accountability Office: http://www.gao.gov/new.item/d03149.pdf

Keller, G. (2005). *Statistics for Management and Economics, Ed. 7.* Belmont: Thompson South-Western.

Kohn, A. (1993). *Ultimate Rewards. A Harvard Business Review Book.* Boston: Harvard Business School Press.

Military.com. (2009, October). *Military Professional Development Center*. Retrieved October 15, 2009, from Navy Enlisted Advancement System: http://www.military.com/MilitaryCareers/Content/0,14556,Promotion_Navy_E5,00.html

Morben, D. M. (September 1989). *A Simulation Study of an Optimazation Model for Surface Nuclear Accession Planning.* Monterey, CA: Naval Postgraduate School.

Navy.mil. (2002, 7 10). *Navy announces High Year Tenure Policy Revision*. Retrieved November 9, 2009, from http://www.navy.mil/search/display.asp?story_id=2472

Nelson, B. (1999, February 1). *Inc.* Retrieved March 15, 2010, from Incentives for All Generations: http://www.inc.com/articles/1999/02/16431.html#

Schlegel, R. J. (2000). *An Activity Based Costing Analysis of the Department of the Navy's Enlisted Detailing Process.* Monterey: Naval Postgraduate School.

Stitt, R. D. (December 2009). *Identifying the Cost of Non-monetary Incentives (ICONIC).* Monterey, CA: Naval Postgraduate School.

Zimmermann, B. (December 2008). *Integrating Monetary and Non-Monentary Reenlistment Incentives Utilizing the Combinatorial Retention Auction Mechanism (CRAM).* Monterey: Naval Postgraduate School.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Dr. William R. Gates
   Naval Postgraduate School
   Monterey, California

4. Dr. Rachel T. Johnson
   Naval Postgraduate School
   Monterey, California

5. Andreas Buelow
   Rostock, Germany