RESOURCE ALLOCATION OPTIMIZATION PROBLEMS IN THE PUBLIC SECTOR

A Dissertation
Presented to
The Academic Faculty

by

Taylor J. Leonard

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in Industrial Engineering

Georgia Institute of Technology
May, 2020

RESOURCE ALLOCATION OPTIMIZATION PROBLEMS IN THE PUBLIC SECTOR

Approved by:

Dr. George Nemhauser
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Martin Savelsbergh
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. David Goldsman
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Alan Erera
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Jeffery Weir
Associate Dept Head of Dept of Operational Sciences
Air Force Institute of Technology

Dr. Andrew Armacost (BGen, Ret)
President
University of North Dakota

Date Approved: March 19, 2020

To my wonderful wife, Melissa Leonard

ACKNOWLEDGMENTS

I am extremely grateful to my advisor Dr. George Nemhauser for his time, his patience, his efforts, and his advice. Working with him has been an honor and a privilege, and without him, I would not have made it to the end. Second, I would like to express my gratitude to Dr. Eva Lee for being my advisor during my first three years at Georgia Tech. Dr. Lee is responsible for providing me interesting and critical problems to work on and for being my mentor during my time in school. I am thankful to them both for their guidance, support during my studies, dedication, and patience. Their continuous encouragement has been fundamental for my development as an independent researcher.

I would also like to thank the members of my thesis committee for their service: Dr. Jeffery Weir, Dr. Andy Armacost, Dr. Alan Erera, Dr. Martin Savelsbergh, and Dr. David Goldsman. I am very fortunate to have had such a talented group of experts in their fields, and I only wish that I had more time to work with each one of them.

Thank you to my friend, Alex Stroh, for being a constant source of support both inside and outside of class.

The biggest and most important acknowledgment goes to my wonderful wife, Melissa, who has been by my side throughout these difficult and unforgettable years. I am grateful for the unconditional support, her patience, her understanding, and for being an amazing mother. Thank you to my daughters, Paige, Haddyn, Henley, and Harriet for understanding that Dad was busy doing homework, studying, or doing research for so long. You always brought a sense of joy and happiness whenever I needed it.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This dissertation consists of three distinct, although conceptually related, public sector topics: the Transportation Security Agency (TSA), U.S. Customs and Border Patrol (CBP), and the Georgia Trauma Care Network Commission (GTCNC). The topics are unified in their mathematical modeling and mixed-integer programming solution strategies.

In Chapter 2, we discuss strategies for solving large-scale integer programs to include column generation and the known heuristic of particle swarm optimization (PSO). In order to solve problems with an exponential number of decision variables, we employ Dantzig-Wolfe decomposition to take advantage of the special subproblem structures encountered in resource allocation problems. In each of the resource allocation problems presented, we concentrate on selecting an optimal portfolio of improvement measures. In most cases, the number of potential portfolios of investment is too large to be expressed explicitly or stored on a computer. We use column generation to effectively solve these problems to optimality, but are hindered by the solution time and large CPU requirement. We explore utilizing multi-swarm particle swarm optimization to solve the decomposition heuristically. We also explore integrating multi-swarm PSO into the column generation framework to solve the pricing problem for entering columns of negative reduced cost.

In Chapter 3, we present a TSA problem to allocate security measures across all federally funded airports nationwide. This project establishes a quantitative construct for enterprise risk assessment and optimal resource allocation to achieve the best aviation security. We first analyze and model the various aviation transportation risks and establish

their interdependencies. The mixed-integer program determines how best to invest any additional security measures for the best overall risk protection and return on investment. Our analysis involves cascading and inter-dependency modeling of the multi-tier risk taxonomy and overlaying security measurements. The model selects optimal security measure allocations for each airport with the objectives to minimize the probability of false clears, maximize the probability of threat detection, and maximize the risk posture (ability to mitigate risks) in aviation security. The risk assessment and optimal resource allocation construct are generalizable and are applied to the CBP problem.

In Chapter 4, we optimize security measure investments to achieve the most cost-effective deterrence and detection capabilities for the CBP. A large-scale resource allocation integer program was successfully modeled that rapidly returns good Pareto optimal results. The model incorporates the utility of each measure, the probability of success, along with multiple objectives. To the best of our knowledge, our work presents the first mathematical model that optimizes security strategies for the CBP and is the first to introduce a utility factor to emphasize deterrence and detection impact. The model accommodates different resources, constraints, and various types of objectives.

In Chapter 5, we analyze the emergency trauma network problem first by simulation. The simulation offers a framework of resource allocation for trauma systems and possible ways to evaluate the impact of the investments on the overall performance of the trauma system. The simulation works as an effective proof of concept to demonstrate that improvements to patient well-being can be measured and that alternative solutions can be analyzed. We then explore three different formulations to model the Emergency Trauma Network as a mixed-integer programming model. The first model is a Multi-Region, Multi-

Depot, Multi-Trip Vehicle Routing Problem with Time Windows. This is a known expansion of the vehicle routing problem that has been extended to model the Georgia trauma network. We then adapt an Ambulance Routing Problem (ARP) to the previously mentioned VRP. There are no known ARPs of this magnitude/extension of a VRP. One of the primary differences is many ARPs are constructed for disaster scenarios versus day-to-day emergency trauma operations. The new ARP also implements more constraints based on trauma level limitations for patients and hospitals. Lastly, the Resource Allocation ARP is constructed to reflect the investment decisions presented in the simulation.

CHAPTER 1. INTRODUCTION

The public sector is comprised of a variety of governmental services, including law enforcement, infrastructure, security, and health care. These services provide functions such as airport security, border security, and emergency trauma care, which are responsible for protecting millions of people on a day-to-day basis. Within an industry of this size and critical importance, the need for efficient optimization solutions is high. Practical solutions can result in significant modifications to airport security and border security as well as increase response times in emergency injury incidences. Due to the significant operational scale, complex rules, and various parameters, the resulting optimization problems are complicated and time consuming to solve.

1.1 Background

In each of the resource allocation problems presented, we concentrate on selecting an optimal portfolio of improvement measures. The number of portfolios of improvements that are formed is too large to be expressed explicitly or stored on a computer. Therefore, we apply multiple solution strategies based on column generation and particle swarm optimization (PSO).

Integer programming is used to model and solve a variety of complex problems to include routing, planning, investing, and scheduling. One distinguishing characteristic of integer programming is that the decisions to be made are quantified by integer values, such as ’include this asset in your investment portfolio (1) or not (0)’. The problem is a mixed-integer program (MIP) if only some of the variables are required to integral. Applying integer programming to real-world applications can be said to involve two phases; one – the modeling phase – that consists of the interaction with the owners of an optimization problem to be solved, and the other –the solution phase – that relies on computer software to solve the problem stated in the modeling phase.

In the modeling phase, all aspects to be considered to solve the problem must be defined and quantified to create a mathematical model describing the problem. When faced with complex real-world problems, it can be challenging to construct a solvable model that mirrors reality and whose solutions are of practical interest. It is critical to determine how to provide the model with input data of good enough quality for the solution to be sound while balancing the trade-off between building a solvable model and including as much realistic detail as possible.

The solution phase aims to provide a solution to the problem described by the mathematical model constructed in the modeling phase. This is typically done by the appropriate use of standard optimization software. When even software is not capable of effectively solving the problems, new solutions methods are developed, or existing methods are tailored.

After finding a solution or several to the problem, they are presented to the decision-makers for feedback on the quality of the solutions. It is natural that feedback will result in changes in the model and can result in changes to the solution method, establishing an interdependency between the two. After finishing this iterative process, the resulting model and solution method can be put to use, either as a decision-support system or as a tool that repeatedly and automatically provides and uses solutions to problem instances generated in real-time.

The level of effort involved to solve a problem can range from combining well-known solution strategies in a new scenario or application to applying an idea that creates an entirely new area of research. In integer programming research, both the modeling and solution phases are challenging in their own right; therefore, progress needs to be made in both areas.

In column generation, the columns correspond to which portfolio of resources to be allocated. For problems of this size, models often have an exponential number of variables. The main advantage of column generation is that not all portfolio possibilities need to be enumerated upfront, but instead can be generated on the fly. The original problem is first continuously relaxed and then divided into a master problem, and a subproblem called the pricing problem. The master problem, when defined on a subset of all possible portfolios, is called the restricted master problem. The optimal set of portfolios is found in an iterative process by alternating solving the restricted master problem and the pricing problem.

In the pricing problem, new portfolios are generated and then added to the restricted master problem. The pricing problem can be solved for one airport/sector/region at a time, and hence several different and separate pricing problems are typically solved in each column generation iteration. Since the quality of the solution highly depends on which portfolios are generated, the pricing problem is an essential step of the column generation algorithm. The objective of the pricing problem is to generate portfolios that are cheap to assign to and that are likely to improve the solution to the restricted master problem. The portfolios found in the pricing problem must be legal in the sense that they fulfill all scenario-specific rules, and no illegal portfolios are allowed to enter the restricted master problem.

Due to the mentioned properties, heuristics are often used to solve the pricing problem, and therefore, it is interesting to investigate alternative methods to discover if the performance can be enhanced. The aim is to find, implement, and compare alternative methods for solving the pricing problem. The new methods should be evaluated within the existing column generation framework.

Relating the above description to the contributions of this thesis, Chapter 3 describes the application of integer programming for the Transportation Security Administration (TSA) security measure allocation problem and involves both the modeling and the solution phases. Chapter 4 describes the application of integer programming to Customs and Border Patrol (CBP) security measure allocation problem and involves both the modeling and solution phases. Chapter 5 describes the application of simulation and integer programming to emergency trauma care, respectively, and include both the modeling and solution phases. The goal of this research is to contribute to the development of modeling techniques and solution strategies that can be applied to common problem structures in large scale integer programming.

1.2 Limitations

This dissertation is focused on solving critical, challenging, real-world problems, and comparing solutions techniques. The methods implemented in this thesis are compared with each other. For the methods to be evaluated within the column generation framework, some consideration to computational time must be taken into account, and therefore, for computational reasons, some solution methods might not be appropriate to implement. Apart from that, no specific run time limitations apply. The methods should work for any given problem instance, i.e., it should be possible to apply the techniques using data regardless of the application.

1.3 Dissertation Outline

Chapter 1 gives a short introduction to the application of integer programming to resource allocation problems. The purpose of this chapter is to provide a background to the areas of application that are the focus of Chapters 3, 4, and 5, which themselves give a short introduction to the integer programming modeling of each scenario. The papers discussed are not presented in their journal format, but rather in an extended and more cohesive format suited to the dissertation.

Chapter 2 introduces the fundamentals of integer programming column generation and Particle Swarm Optimization (PSO). It then discusses the solution methodologies for each public sector problem, first Dantzig-Wolfe decomposition, and then the exact solution technique of column generation and the PSO heuristic.

Chapter 3 covers all the details of prior TSA work to include an in-depth literature review of optimization models for the TSA (Leonard et al., 2019). It is also in Chapter 3 where we discuss Enterprise Risk Management (ERM) and any relative quantitative techniques and how best to incorporate ERM into an OR problem. Chapter 4 provides a brief literature review for optimization type models for the CBP, (Leonard & Lee, 2020).

Chapter 5 reviews of all the literature regarding the development of models for emergency trauma networks (Lee et al., 2020) and (Leonard & Lee, 2020). This development includes trauma network simulation, multi-region, multi-depot, multi-vehicle, multi-trip, vehicle routing problems, ambulance routing problems, and finally, a combination portfolio optimization ambulance routing problem. This also includes simulation type constructs to approach the emergency trauma network problem. In chapters 3, 4, and 5 we define a mixed-integer resource allocation problem and solve using both multi-swarm PSO and column generation with and an embedded multi-swarm PSO pricing problem. And lastly, Chapter 6 provides summaries, conclusions, and areas of future research.

1.4 Contributions

1.4.1 Transportation Security Agency

This project aims to establish a quantitative construct for enterprise risk assessment and optimal portfolio investment to achieve the best aviation security. We first analyze and model the various aviation transportation risks and establish their interdependencies. Using the security measures and their capabilities, we formulate the multi-objective portfolio investment model via a MIP framework. The portfolio risk model determines the best capabilities of the current budget and can also pinpoint potential capabilities when changes in budget occur. The computational framework allows for marginal cost analysis, which determines how best to invest any additional resources for the best overall risk protection and return on investment. Our analysis involves cascading and inter-dependency modeling of the multi-tier risk taxonomy and overlaying security measurements. The model selects optimal security measure portfolios (which type and how many) to distribute across airports nationwide with the objectives to minimize the probability of false clears, maximize the probability of threat detection, and maximize the risk posture (ability to mitigate risks) in aviation security. This study presents the first comprehensive model that links all resources across the 440 federally funded airports in the United States. We experiment with several computational strategies, including Dantzig-Wolfe decomposition, column generation, multi-swarm particle swarm optimization, and a greedy heuristic. We present results to contrast the current baseline performance versus some of the near-optimal solutions obtained by our system. Our results demonstrate higher risk posture, lower false clear, and higher threat detection across all the airports, indicating a better risk enterprise strategy and decisions obtained from our system. The risk assessment and optimal portfolio investment construct is generalizable and can be readily applied to other risk and security problems.

The risk evaluation methodology is new and approached differently. It was developed to build a quantitative framework for non-quantitative risk topics. We now have a consistent analysis framework that is easily adaptable for other ERM type scenarios. To do this, we reviewed all of the current TSA enterprise risks, tracked their associated risk appetites, and defined the interdependency relationships between all of the risk factors. Since we do not have reliable estimates for the correlation or historical data to derive them from, we adapted network analysis techniques to make a pseudo correlation matrix. We implemented a topological overlap matrix that is typically used for biological networks and computer networking applications, and, for the first time, is being integrated into an optimization model. Lastly, an adaptable risk posture metric was developed as the primary objective and incorporated to assess the risk factors and security measures.

1.4.2 Customs and Border Patrol

A large-scale resource allocation integer program was successfully constructed that quickly runs to optimality and provides results. This CBP model was directly influenced by the current state-of-the-art TSA security screening research that we have designed in (Leonard et al., 2019). The overall model continues to be very flexible and can comfortably accommodate different resources, new constraints, and additional objectives. The solution methodologies that are being put in place are complex, current, and effective. They will allow further development of a mathematically supported decision analysis computational tool for the CBP to give more justification for their capability gaps and develop smart investments.

With a strong model foundation in place, this formulation is very flexible and can comfortably accommodate additional and/or different objectives and constraints. We acknowledge our model estimates the following input: false alarm detection rate for surveillance devices; list of new and potential technologies to be considered; different measures of performance that can be included; Accurate list of current methods that are employed and their locations.

A substantial research gap is due to existing ERM optimization models only perform at an operational level and not at a strategic level. As far as we know, the TSA model is the strategic ERM model of its kind and results in close to ½ billion decision variables. The

CBP model is more manageable with 13,888 integer variables (448 of those are binary) and is capable of covering a full multi-tier ERM framework (strategic, tactical, and operational levels). To the best of our knowledge, this is the first model to mathematically determine security strategies for the Customs and Border Patrol, as well as to introduce a utility factor to emphasize deterrence/detection impact. The model continues to be very flexible and can easily accommodate different resources, new constraints, and additional objectives.

This dissertation offers an application to the large-scale system we developed for TSA risk analysis and determines an optimal solution methodology for solving the security measure resource allocation model across multiple border sectors. Under physical/cyber/resource/logistics constraints, this model optimizes the allocation of limited quantities of deterrence and detection security measures across the entire southern continental U.S. border to maximize the total utility of the measures utilized, maximize the probability of deterrence and/or detection, and minimize cost. A utility factor is introduced to rate the impact of a security measure. Dantzig-Wolfe decomposition is used to solve the nonlinear MIP problem instances, where optimal solutions are shown to be obtained in several seconds through several computational examples. Working with CBP, there is an opportunity to integrate a multi-tier risk taxonomy framework, (Lee et al., 2019), e.g., incorporating migrants, cargos, materials, etc. and their risk interdependencies within the resource allocation framework problem to structure a risk-based screening strategy that makes effective use of limited screening resources. We acknowledge that our application only addresses operational and logistics challenges, and complicated human factors remain to be investigated.

1.4.3 Emergency Trauma Networks

The emergency trauma network was first analyzed by simulation to handle the size of the problem. The results are subject to change for a different trauma system, with different parameters, cost structures, and submissions. However, the most significant contribution of this study is that it offers a framework of investment allocation for trauma systems and possible ways to evaluate the impact of the investments on the overall performance of the trauma system. It is a top-down approach on a strategic level, but it uses the tactical level decisions to evaluate several strategies to improve the system. Simulation is a powerful tool to perform a thorough analysis and systematic update of the system with given investments and facilitates the decision-making process of decision-makers in the trauma network. This simulation was intended as a proof of concept to demonstrate that improvements to patient well-being can be measured in some manner and that alternative solutions can be analyzed.

The problem then went through three different modeling formulations to reach a mixed-integer formulation that resembles the simulation but can provide more accurate results. The first formulation state was developing the Multi-Region, Multi-Depot, Multi-Trip Vehicle Routing Problem with Time Windows. This is a known variation of the vehicle routing problem but was only portrayed on a smaller scale with two regions and was further expanded for the Georgia trauma network. The Ambulance Routing Problem (ARP) was adapted from the above mentioned VRP. There are no known ARPs of this magnitude/extension of a VRP. One of the primary differences is many ARPs are constructed for disaster scenarios versus day-to-day emergency trauma operations. The new ARP also implements more constraints based on trauma level classifications for patients and hospitals. Lastly, the resource allocation ARP was constructed to reflect the decision capabilities from the simulation. This model final allows us to compare the results with the original trauma simulation. There no known models of this type, combining portfolio optimization with VRP.

1.4.4 Solution Techniques

All models presented in this dissertation were evaluated in the same manner due to having similar foundations. The solution techniques were chosen specifically because of their characteristics. They all have multiple objectives that represent different goals of the decision-makers. Additionally, they all contain some type of resource allocation or portfolio optimization construct.

The standard method of column generation was utilized for exact solutions if problems were too large to be solved with a standard solver. This was necessary for nearly all the problems, with the exception of the CBP problem. Even though column generation can be applied, it is still computationally intensive and very time-consuming. Although it is helpful to model difficult problems, in any case, it's not very helpful if the problems cannot be solved effectively. In order to generate strong solutions efficiently, PSO was first applied.

The PSO for problem set was implemented for discrete optimization and required extra consideration for the binary and integer variable boundaries. Many modifications and/or potential improvements for PSOs have been studied over the years. In our PSO format, we incorporated the velocity controlled hybrid PSO algorithm (VC-HPSO) (Yaakob & Watada, 2010). VC-HPSO is a modified standard PSO with the addition of the mutation operator of Genetic Algorithms and controlled velocity adjustment. PSOs typically perform well in early iterations but have issues reaching near-optimum solutions. A standard PSO for our set of large-scale problems was only moderately effective due to not being able to reach near-optimal solutions and was also computationally intensive due to the size of the data.

Multi-swarm optimization is derived from standard PSOs but uses multiple sub-swarms rather than the standard single swarm. Multi-Swarm PSO (MSPSO) were then investigated and applied. The purpose of MSPSO is to divide the population of potential into subswarms. Each subswarm utilizes a different global best for particle movements. After each subswarm is run for a set number of iterations, the global best with a better value is copied for all of the swarms. This helps broaden the search space and improve convergence towards better solutions.

Lastly, given a traditional exact methodology and an efficient heuristic method, the two were combined into one solution method. We implemented the MSPSO inside column generation to solve the pricing problem. The pricing problem is often the most time-consuming element of the column generation procedures due to needing to solve a large number of subproblems. In this work, we use the MSPSO to solve the pricing problem within the column generation framework. The MSPSO is a much more efficient method to select potential columns or portfolios to enter the potential portfolio solution subset. As mentioned previously, the following chapter will cover the literature review of all of the necessary background materials.

CHAPTER 2. SOLUTION METHODOLOGY

This chapter introduces the fundamentals of mixed-integer programming column generation and PSO. The problems explored in this dissertation are prohibitively large in terms of their data requirements and model formulation and require reformulation prior to solving. After reformulating the problem, we found success in solving large instances with traditional column generation and PSO. The downside to both methods is solution time. Due to the number of subproblems in the pricing problem, column generation takes an excessive amount of time. Due to the number of variables in the problem, PSO also takes an excessive amount of time to complete.

In order to make solving extremely large problems more accessible, we implemented a multi-column generation and multi-swarm PSO. Neither of these methods is new at this time, but uniting the two methods into a multi-column generation with a multi-swarm PSO is. In the following sections, we will cover the full background of the solution techniques implemented.

2.1 Dantzig-Wolfe Decomposition

A decomposition was necessary to linearize the risk structures or nonlinear portions of the model. A Dantzig-Wolfe decomposition is applied and discussed further in the following section. The Dantzig-Wolfe decomposition makes sense based on the problem structure. It allows division of the optimization problem into two groups of "easy" and "hard" constraints. The hard constraints are not necessarily difficult, but they complicate the LP by making it nonlinear and more difficult to solve. When these hard constraints are removed from the problem, then more efficient techniques can be applied to solve the remaining linear program or, in this case, IP.

Mathematical programs that contain a large space of integer variables are particularly suited for Dantzig-Wolfe decomposition that reformulates the original compact problem to provide a tighter linear programming relaxation bound. This decomposition relies on a delayed column generation algorithm. Many programs are too large to consider all the variables explicitly, and most of the variables will be neglected in the optimal solution, so the algorithm only considers a subset of variables when solving the problem. Column generation only generates the variables with negative reduced costs that have the potential to improve the objective function. The primary use of Dantzig-Wolfe decomposition here is to reformulate the portfolio of selected resources (security measures or emergency resources) and employing the column generation algorithm or PSO.

The Dantzig-Wolfe reformulation results in a master problem and subproblems, whose typically large number of variables are dealt with implicitly by using an integer programming column generation procedure, known as a branch-and-price algorithm. Solving the master problem does not require an explicit enumeration of all its columns because the column generation algorithm allows one to generate columns if/when needed. Often, this allows one to solve huge integer programs that were previously considered intractable. The combination of the Dantzig-Wolfe reformulation and column generation solution strategy has been successfully applied in many classical problems. (Belaid & Eyraud-Dubois, 2015)

Dantzig-Wolfe decomposition is a valuable tool to solve large structured models that cannot be solved using standard algorithms due to the limited capacity of solvers and the underlying CPUs. The main idea behind the technique is to decompose the original problem into several independent subproblems, whose solutions are then assembled by solving a so-called restricted master problem. The restricted master problem is then solved iteratively. In our scenarios, we can identify the natural decomposition of the problem: for different values of k (i.e., for each airport/border sectors/emergency regions), the corresponding sets of constraints are independent, because they contain disjoint sets of variables. If we were able to assume that all k (airports/border sectors/emergency regions) are homogeneous, all those subproblems would be identical, and we would have a particular case where solving the subproblem once is enough. However, this is not the case, which is part of what makes these problems extremely large.

In the Dantzig-Wolfe reformulation, the final result is a master problem which contains one variable for each solution to the original formulation. In our scenarios, such a solution represents a valid security measure portfolio to be installed at the respective airports, or a valid security measure portfolio to be installed at the respective border sectors, or a portfolio of upgrades to be made within the emergency trauma regions.

2.2 Column Generation

After completing the decomposition, a result is a massive number of decision variables with a very small number of constraints. The main advantage of using column generation on a problem with this many variables is that only a relatively small subset of the decision variables are included in the model, and the rest are taken into account implicitly.

Assume that the following problem, with *m* constraints and *n* decision variables, is to be solved by the simplex method for linear programs. This problem is the master problem (MP) seen below in equations (2.1) - (2.3),

(MP)

$$z^* = \ min \quad \sum_{j \epsilon N} c_j x_j \tag{2.1}$$

$$\text{s.t.} \quad \sum_{j \epsilon N} a_j x_j \geq b \tag{2.2}$$

$$x_j \geq 0, j \epsilon N \tag{2.3}$$

Let x_j, $j \epsilon N$, be a non-negative, continuous decision variable and $c \, \epsilon \mathbb{R}^n$, $x_j \, \epsilon \mathbb{R}^n$, $a_j \, \epsilon \mathbb{R}^{\text{n}}$, and $b \, \epsilon \mathbb{R}^m$. The vector, *u*, represents the dual variables associated with the linear constraints. The pricing step of a simplex iteration performs the following evaluation (equation (2.4)) on a non-basic variable x_j' such that

$$j' = arg \ min\{\bar{c}_j = c_j - u^T a_j : \ j \in N\} \tag{2.4}$$

is chosen to enter the basis if $\bar{c}_{j'} < 0$. If $\bar{c}_{j'} \geq 0$, an optimal solution has been found, and the algorithm terminates.

In a column generation application, the set of columns *N* is assumed to be very large such that it is practically challenging or not possible to store all columns explicitly. Instead, only a subset of the columns $\bar{N} \subseteq N$ is initially used, forming a restricted master problem (RMP), which will be assumed to contain a feasible solution in this chapter.

Let $\bar{\lambda}$ and $\bar{u}$ be an optimal primal solution and the corresponding complementary dual solution, respectively, to *RMP*. In the pricing step, a subproblem is solved over set P, which uses constraints to describe the feasible columns implicitly instead of an explicitly searching among variables $j \in N \backslash \bar{N}$. The cost of the column is represented by $c_j, j \in N$. The objective of the subproblem (SP) is to find

$$(SP) \qquad \bar{c}^* = min\{c_j - u^T a_j : a_j \in P, j \in N\} \qquad (2.5)$$

The problem formulation using *MP* and *SP* introduced above in (2.5), is called an extensive formulation of a problem.

Depending on the structure of the coefficient matrix and the cost coefficients, different techniques can be used to solve the subproblem. The only requirement for a column to enter the RMP is that it should have a negative reduced cost. Therefore, it is not necessary to solve the pricing problem to optimality as long as one column with a negative reduced cost can be selected to enter the RMP. A potential downside to selecting the first column to enter the RMP with a negative reduced cost is the possibility of missing a "better" column with a more negative reduced cost.

Assuming the solution method for solving the pricing problem finds the column with minimal reduced cost, then when no column with a negative reduced cost can be found, the optimal solution to the original problem has been found.

Ronnberg (Ronnberg, 2012) provided a thorough, yet brief summary on the little literature available on branch-and-price alternatives for obtaining integer solutions in a column generation setting. The original reference is (Barnhart et al., 1998) and for additional references, see (Baldacci et al., 2006), (Baldacci, et al., 2008), and (Sweeney &

Murphy, 1979). (Barnhart et al., 1998) and (Lübbecke & Desrosiers, 2005) provide an excellent survey of integer programming and column generation, and are essential to building the foundation. Wilhelm (Wilhelm, 2001) provides another excellent survey that includes several detailed examples.

Ronnberg (Ronnberg, 2012) also identified the following papers as the foundation of column generation as an IP solution technique. Ford and Fulkerson (Ford Jr & Fulkerson, 1958) first suggested that the variables of a multicommodity flow problem should be dealt with only implicitly. Dantzig and Wolfe were able to develop this initial column generation strategy when they introduced Dantzig-Wolfe decomposition (Dantzig & Wolfe, 1960). Gilmore and Gomory presented the first complete implementation of a column generation strategy in (Gilmore & Gomory, 1961) and (Gilmore & Gomory, 1963). Appelgren then discussed the challenges of combining column generation and linear-programming-based branch-and-bound in (Appelgren, 1969).

2.3 Discrete Particle Swarm Optimization

In the past 25 years, the PSO algorithm has established itself as a very efficient global optimizer. PSO was first introduced in (Eberhart & Kennedy, 1995) as a population-based technique for solving continuous optimization problems. It was inspired by the swarming behavior found in bird flocks and fish schools. The problem is iteratively solved by moving an improving a set of candidate solutions (particles) towards previously known good solutions. Each particle has an associated position and velocity within the search space. In each iteration, a linear combination of each particles' previously known local best position and the global best position of the swarm steers the next particle value.

Kennedy and Eberhart (Kennedy & Eberhart, 1997) proposed binary (discrete) PSO (BPSO) to solve binary optimization problems. The particle velocity is passed through a continuous transfer function taking values in the interval [0,1] or, in the case of discrete integer values greater than one, in the interval of [0, upper bound]. The output of the function is the probability of the particle position in the next iteration.

The general BPSO algorithm is presented in a very succinct manner in (Curry, 2018) and is described below. Let N be the number of particles in the swarm, and d represents the dimension/s of the decision variable. The position and the velocity of the particle are then $p_i = [p_1^i, p_2^i, \dots, p_d^i]$ and $v_i = [v_1^i, v_2^i, \dots, v_d^i]$ respectively with $i = 1, 2, \dots, N$. The algorithm begins by initializing the position and velocity of each particle and evaluating the objective function, f, using the position of each particle, $f(p_i)\ \forall\ i = 1, 2, \dots, N$. The best position found by each particle i and the best position found by the entire swarm are stored as p_{pBest}^i and p_{gBest} respectively. The functions (2.6) and (2.7) take the particles' current position, p^i, and update p_{pBest}^i and p_{gBest} in each iteration.

$$\text{if } f(p_i) < f(p_{pBest}^i), \text{ then } p_{pBest}^i \leftarrow p^i \tag{2.6}$$

$$\text{if } f(p_i) < f(p_{gBest}), \text{ then } p_{gBest} \leftarrow p^i \tag{2.7}$$

Using the best positions found so far, the velocities and positions for each particle i = 1, 2, . . . , N are updated using equations (2.8) and (2.9) below. The cognitive component and the social component are the constants c_1 and c_2 and r_1 and r_2 are random numbers in the interval [0,1]. The inertia weight, w, represents the trade-off between exploration and exploitation.

$$v_i^{t+1} \leftarrow w v_i^t + c_1 r_1 (p_{pBest,t}^i - p_i^t) + c_2 r_2 (p_{gBest,t} - p_i^t), \quad t=1,2,\dots,d \tag{2.8}$$

$$p_i^{t+1} \leftarrow p_i^t + v_i^{t+1} \tag{2.9}$$

If $w > 1$, exploration is favored and if $w < 1$ exploitation is preferred by drawing the particles towards the current best position. For our purposes, we maintain a $w > 1$ to encourage exploration of a very large search space. Selecting an appropriate value of w, along with common values, is presented in (Wahde, 2008). A different approach encourages exploration at the beginning of the algorithm to provide a broader global search and then reducing w towards the end to exploit previously known good areas. The adjustment is made by reducing w by a constant factor in each iteration until it reaches a lower bound. In future research, it might be beneficial to adjust w to see if the optimality gap can be reduced.

Although the velocities are calculated with each iteration, they are bound by the interval $[-v_{max}, v_{max}]$, and depending on the sign; they are set to the upper or lower bound if they fall outside the interval. Generally, the particle is more likely to move towards a previously known good solution, but by using a transfer function, the particle, with some probability, will move in another direction. The velocities passed through the transfer function, explained in the next section in equations (2.13) and (2.14), are used to update the particle positions. The new position is then used to update the velocity once again, according to the update rule in equation (2.8). The velocity is also used to determine new probabilities by using the transfer function. The sequence of calculating probabilities, updating positions, and updating velocities is repeated until the maximum number of iterations has been performed, or some other stopping criterion is met, such as no improvements in the best position after some proportion of iterations has been performed.

This is a relatively simplistic metaheuristic that applies to many problem types and can search very large spaces of candidate solutions. Unfortunately, it does not guarantee an optimal solution is found, but the goal is to see relatively quick convergence of the particles towards a solution. In multi-objective problems, Pareto dominance is taken into account when moving the particles and non-dominated solutions are stored as to approximate the Pareto front.

The original design (Eberhart & Kennedy, 1995) and (Kennedy & Eberhart, 1997) have been regularly modified (Shi & Eberhart, 1998) and (Engelbrecht, 2010) and analyzed extensively in (Van den Bergh & Engelbrecht, 2006). For our work, the velocity controlled hybrid PSO (VC-HPSO) algorithm was included in our overall PSO framework (Yaakob & Watada, 2010). This update was specifically included due to Yakoob's application of the PSO for the portfolio selection problem, which is essentially what the problems in this dissertation result in after decomposition. Yakoob incorporated the mutation operator to encourage PSOs to converge on local points. Without the appropriate operator, if a particle's current position is similar to the global best and its inertial weight and previous velocity are not equal to zero, the particle will actually move away from the global best. If the previous velocities are very close to zero, then all the particles will stop moving around the current best solution, possibly leading to premature convergence and stopping the algorithm short of completion. If all of the particles converge to only the best solution thus far (not the optimal), the algorithm has reached stagnation. A mutation operator helps the algorithm avoid stagnation by introducing new genetic material in the existing individual. The mutation occurs with a probability p_m, the mutation rate. We utilize a small $p_m \in [0,1]$

to avoid overly distorting reasonable solutions. Using a large p_m initially ensures that ample search space is covered, while p_m rapidly decreases when individual particles begin converging to the optimum, seen in equation (2.10) below.

$$p'_m = p_{max} - \frac{p_{max} - p_{min}}{G_{max}} \times t \quad (2.10)$$

The multi-swarm is one of the most popular approaches for PSO modification (Liang & Suganthan, 2005), (Ostadrahimi et al., 2012), and (Solomon et al., 2011).

2.3.1 Multi-Swarm PSO

The multi-swarm PSO (MS-PSO) is based on the local version of PSO with a new neighborhood topology. One of the important differences is that many existing evolutionary algorithms require larger populations, while PSO needs a comparatively smaller population size. A PSO with a population of three to five particles can achieve satisfactory results for simple problems (Zhao et al., 2008). According to many reported results on the local version of PSO (Kennedy, 1999) (Broyden, 1970), small neighborhoods perform better on complex problems. Therefore, the MS-PSO using small neighborhoods can slow down convergence speed and increase diversity to achieve better results on multimodal problems.

In the multi-swarm approaches, the population is divided into multiple sub-populations (sub-swarms) with different levels of communication. This allows the population to maintain divergence, search various promising regions, and partially converge into multiple optima. In (García-Nieto and Alba, 2012), the optimal swarm (subswarm) size is discussed in great detail. Pluhacek proposed that six particles per swarm might be the optimal number for PSO based algorithms (Pluhacek 2016).

It was demonstrated in (Pluhacek, 2016) that the multi-swarm performance was superior to the single swarm PSO in all cases. Based on the comparative study of single swarm PSO versus multi-swarm PSO performed in (Pluhacek, 2016), we decided to utilize multi-swarm PSO, with five sub-swarms, with varying particle sizes from 5 to 10 particles per swarm. Out standard control parameters were set as follows:

- Population Size: 5 to 10 particles (solutions) in each population
- Iterations: 10
- $v_{initial}$: 10% of position (new velocity of the ith particle in iteration t+1)
- w_{max}: 0.9 (maximum inertia weight value)
- w_{min}: 0.4 (minimum inertia weight value)
- c_1, c_2 = 1.49445 (learning factors or acceleration constants)
- pBest – Local (personal) best solution found by the ith particle
- gBest – Best solution found in a population
- x, z = current positions of the ith particle
- rand = Pseudo-random number, interval (0,1)

The searching within a subswarm is a repeated with stop criteria occurring when the maximum number of iterations has been reached, or the minimum error condition is satisfied. The process continues across all subswarms, where the global best is now recorded amongst all of the subswarms and compared to one another. An advantage of PSO is not many parameters require tuning. The dimension of the particles (dimension of solution set) is prohibitively large in this case, require keeping the number of particles to a minimum size. The upper and lower bounds of the decision variables determine the range of the particles. v_{max} determines the maximum change one particle can take during one iteration. We require two different velocities to track along with our binary and integer variables.

The multi-swarm optimization algorithm works as follows in Algorithm 1:

Algorithm 1: *MS-PSO()*
Input: MP (1)
Swarm_size: number of the swarm particles
No_subswarms: number of subswarms
Step 1: Calculate Subswarm size= Swarm_size/No_subswarms
Step 2: *For* subswarm = 1 to No_subswarms do
For t=1 to Max_iterations do
Apply PSO algorithm as in Eqs. (2.8) and (2.9)
Update p^{i}_{pBest}
Update p_{gBest}
End For
Return final result in p_{gBest}
Append the result to the results list
End For
Select best global subswarm result

2.4 Solving the Pricing Problem Using MS-PSO

In Section 2.2, we reiterate that the pricing problem consists of finding columns with negative reduced costs. For our problems, a column corresponds to a portfolio of security measures/trauma investments, and the pricing problem is solved separately for each airport/border sector/region, $k \in K$.

During each MSPSO iteration, each particle has its own position vector that is comprised of a set of columns p_j , $j \in J_k$, where J_k is the set of all column indices for each airport/border sector/region, $k \in K$. The position of the particle is determined by the unique set of investments that are selected for each *k*. Therefore, if the position changes in the next iteration, the particle is associated with a new set of potential columns. The number of investment measures to be installed for the respective installation site, k, are determined within each iteration and are different for all particles and represented by z_{dkj}. Therefore, each variable requires its own particle position, denoted by $p_i \in \{0, 1\}^{|T|}$ and its velocity by $v_i \in \mathrm{R}^{|T|}$ with $i = 1, 2, \ldots, N$.

We begin by randomly generating the initial N particle positions from the set of the potential investment portfolios for each k, $j \in J_k$. The algorithm is considered to be a type of local search (Pirlot, 1996) due to new columns being influenced by the best-known set of columns. The velocity vector for each particle is initialized within the predefined lower and upper bounds, $[-v_{max}, v_{max}]$, and the algorithm continues to search for new columns until the stop criteria is reached.

Each column is evaluated within our column generation code, and the reduced cost for each particle is computed for each iteration. A column is saved to the list of entering columns to the RMP if it has a negative reduced cost. The reduced costs of the previously known best positions are compared to the reduced cost of each particle's current position to potentially update both p^i_{pBest} and p_{gBest}. Then, as previously mentioned, the velocities are updated according to equations (2.11) and (2.12) below.

$$\text{if } v^i_t > v_{max}, \text{then } v^i_t \leftarrow v_{max} \tag{2.11}$$

$$\text{if } v^i_t < -v_{max}, \text{then } v^i_t \leftarrow -v_{max} \tag{2.12}$$

Lastly, the particle positions are updated by applying a transfer function. Transfer functions are selected such that each particle is encouraged to stay in its current position unless the absolute value of the velocity is high. For our work, we incorporate the transfer function first presented in the original BPSO (Kennedy & Eberhart, 1997), below in (2.13). The function in (2.13) is a sigmoid limiting transformation, and the position in the next iteration is updated according to (2.14) where *r* is a random number in the interval [0,1].

$$\sigma(v_t^i) = \frac{1}{1 + e^{-v_t^i}} \tag{2.13}$$

$$p_t^i \leftarrow \begin{cases} 1 & if\ r < \sigma(v_t^i), \\ 0 & otherwise, \end{cases} \tag{2.14}$$

The complete PSO for generating new columns with MSPSO is presented in Algorithm 2.

Algorithm 2: *CG Pricing with MS-PSO()*
Result: Columns with negative reduced cost, that represent potential portfolio options are added to the set of feasible portfolios

Input: MP (1)
Swarm_size: number of the swarm particles
No_subswarms: number of subswarms
Step 1: Calculate Subswarm size= Swarm_size/No_subswarms
Step 2: Initialize all particle positions
Step 3: *For* subswarm = 1 to No_subswarms *do*
While t=1 < Max_iterations *do*
For particles *i do*
Apply PSO algorithm as in Eqs. (2.8) and (2.9)
if $cost^i < 0$ *then*
NewColumns ← saveColumnToRMP(p^i)
End if
Update p_{pBest}^i
Update p_{gBest}
End For
End While
Return final result in p_{gBest}
Append the result to the results list
End For
Select best global subswarm result
return NewColumns

CHAPTER 3. TSA RESOURCE ALLOCATION

3.1 Introduction

Aviation security has been a regular topic of study for the last two decades. In the aftermath of the terrorist attacks of September 11, 2001, the evolution of risk-based aviation security began. In response to the attacks, the Aviation and Transportation Security Act went into effect, requiring that a computer-assisted passenger prescreening system evaluate all passengers. This system has been through several development cycles, starting with the Computer-Assisted Passenger Prescreening System (CAPPS), then the second-generation CAPPS II (operated by TSA), and now Secure Flight. Secure Flight is a risk-based passenger prescreening program that matches passengers' names against trusted traveler lists and watchlists and then identifies them as high or low-risk (Administration, n.d.). Based on information derived from both government and commercial databases, Secure Flight conducts risk assessments to determine which passengers might be eligible for TSA precheck screening or standard screening. The results also prevent potential passengers on the No Fly List and Centers for Disease Control and Prevention Do Not Board List from boarding an aircraft. (Sadler, 2016)

Security constructs have been designed as multi-layered systems to incorporate several security measures for screening methods. We are now able to integrate passenger, baggage, and cargo screening operations to model complex airport security paths. Over the past 18 years, there have been regular changes to all aspects of aviation security systems. Each change or new security measure is only considered if it improves the security stance for all travelers throughout the United States. Billions of dollars were initially invested in security measures before official plans were put in place. Initial analysis was completed after the fact, and changes have been necessary to ensure that we are doing our best to protect all travelers as technology and research have evolved.

Although there are many familiar elements of this research, there are new contributions attributed to the DHS's current ERM efforts and the desire to implement an all-encompassing model. The Office of Management and Budget has established government-wide ERM in recent years. However, two years prior, TSA was already experimenting with their implementation of ERM (TSA, 2014). Most organizations that have been able to implement ERM are traditional corporations that are looking to change their organizations' decision-making constructs to reduce risk and maximize profits. In this case, TSA is taking a security-based approach to implementing ERM by reducing risk and maximizing their risk posture. The overall understanding of ERM is the same across all organizations, however. ERM will be used to determine how the organization approaches decision-making, resource allocation, and all of its operations. The goal of ERM is not to eliminate all risk, but to effectively prioritize the response to issues an organization faces. For TSA, especially, the focus is on how ERM can help organize their resources to achieve their organizational and strategic objectives.

3.2 TSA Prior Work

3.2.1 TSA Enterprise Risk Management

DHS defines risk as “the potential for an unwanted outcome resulting from an incident, event, or occurrence, as determined by its likelihood and the associated consequences” (Council, 2010). By incorporating ERM into its strategy, TSA hopes to use a consistent analysis framework to balance risk and cost on a common basis across the enterprise (Minsky, 2013). Risk assessments must be connected to goals and activities within a risk taxonomy to give purpose and measurement of effectiveness. The TSA ERM program will provide a balanced quantitative approach in their RM program. Only by quantifying risks and tolerances upfront and using a common framework can the allocation of resources be applied to the methods that manage them effectively.

With regard to applying ERM to the TSA organization, Fletcher and Abbas provided a case study in 2017 (Fletcher & Abbas, 2018). The primary benefit of the case study is that it defines a clear alternative objective to profit by using Public Value (made up of multiple attributes). The same analysis was performed using Value-at-Risk (VaR) to determine the effect of focusing on risk thresholds (probability of exceeding a negative outcome). The authors show that the public value approach is preferred. However, for our research, the element of VaR can easily be translated to a similar measurement, such as maximizing the successful security alert (true alarm) rate. Although the research topics are similar, there is a significant difference in our quantitative approaches and the desired output. We are interested in a resource allocation optimization model to optimize risk and reward, and the Pareto frontier will represent the tradeoffs between the multiple objectives. The case study example is also based on passenger security screening in the threat scenario of the catastrophic destruction of a mid-flight large commercial aircraft due to a terrorist detonation of an improvised explosive device (IED). (Fletcher & Abbas, 2017).

3.2.2 Risk-Based Security

Passengers are categorized by risk to receive either less or more security screening from a notional baseline applied to any passenger with unknown risk. Passengers are grouped into four categories, with travelers in either of the first two groups subject to reduced screening. Trusted travelers have successfully completed an extensive background check. Low-risk passengers have completed a less comprehensive background check. Unknown Risk passengers are subject to the same level of primary screening currently applied to all passengers. High-risk passengers undergo far more extensive security screening due to suspected ties to terrorist organizations or individual terrorists (Fletcher, 2011). The risk-based approach seeks to find an appropriate balance between identifying individuals with ill intent and broadly applied searches for weapons and other prohibited items. The risk-based approach further develops the trusted traveler's program by reducing primary screening measures for low-risk passengers and implementing more significant security measures for high-risk passengers.

3.2.3 Topological Networks

Network topology refers to how network nodes are physically or logically organized with respect to one another. If you think of your network as a city, and the topology as the road map, there are several ways to arrange a network, each with its own advantages and disadvantages. Depending on the needs of the scenario, certain arrangements can provide a greater degree of connectivity and security (What Is Network Topology?, 2019).

A topological network is formed by two sets of elements: nodes (or vertices), which are connected points in the network, and links (or edges), which are physical or defined, connections between pairs of nodes. Within each set, elements are non-distinguishable, and can have non-directed and have equal weighted links or are distinguishable with potentially directed and unequal weighted links.

A real-world example is a structure consisting of atoms and bonds which represent the nodes and links, respectively. A path is a series of sequentially connected nodes and links without overlap, and a ring is simply a closed path. Yuan (Yuan, 2002) presents an efficient algorithm for finding primitive rings in a topological network. A ring with n links is called an n-ring. For each node (link), there should exist a local cluster of rings that contains that node (link). We call such a set of rings a ring-cluster of that node (link). A primitive ring is simply a ring without a shortcut.

In Kos (Kos et al., 2002), they focus on topological planning of large-scale communication networks like those used by telecom operators. Due to the high costs of network equipment and the large geographical spread of the networks, finding an optimal topology is critical. They present a 3-stage network design process that decides which network elements to include in the backbone, selects network topology, and determines node and link capacities needed for traffic management and routing.

Rai (Rai, 2019) considers the problem of inferring the topology of a network using the measurements available at the end nodes without cooperation from the internal nodes.

They provide a simple method to obtain path interference, which identifies whether two paths in the network intersect with each other. They take this information and formulate the topology inference problem as an IP and develop algorithms to solve it optimally. The method is applicable for networks with tree and ring topologies.

In Gounaris (Gounaris, 2015), they propose the use of mixed-integer linear optimization modeling and solution methodologies to address the Network Generation Problem. They present several useful modeling techniques and apply them to mathematically express and constrain network properties in the context of an optimization formulation. They then develop complete formulations for the generation of networks that attain specified levels of connectivity, spread, similar node connectedness, and robustness.

Topological networks are prominent in biological structures. Yip (Yip, 2007) introduces a general class of node dissimilarity measures based on the notion of `topological' overlap. The resulting generalized topological overlap measure (GTOM) generalizes the standard topological overlap measure (TOM) introduced by Ravasz et al. (Ravasz, 2002). Specifically, the *m*-th order version of this family is constructed by counting the number of *m*-step neighbors that are shared by a pair of nodes and normalizing it to take a value between 0 and 1. The TOM was introduced to analyze metabolic networks with distinct organisms that are organized into connected topological modules that combine in a hierarchical manner (Ravasz, 2002). The primary use of the GTOM measures is the identification of network modules (sets of tightly connected nodes). But it can also be used to define novel measures of node connectivity. These GTOM based connectivity measures go beyond the usual nodal degree (number of connections) by taking into account higher-order connections. In (Yip 2007), they discuss the properties of the GTOM measures and provide empirical evidence that they are useful in the context of gene co-expression network analysis.

A topological representation of the TSA risk factors became a natural fit for the problem. It provides a method to detail the interdependencies and hierarchy for a correlated network that operates without quantitative values. From here, the GTOM method utilized for gene co-expression (Yip, 2007) was successfully applied to the risk factors and then integrated into the primary objective for the integer program.

3.2.4 TSA Integer Programming Problems

Early discrete optimization research for aviation security dates back prior to September 11, 2001. However, the first screening optimization models appear to develop post 9/11. These models included checked baggage for high-risk passengers screened for explosives, selectee, and non-selectee screening, where the goal was determining how to deploy and use limited baggage screening devices optimally. These models led to the development of the following baggage security models (McLay, 2011). First, the Uncovered Flight Segment Problem (UFSP), which found a subset of flights to screen such that the total amount of covered flights subject to a screening capacity is maximized. Next came the Uncovered Passenger Segment Problem (UPSP), which found a subset of flights to screen such that the total amount of passengers of covered flights subject to a screening capacity was maximized. Last, the Uncovered Baggage Segment Problem (UBSP) found a subset of flights to screen that will maximize the total number of bags screened subject to screening capacity.

The research continued to develop and expand every year, as did the growing concern for the safety of travelers and citizens. The next iteration of models included how to match the limited security measures to the number of passengers that needed to be screened. (Poole & Passantino, 2003) proposed a risk-based airport security system that depends on sorting passengers and their bags into two or more risk classes with screening resources applied to each class according to its risk level. The significant finding was that such a risk-based system might be more effective than the system where all passengers and bags receive equal scrutiny. (McLay, 2011) considered the multilevel allocation problem (MAP) where every would-be passenger is assigned an assessed threat value, which quantifies the risk associated with the characteristics of the passenger. The assignment of passengers to a given number of classes of checking devices is done to maximize the true alarm rate, subject to budget constraints. (Sewell et al., 2012) used a similar idea to consider how to allocate explosive screening devices for checked baggage in multiple airports setting where passengers are divided into classes according to their perceived risk levels.

A different approach was taken by (Babu et al., 2006) who assumed that passengers are indistinguishable with respect to risk attributes, and considered how to assign passengers to different combinations of check stations such that the false alarm rate is minimized while keeping the false clear rate within specified limits. A major conclusion was that passenger grouping is beneficial even when the threat probability is assumed constant across all passengers. (Nie, et al., 2009) took Babu’s model one step further, and instead of assuming that all passengers maintain a constant threat probability, they assumed that passengers are classified into several groups of passengers. The objective was to minimize the probability of false alarm, and they compared it to Babu's model through performance measures of the overall probability of false alarm and the total number of screeners needed.

3.3 Our Contributions

This research takes into consideration many of the previous operational level airport security models and their security resource allocation objectives and aggregates and expands their objectives and constraints into a single large scale mixed integer programming portfolio optimization problem with a primary objective of maximizing the risk posture of the TSA. The risk posture of the TSA is determined by their risk levels and risk factors, none of which I will go into detail in this article due to the necessary discretion. The chapter is organized as follows. Section "TSA Prior Work" provides a background of the various types of security models that have been proposed/implemented and a brief history of ERM and risk analysis. Section "Mathematical Model" models the problem as a multi-objective nonlinear integer program. Section "Solution Methodology" describes using a Dantzig-Wolfe decomposition type of approach to handle the nonlinear constraints and objective and the binding constraint of allocating the devices across the set of airports. It also describes implementing a column generation approach to solve the model to optimality due to a large number of decision variables. Section "Computational Challenges" reports computational results from several problem instances that demonstrate achieving solution optimality, while the "Conclusions" section provides concluding remarks.

3.4 Risk Design

In considering risk, we needed to explore the effect of risk correlation (Kendrick, 2008). The overall risk across all projects depends on correlation. For programs or portfolios of related projects, the risks are correlated, and uncertainty may increase. For portfolios of independent projects, risks may be offset and possibly decrease uncertainty. This initial information was critical in helping to identify all "projects" and the project-level risks that represent significant exposure. All program risks were listed related to complexity or scale, and all shared resources were identified too. On a strategic level, all project interconnections or interdependencies were fully mapped out. Portfolio correlation factors can be attributed to the reliance on similar technologies or resources and also due to having common project risks. So we pursued risk correlation analysis to lower the program and portfolio risk and begin the process by generating a correlation matrix, including all of the organizational risk factors.

Developing the interdependencies in enterprise risk was an intricate process. It required a certain level of understanding of TSA as an enterprise, their risk appetite, and the associated risks. Due to the nature of the organization and the security of our nation, I will discuss the evolution of a risk interdependency mapping but will exclude any pertinent information. Although TSA is a governmental organization that does not ascribe to a capitalist set of objectives, ERM is still a very critical tool for organizations to implement. The tricky part is making ERM work for "you." With that said, we reviewed all of the current TSA enterprise risks, tracked their associated risk appetites, and then defined interdependency relationships between all of the risk factors.

After developing the initial risk level and risk factor construct, the model development was the next critical stage. Portfolio optimization is the optimal assignment of limited capital to available financial assets to achieve a reasonable trade-off between profit and risk objectives. The classical Markowitz model uses the variance as the risk measure and is a quadratic programming problem (Markowitz, 1952). Although we are not constructing traditional portfolio optimization models, there are similar thoughts in assigning limited capital to available security measures. Portfolio models are typically adapted from this original construct, and correlation matrices play a critical role in risk management. Since we do not have reliable estimates for the correlation or historical data to derive them from, we will adapt network analysis techniques to make a pseudo correlation matrix.

A network can be represented by an adjacency matrix, $A = [a_{ij}]$, that encodes whether or how a pair of nodes is connected (Ravasz et al., 2002). Let A be a symmetric matrix with entries in [0,1]. For an unweighted network, entries are integer values of 0 or 1 depending on whether or not two nodes are adjacent (connected). A more complex network might depend on the degree of interaction between nodes. The matrices are then normalized such that the diagonals are equal to 1. The off diagonals are scaled values, thereby extending the adjacency matrix from the binary case to values in the range of [0,1]. Generalized Connectivity equals the row sum of the adjacency matrix. For unweighted networks, it is the number of direct neighbors, $k_i = \sum_j a_{ij}$.

A critical aim of network analysis is to detect subsets of nodes (modules) that are tightly connected to each other (Yip & Horvath, 2007). The topological overlap matrix (TOM) is a similarity measure for biological networks. In a hierarchical network, nodes can be connected by links carrying a weight J_{ij}. The weighted degree of node i is defined as: $w_i = \sum_{j=1:j\neq \mathrm{i}}^{N} J_{ij}$.

The original TOM does not account for the presence of weights $O_{ij} = \frac{|N(i)\cap N(j)|+A_{ij}}{\min\{|N_1(i)|,|N_2(j)|\}+1-A_{ij}}$ The presence of weights can be accounted for by modifying the previous equation by replacing the unweighted adjacency matrix with the normalized coupling matrix ($J_{ij}/J_{\max}$) $O_{ij} = \frac{1}{J_{\max}} \times \frac{\sum_{k=1}^{N} J_{ik}J_{kj}+J_{ij}J_{\max}}{\min\{w_i,w_j\}-J_{ij}+J_{\max}}$, as seen in Table 3.1. If $O_{ij} = 1$ then the node with fewer connections satisfies the conditions that all of its neighbors are also neighbors of the other node, and it is connected to the other node. Alternatively, $O_{ij} = 0$ if i and j are un-connected and the two nodes do not share any neighbors.

Table 3.1 - Weighted Topological Overlap Matrix

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16	R17
R1	1.00	0.22	0.13	0.17	0.07	0.07	0.07	0.13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R2	0.17	1.00	0.17	0.07	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R3	0.07	0.17	1.00	0.00	0.17	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R4	0.11	0.14	0.14	1.00	0.14	0.11	0.33	0.21	0.00	0.00	0.00	0.00	0.17	0.00	0.17	0.00	0.00
R5	0.00	0.07	0.17	0.00	1.00	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R6	0.00	0.00	0.08	0.00	0.20	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R7	0.06	0.06	0.06	0.33	0.06	0.06	1.00	0.33	0.00	0.00	0.00	0.00	0.22	0.00	0.22	0.00	0.00
R8	0.24	0.24	0.24	0.30	0.24	0.24	0.33	1.00	0.19	0.00	0.00	0.00	0.20	0.00	0.20	0.11	0.00
R9	0.33	0.33	0.33	0.00	0.33	0.33	0.00	0.33	1.00	0.14	0.00	0.00	0.20	0.00	0.20	0.11	0.00
R10	0.18	0.26	0.22	0.10	0.22	0.18	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R11	0.00	0.00	0.00	0.00	0.40	0.40	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.29	0.00	0.00	0.00
R12	0.27	0.21	0.27	0.19	0.27	0.33	0.22	0.22	0.30	0.33	0.29	1.00	0.41	0.33	0.41	0.24	0.22
R13	0.00	0.00	0.00	0.19	0.00	0.00	0.22	0.22	0.22	0.11	0.00	0.00	1.00	0.00	0.38	0.00	0.00
R14	0.00	0.00	0.11	0.00	0.33	0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
R15	0.26	0.36	0.30	0.33	0.30	0.23	0.22	0.13	0.00	0.00	0.00	0.00	0.17	0.00	1.00	0.14	0.07
R16	0.21	0.18	0.26	0.14	0.26	0.26	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.17
R17	0.25	0.18	0.25	0.14	0.23	0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.14	1.00

As previously mentioned, TSA employs a system of interconnected security layers to deter, detect, and prevent exploitation of commercial aviation by terrorists. This analysis incorporates all current measures and newly tested measures but is not a comprehensive list of security measures employed. There are likely to be additional measures or new technology that will be considered for model inputs. Each security measure has an interdependent relationship with the enterprise risk factors identified by TSA. An additional assignment matrix, Table 3.2, can be constructed to show direct relationships between them. From here, we generate an assignment matrix allowing us to relate the risk taxonomy to the security measures put in place. Depending on the security measure, a failure to detect a threat could impact multiple risk elements of the taxonomy.

Table 3.2 - Security Measure Assignment (SMA) Matrix

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16	R17
SM1	1	0	1	1	0	1	0	0	0	0	1	0	1	1	0	1	1
SM2	1	0	1	1	0	1	0	0	0	0	1	0	1	1	0	0	1
SM3	1	0	1	1	0	1	0	1	0	0	0	0	1	0	1	1	1
SM4	1	0	1	1	1	1	0	1	0	0	0	0	1	0	1	1	1
SM5	1	0	1	1	1	1	0	1	0	0	1	0	1	0	1	1	0
SM6	1	0	1	1	0	1	0	1	0	0	1	0	1	0	1	1	0
SM7	1	0	1	1	0	1	0	1	0	0	1	0	1	1	1	1	0
SM8	1	0	1	1	0	1	0	0	0	0	1	0	1	1	0	1	0
SM9	1	1	1	0	1	1	0	0	1	1	1	1	1	0	0	0	0
SM10	1	1	1	0	1	1	0	0	1	1	1	1	1	1	0	0	0
SM11	1	1	1	0	1	1	0	0	1	1	1	1	1	1	0	0	0
SM12	1	1	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0
SM13	1	1	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0
SM14	1	1	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0
SM15	1	1	1	1	1	1	1	1	1	1	0	1	1	0	0	1	0
SM16	1	1	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0
SM17	1	1	1	1	1	1	1	0	0	0	1	0	1	1	0	1	0
SM18	1	1	1	1	1	1	1	1	0	0	1	0	1	1	1	1	0
SM19	1	1	1	0	1	1	0	0	0	0	1	0	1	1	0	1	0
SM20	1	1	1	0	1	1	0	0	0	0	1	0	1	1	0	1	0
SM21	1	1	1	0	1	1	0	0	0	0	1	0	1	1	0	1	0
SM22	1	1	1	1	1	1	0	0	0	0	0	0	1	0	0	1	0
SM23	1	1	1	1	1	1	0	1	0	0	0	0	1	0	1	1	0
SM24	1	1	1	1	1	1	0	0	0	0	0	0	1	0	1	1	0
SM25	1	1	1	1	1	1	0	0	0	0	0	0	1	0	1	1	0
SM26	1	1	1	1	1	1	0	1	0	0	0	0	1	0	1	1	0

We developed a method to calculate the risk posture evaluation metric as a means to integrate the risk factors and security measures that are put in place by TSA. Based on the type of information that we are working with and the data available, our primary focus will be to maximize our Risk Posture versus minimizing a Risk Score. Risk Posture is used to describe overall readiness to take the risk, which is an accurate description of TSAs' strategy to always be prepared. Our goal is to maximize the overall risk posture by minimizing our risk. We chose this approach because it allowed us to utilize the probability of detection versus the probability of attack. Although we do not know exact values for the probability of detection, there are estimated values of the conditional probability of detection given there is a particular type of threat, $p_d \ \forall \ d \in D$. These values are derived from manufacturer capability tests. The risk posture is then calculated by multiplying the adjusted risk values by the selected security measures. We approached the calculation of the risk posture in the equations (3.1) - (3.3) below. The result of equation (3.2) is the Adjusted Risk Values, a reduced set of coefficients for each security measure resource.

- The product of Security Measure Assignment (SMA) matrix and TOM

$$\text{Risk Impact Values (RIV)} = \text{TOM*SMA} \tag{3.1}$$

- A formula that calculates the score

$$\text{Adjusted Risk Values (ARV)} = p_d \times RIV_d \tag{3.2}$$

- A set of thresholds that help translate the calculated score

$$\text{Risk Posture} = \sum_{d=1}^{D} x_d \times ARV_d \tag{3.3}$$

3.5 Data Collection

In this chapter, we re-examine, integrate, and expand the works of (Nie et al., 2009) and (Sewell et al., 2013). In the context of a type of passenger prescreening system exemplified by Secure Flight, we want to determine the optimal allocation of threat detection devices and measures for screening checked baggage, carry-on baggage, and passengers across a set of airports such that we maximize risk posture, maximize the number of threats to be detected, and minimize the overall false alarm rate while considering passenger threat classification. We imposed constraints on time available at each check station, flow capacity at security stations, budget, as well as staffing needs at each check station.

At airports, all passengers and items pass through various check stations, each outfitted with several security measures for threat detection. It is standard practice for all passengers, and items are subjected to a series of screening at mandatory check-ins. For example, document verification, walk through metal detectors, water bottle scanners, etc. After inspection of a passenger/item, the screening measure or personnel will give a clear signal (No Threat) or an alarm signal (Threat). There are four types of alarms, and while all four are critically important, the two alarms that we are concerned with are true alarms and false clears. True alarms correctly detect existing threats, and false alarms give an alarm when no threat exists.

False alarm and false clear probabilities are performance measures for the screening system. Higher performance means lower values of these probabilities. False clears are potentially fatal for allowing threats to go undetected, and false alarms increase inspection delays and mean that the system is not as reliable as we hope.

Risk-based security paradigms classify passengers into different security classes based on the perceived risk of each passenger, where the passengers and their checked and carry-on baggage are screened using pre-specified combinations of detection devices (e.g., magnetometer, x-ray machine) and procedures (e.g., hand search, pat-down). Within each security class, a passenger or bag may undergo screening from multiple devices or procedures. A passenger or bag clears the security checkpoint only if all devices and procedures used in this class detect no threat. If a threat is detected or if reasonable suspicion of a threat arises, then the passenger or bag undergoes additional screening, usually through a more threat-specific, time-consuming process. The use of devices as part of the security operations endure costs associated with installing, operating, and maintaining the devices. The preponderance of costs associated with screening procedures is associated with employing personnel and implementing these procedures. The fixed costs are associated with installing devices and maintaining the devices for screening procedures. The costs associated with operating the devices are based on the expected life and time in the operation of each device, while the implementation costs of screening procedures are based on the employee compensation of security personnel. In addition to these cost restrictions, each device is manufactured to provide a maximum throughput capacity. Thus, the expected number of passengers in each security class aids in determining the capacity requirements for deploying existing and new detection devices at each airport. Sewell's device allocation model aids in the inherent trade-off decision between using faster, more accurate, and expensive devices versus slower, less reliable, but less expensive devices, or even some combination of the two. These decisions are highly influenced by resource constraints, including cost, personnel, and space availability, hence the decision as to the type and number of devices and procedures to use for screening high-risk and low-risk passengers to maximize the total security (probability of threat detection) can be very challenging. This is especially so when considering a limited number of devices available to deploy across a set of airports, each with its own individual resource constraints.

3.6 Mathematical Model

Several assumptions must be made to formulate a mathematical model for this problem. First, a passenger prescreening system is used in a risk-based security screening approach to quantify the perceived risk of each passenger. Second, the resulting threat assessment is viewed as an accurate representation of the passenger's true risk to the air transportation system, based on intelligence gathered by the TSA pertaining to prior travel history, origin and destination itinerary, ticket purchase method, current behavioral attributes and other security-sensitive information. Third, the detection devices used to screen passengers and their baggage operates independently of one another, such that the use of one type of device does not affect the cost or threat detection performance associated with any other device under consideration. Lastly, while there is a cost associated with deploying new devices at an airport, it is assumed that there is no cost associated with removing existing devices from an airport security checkpoint.

This section outlines the notation, constraints, and objective function used to describe the screening device allocation model. The notation for the parameters and decision variables used in the model are as follows:

The parameters and decision variables used in the model are as follows:

Parameters	**Description**
T	The total number of airports under consideration
k	Index for airport $k=1,2,\ldots,T$
D	The number of screening device types
d	Detection device type $d=1,2,\ldots,D$
J	Number of screening groups
j	Screening group $j = 1,2,3,4$
$D(j)$	Detection devices d within screening group j
M_c	Number of passenger classes at airport k
c	Index for passenger class $c=1,2,\ldots,M_c$
A_{ck}	Average value of perceived risk for passengers assigned to class c at airport k
B_{ck}	Number of checked bags per hour screened in class c at airport k
Cj	Maximum throughput (passengers or bags/hour) within screening class j
E_{dk}	Number of existing devices of security measure type d at airport k
F_d	Fixed Cost ($/device) associated with device type d
G_{ck}	Number of carry-on bags per hour screened in class c at airport k
H_{ck}	Number of passengers per hour screened in class c at airport k
K_{dk}	The capacity of device d at airport k
I_d	Installation cost ($/device) associated with device type d
O_d	Operating cost ($/device) associated with device type d
P_d	Conditional probability of detecting a threat given there is a threat for device type d
cp_c	Probability of a passenger belonging passenger class c
α_c	The conditional probability that passenger carries a threat given they belong to class c carries a threat
β_{jc}	The conditional probability that there is a threat in screening group j given a class c
q_d	Conditional probability of clearing a non-threat item given there is no threat for device type d
TB_k	Total hourly budget ($) available at airport k
t_d	Time taken to check one passenger or bag at device d
U_d	Number of device type d available for installation
z_d	Time multiplier to verify any alarm at any device

Decision Variables	**Description**
x_{cdk}	Binary variable where $x_{cdk} = 1(0)$, if security measure type d is (not), used to screen class *c* bags at airport *k*
y_{dk}	Number of security measure type *d* to be used at airport *k* (integer)
s_{dk}	Number of security measure type *d* to be installed at airport *k* (integer)

The explosive screening device allocation model proposed in (Sewell et al., 2013) assigns the types, d, and numbers, x_{cdk}, of detection devices to each class at each airport. To accomplish this, the number of devices of type d to be installed at each airport, s_{dk}, is found by subtracting the number of devices of type d currently existing from the number of devices of type d used in total at each airport, equation (3.4). Therefore,

$$s_{dk} = y_{dk} - E_{dk}, \text{(Device Installation Constraint)} \tag{3.4}$$

provided $y_{dk} \geq E_{dk}$ (and 0 otherwise), for $d = 1,2, \ldots, D$ and $k = 1,2, \ldots, T$.

Using the notation provided, the installation, operating, and total fixed costs at each airport k can be found such that the combined installation, operating, and fixed costs satisfy the total hourly budget, TB_k, for airport $k = 1,2, \ldots, T$, since it is assumed that there is no cost associated with uninstalling a screening device. Next, in equation (3.5) the number of new devices to install at each airport, y_{dk}, must be considered. This relies on the capacity performance of the screening devices, captured by the number of bags each device type can handle per hour, C_d, and the number of bags screened in each class within a particular airport, B_{ck}. Dividing the hourly rate of bags screened in class c at airport k by the maximum throughput of device type d yields the number of security devices of type $d = 1,2, \ldots, D$ necessary to screen all baggage using this particular device,

$$y_{dk} = \left\lceil \sum_{c=1}^{M_k} B_{ck} x_{cdk} / C_d \right\rceil \qquad \forall d = 1,2,\dots,D \; and \; k = 1,2,\dots,K \tag{3.5}$$

(Resource Capacity Constraint)

Lastly, the number of new devices installed at all airports must be less than or equal to the total number of new devices available, equation (3.6), and so, the device resource availability constraint becomes

$$\sum_{k=1}^{T} s_{dk} \leq U_d, \qquad \forall d = 1,2,\dots,D \quad \text{(Resource Availability Constraint)} \tag{3.6}$$

In Nie's (Nie et al., 2009) mathematical model, their objective was to determine the fraction of passengers that are assigned to threat class c and the staffing needs at each check station within each screening group. Changing the parameters into our notation, we have the following equations.

$$\sum_{c=1}^{M_k} cp_c \alpha_c \sum_{j=1}^{J} \beta_{jc} \prod_{d \in D(j)} (1 - p_d) x_{cdk} \leq \delta \qquad \forall\, k \in T \tag{3.7}$$

$$\sum_{c=1}^{M_k} H_{ck}\, cp_c \sum_{d \in D(1)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c1} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c\right)\right) t_d \leq \sum_{d \in D(1)} C_1\, K_{dk}, \quad \forall k \tag{3.8}$$

$$\sum_{c=1}^{M_k} B_{ck}\, cp_c \sum_{d \in D(2)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c2} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c\right)\right) t_d \leq \sum_{d \in D(2)} C_2\, K_{dk}, \quad \forall k \tag{3.9}$$

$$\sum_{c=1}^{M_k} G_{ck}\, cp_c \sum_{d \in D(3)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c3} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c\right)\right) t_d \leq \sum_{d \in D(3)} C_3\, K_{dk}, \quad \forall k \tag{3.10}$$

$$\sum_{c=1}^{M_k} H_{ck}\, cp_c \sum_{d \in D(4)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c4} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c\right)\right) t_d \leq \sum_{d \in D(4)} C_4\, K_{dk}, \quad \forall k \tag{3.11}$$

$$\sum_{c=1}^{M_k} (1 - \alpha_c) cp_c \sum_{j=1}^{J} \left(1 - \prod_{d \in D(j)} q_d x_{cdk}\right) \qquad \forall k \in T \tag{3.12}$$

Constraint (3.7) ensures that the false clear probability is within the upper bound, δ, set by the appropriate security authority. Constraints (3.8) - (3.11) guarantee that checking of baggage or passengers at each screening group is completed before the allotted time. The objective function, equation (3.12), minimizes the probability of false alarm across each airport.

The objective defined for the device allocation model is based on the probability of a device correctly detecting a threat, the underlying risk level of the bags screened, and the number of bags screened within each security class. (Sewell et al., 2012) defines the probability of detecting a threat within security class c, L_{ck}, (at airport k) as the probability that at least one of the device types used in that class detects the threat, equation (3.13),

$$L_{ckj} = 1 - \prod_{d \in D(j)} (1 - x_{cdk} P_d) \qquad \forall\ c = 1,2,\dots,M_k, k = 1,2,\dots,T, and\ j \in J \tag{3.13}$$

and where P_d is defined as the conditional probability that a threat is detected by security device type d, given that a threat is present.

The risk level of each class, $R_{c,k}$, is defined as the average perceived risk value of the passengers in security class c at airport k times the rate of baggage screened within that class. This value is normalized between zero and one by dividing over the total risk associated with all security classes within airport k in equation (3.14),

$$R_{ck} = \frac{A_{ck} B_{ck}}{\sum_{c'=1}^{M_k} A_{c'k} B_{c'k}} \tag{3.14}$$

The risk level of each security class relies heavily on the assumption that the prescreening system provides an accurate risk perception of the passenger population.

The objective function for the device allocation model is obtained by weighting each airport by the rate at which passengers/bags/carry-on bags must be screened at that airport and the risk level associated with screening these bags using either new or existing detection devices. Using equations (3.13) and (3.14), the objective value of each airport is defined as the expected number of detected threats in equation (3.15),

$$SL_{2k} = \sum_{c=1}^{M_k} L_{ck,j=2} B_{ck} R_{ck} \tag{3.15}$$

By summing over all the airports under consideration, the total security level captures the expected total number of detected threats, equation (3.16),

$$\sum_{k=1}^{T} SL_k = \sum_{k=1}^{T} \sum_{c=1}^{M_k} L_{ck1} H_{ck} R_{ck} + L_{ck2} B_{ck} R_{ck} + L_{ck3} G_{ck} R_{ck} + L_{ck4} H_{ck} R_{ck} \tag{3.16}$$

Combining the objectives from equations (3.3), (3.12), and (3.16), the device allocation problem for multiple airports are defined by the nonlinear integer program,

Maximize

$$\sum_{c=1}^{M_k} \sum_{d=1}^{D} x_{cdk} p_d \times RIV_d \qquad \forall k = 1,2, \ldots, T \tag{3.17}$$

$$-\sum_{c=1}^{M_k} (1-\alpha_c) cp_c \sum_{j=1}^{J} \left(1 - \prod_{d \in D(j)} q_d x_{cdk}\right) \qquad \forall k = 1,2, \ldots, T \tag{3.18}$$

$$\sum_{c=1}^{M_k} \sum_{d=1}^{D} SL_k x_{cdk} \qquad \forall k = 1,2, \ldots, T \tag{3.19}$$

Subject To:

$$\sum_{d=1}^{D} (y_{dk} F_d + s_{dk} I_d) + \sum_{c=1}^{M_k} \sum_{d=1}^{D} x_{cdk} O_d B_{ck} \leq TB_k, \forall k = 1,2, \ldots, T \tag{3.20}$$

$$y_{dk} = \left\lceil \sum_{c=1}^{M_k} H_{ck} x_{cdk} / C_d \right\rceil, \forall d = 1,2,\dots,D(1), \text{and } k = 1,2,\dots,T \quad (3.21)$$

$$y_{dk} = \left\lceil \sum_{c=1}^{M_k} B_{ck} x_{cdk} / C_d \right\rceil, \forall d = 1,2,\dots,D(2), \text{and } k = 1,2,\dots,T \quad (3.22)$$

$$y_{dk} = \left\lceil \sum_{c=1}^{M_k} G_{ck} x_{cdk} / C_d \right\rceil, \forall d = 1,2,\dots,D(3), \text{and } k = 1,2,\dots,T \quad (3.23)$$

$$y_{dk} = \left\lceil \sum_{c=1}^{M_k} H_{ck} x_{cdk} / C_d \right\rceil, \forall d = 1,2,\dots,D(4), \text{and } k = 1,2,\dots,T \quad (3.24)$$

$$s_{dk} \geq y_{dk} - E_{dk}, \forall d = 1,2,\dots,D, \text{and } k = 1,2,\dots,T \quad (3.25)$$

$$\sum_{c=1}^{M_k} cp_c \alpha_c \sum_{j=1}^{J} \beta_{jc} \prod_{d \in D(j)} (1 - p_d) x_{cdk} \leq \delta \qquad \forall k \in T \quad (3.26)$$

$$\sum_{c=1}^{M_k} H_{ck}\, cp_c \sum_{d \in D(1)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c1} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c \right) \right) t_d \leq \sum_{d \in D(1)} C_1 K_{dk}, \quad \forall k \quad (3.27)$$

$$\sum_{c=1}^{M_k} B_{ck}\, cp_c \sum_{d \in D(2)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c2} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c \right) \right) t_d \leq \sum_{d \in D(2)} C_2 K_{dk}, \quad \forall k \quad (3.28)$$

$$\sum_{c=1}^{M_k} G_{ck}\, cp_c \sum_{d \in D(3)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c3} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c \right) \right) t_d \leq \sum_{d \in D(3)} C_3 K_{dk}, \quad \forall k \quad (3.29)$$

$$\sum_{c=1}^{M_k} H_{ck}\, cp_c \sum_{d \in D(4)} x_{cdk} \times \left(1 + z_d \left(p_d \sum_{c=1}^{M_k} \beta_{c4} \alpha_c cp_c + (1 - q_d) \sum_{c=1}^{M_k} (1 - \alpha_c) cp_c \right) \right) t_d \leq \sum_{d \in D(4)} C_4 K_{dk}, \quad \forall k \quad (3.30)$$

$$\sum_{k=1}^{T} s_{dk} \leq U_d \quad \forall \ d = 1,2,\dots,D \quad (3.31)$$

$$x_{cdk} \in 0,1 \quad (3.32)$$

$$y_{dk} \in Z^+ \quad (3.33)$$

$$s_{dk} \in Z^+ \quad (3.34)$$

Constraint (3.20) is the airport budget constraint (3.21) - (3.24) are the resource capacity constraints based on the screening rates for the screening areas, and constraint

(3.25) is the device installation constraint. Constraint (3.26) ensures that the false clear probability is within the upper bound, δ, set by the appropriate security authority. Constraints (3.27) - (3.30) guarantee that checking at each station is completed before the allotted time. Constraint (3.31) is the overall resource availability constraint. The integer program is nonlinear due to the product of the x_{cdk} decision variables contained in the threat detection term, L_{ck}, in (3.13). Constraint (3.31) effectively ties together the decision variables across all airports, potentially impacting the ability to decouple the problem and solve for each individual airport. The following section presents a Dantzig-Wolfe decomposition type of approach to solving the device allocation problem across multiple airports.

3.7 Solution Methodology

The optimization problem, in the form given by (3.17)-(3.34), is computationally intractable for a large number of airports. The objective function is nonlinear in the x_{cdk} decision variables, but is separable due to being a sum of the threat detection probability performance measures of each individual airport security system.

Constraints (3.20) - (3.30) correspond to the individual airports $k = 1,2, \dots, T$, indicating a potential solution methodology of optimizing the device allocation problem for each airport independently. However, constraint (3.31) contains an interaction among the devices used in all the airports, by ensuring that all new devices allocated to be installed nationwide have in fact been produced and are available.

The device allocation problem in (3.17)-(3.34) cannot be solved by a standard integer programming package because of the nonlinear objective function. (Sewell, et al., 2012) presents a heuristic approach to solving this nonlinear integer program by forming a Lagrangian relaxation of the device allocation model. In determining all of the feasible portfolio options, we also calculate all of the potential values for constraints (3.20) - (3.30), allowing us to remove the nonlinear constraints and further reduce the number of potential device combinations. We utilize the previous Dantzig-Wolfe decomposition (Sewell et al., 2013) approach on constraints (3.20) - (3.30). This allows us to eliminate the nonlinearities in the objective function, and if the model were small, it is shown to provide an optimal solution in a short time on a set of small computational examples. In our case, the model is significantly larger due to the increase in the airport network and the inclusion of the additional objectives and constraints. For the current Dantzig-Wolfe decomposition, we define the constraints for airport *k*.

After the decomposition, we can eliminate *j* as the index for the screening group. Let index *j* represent the portfolio of security measures selected for allocation at airport *k*. Then, let $x^{kj} = (x_{11}^{kj}, x_{12}^{kj}, \cdots, x_{1D}^{kj}, x_{21}^{kj}, x_{22}^{kj}, \dots, x_{2D}^{kj}, \cdots, x_{M_k1}^{kj}, x_{M_k2}^{kj}, \cdots, x_{M_kD}^{kj}$ be a binary vector of length $M_k x D$, where $X_k = x: x$ is feasible for airport $k = x^{k1}, x^{k2}, \dots, x^{k1}k, y^{kj} = \left(y_1^{kj}, y_2^{kj}, \dots, y_D^{kj}\right)$ be an integral vector of length D defined by $y_d^{kj} = \left\lceil \sum_{(c=1)}^{(M_k)} B_{ck} x_{cd}^{kj} / C_d \right\rceil$, for $d = 1,2, \dots, D$, and $s^{kj} = \left(s_1^{kj}, s_2^{kj}, \dots, s_D^{kj}\right)$ be an integral vector of length D defined by $s_d^{kj} = \max\left(0, y_d^{kj} - E_{dk}\right)$, for $d = 1,2, \dots, D$. Also, define n_k as the number of feasible solutions for airport k, and SL^{kj} as the security level for airport k associated with x^{kj}. The

x^{kj} is defined to be feasible for airport k if x^{kj}, y^{kj}, s^{kj} can satisfy the constraints for airport k. Notice that all of the feasible solutions for airport k can be generated by generating all binary vectors x^{kj} of length $M_k \times D$, computing y^{kj} and s^{kj} from x^{kj}, and then determining if x^{kj}, y^{kj}, s^{kj} is feasible for airport k.

Next, we define a binary variable r_{kj} for each feasible solution for each airport, where $r_{kj} = 1(0)$ if solution x^{kj} is (not) selected to be used at airport k. The master problem can now be written as the binary integer program seen below in equations (3.35)-(3.40),

Maximize (across all airports)

$$\sum_{j=1}^{J} \sum_{c=1}^{M_k} \sum_{d=1}^{D} r_{kj} x_{cdk} p_d \times RIV_d \qquad \forall\, k = 1,2,\dots,T \qquad (3.35)$$

$$-\sum_{c=1}^{M_k} (1-\alpha_c) p_c \sum_{j=1}^{J} r_{kj}\left(1 - \prod_{d \in D(j)} q_d x_{cdk}\right) \qquad \forall\, k = 1,2,\dots,T \qquad (3.36)$$

$$\sum_{k=1}^{T} \sum_{j=1}^{J} SL^{kj} r_{kj} \qquad \forall\, k = 1,2,\dots,T \qquad (3.37)$$

Subject To:

$$\sum_{k=1}^{T} \sum_{j=1}^{J} s_{dk} r_{kj} \le U_d \qquad \forall\, d = 1,2,\dots,D \qquad (3.38)$$

$$\sum_{j=1}^{J} r_{kj} = 1 \qquad \forall\, k = 1,2,\dots,T \qquad (3.39)$$

$$r_{kj} \in \{0,1\} \;\; \forall\, k = 1,2,\dots,T \text{ and } j \in J \qquad (3.40)$$

The objective functions in (3.35) - (3.37) now select the solutions that maximize the total security level of the airports and that maximize the overall risk posture of the DHS, taking into account all airports. Constraint (3.38) ensures that the total number of new devices installed does not exceed the total number available (i.e., similar to the original binding constraint in (3.31)), while constraint (3.39) ensures that precisely one solution is chosen for each airport. The parameters for the master problem, SL_{kj} and s_d^{kj}, were created by using a combinatorial algorithm to generate all the possible solutions (feasible and infeasible) for each airport, and then selecting the feasible ones. The master problem is a binary integer program that can be solved by any standard solver (Gurobi 9.0). After solving the master problem, the optimal solution to the original problem can be constructed for each airport k from the x_{kj} corresponding to variable r_{kj} that equals one.

3.8 Computational Challenges

As previously mentioned, solving a model of this type is difficult purely due to the computational complexity. Much of this research effort is dedicated to designing solution strategies that will provide strong solutions to massively large mixed-integer programs. In the original model, the MIP includes 45,760 decision variables and 35,666 constraints. In order to linearize the risk structures or nonlinear portions of the model, a decomposition was necessary. A Dantzig-Wolfe decomposition is applied and discussed further in the following section. The Dantzig-Wolfe decomposition makes sense based on the problem structure. It allows division of the optimization problem into two groups of "easy" and "hard" constraints. The "hard" constraints are not necessarily difficult, but they complicate the LP by making it nonlinear and more difficult to solve. When these hard constraints are removed from the problem, then more efficient techniques can be applied to solve the remaining linear program or, in this case, IP.

Our primary solution technique is column generation due to the number of decision variables being exponentially large. After the Dantzig-Wolfe decomposition was applied, the original three decision variables were reduced to a single composite binary decision variable representing whether or not a specific security measure combination for the threat classes at airport k was applied. The number of possible portfolios of security measures is enormous. Rather than enumerating all the possibilities, we can generate only relevant patterns by solving the subproblem. We encounter this scenario in our problem. Even though the Dantzig-Wolfe decomposition did reduce the overall number of portfolio options, we were still left with 259k possibilities. The number of potential portfolios in this situation is enormous, and while it is possible to enumerate all of the possibilities, it is nearly impossible to generate a model that large or even solve one.

In the column generation method, only a (usually small) subset of the variables is used initially. The method sequentially adds columns (i.e., variables), using information given by the dual variables for finding the appropriate variable to add. In the most recent approach, the Dantzig-Wolfe decomposition presented in (Sewell et al., 2013), Sewell enumerated all the possible combinations of security measures for small screening area for a limited number of airports. However, we encounter some very limiting issues when trying to expand the problem to include all screening areas, all security measures, and all airports. The number of potential portfolio combinations must be significantly reduced to make this problem remotely approachable. The previous number of solutions was small, which allowed solving to optimality. As mentioned, in the larger model formulation, the number of decision variables is significantly larger than the number of constraints making it the perfect candidate for column generation. The main idea is that typically only a subset of variables are required in the basis to reach optimality, while other variables are non-basic and have a zero value. Column generation exploits this by only considering variables with the potential to improve the objective function value, indicated by the negative reduced costs. In each iteration of the column generation method, two problems are solved successively; the RMP and the SP. By solving the RMP, the master problem using a subset of variables, we obtain a vector of the dual values associated with the constraints. The dual information is then inputted into the SP, with the goal of identifying a new variable and an associated coefficient column with a negative reduced cost, which could potentially improve the objective function value. If such a variable and column are identified, then they are added to the RMP. The RMP is then optimized again, and the process is repeated. Otherwise, an optimal solution of the RMP is also an optimal solution to the original problem.

Even with the reduced number of potential combinations and swapping the constraints from the primary linear program to the decomposition, the number of variables and constraints is still enormous. In order to truly make this problem useful, a strong combination of pre-solving and employing column generation as a solution technique is necessary.

The decomposition changes the structure and looks at a full enumerated security measure combination list for all 440 airports. There are 1,048,576 possible security measure combinations for two classes of passengers and 26 security measures. The decomposed model now has 461,373,440 decision variables and 466 constraints. All variables are binary or positive integers. The number of feasible combinations can be reduced to 259k through additional preprocessing based on sensible decision making. This reduces the number of decision variables to 128,480,000. Decomposition increases the number of decision variables, but drastically decreases the number of constraints. However, the model is still too large to be solved outright, which continues to motivate research on solution methods. Having an accurate and close to the optimal solution is essential since the solution represents the best set of security measures, concerning the objectives, for each airport.

3.9 Optimization

Many comparisons have been performed, comparing, and contrasting the various optimization approaches for multi-objective models. (Sawik, 2011) provides a comprehensive analysis of weighting, lexicographic, and reference point approaches to multi-objective portfolio optimization. A hierarchical or lexicographic approach assigns a priority to each objective and optimizes the objectives in decreasing priority order. At each step, the best solution is found for the current objective, but only from the solutions that do not degrade the solution quality for higher-priority objectives. Lexicographic optimization generates efficient solutions found by sequential optimization of the objectives. For our current problem, we want to avoid numerical issues, so we normalize our objectives into comparable, unitless values and, afterward, equally weight the objectives.

3.9.1 Particle Swarm Optimization

As previously mentioned in Chapter 2, we will employ PSO as one of our solution methods. This computational method optimizes a problem by iteratively trying to improve a candidate solution (particle) with regard to a measure of quality. The problem is solved by having a population of candidate solutions (swarm) and moving the particles around in the search space based on the particles' position and velocity. The swarm in PSO consists of a population, and each member of the population is called a particle, which represents a portfolio in this study.

The algorithm is guided by personal experience (pBest), overall experience (gBest), and the present movement of the particles to decide their next positions in the search space described by Kennedy. Each particle remembers its best previous position and the best previous position visited by any particle in the whole swarm. In other words, a particle moves towards its best previous position and towards the best particle. Further, the experiences are accelerated by two factors c_1 and c_2, and two random numbers generated between [0, 1], whereas the present movement is multiplied by an inertia factor w varying between [w_{min}, w_{max}]. This is a relatively simplistic metaheuristic that applies to many problem types and can search very large spaces of candidate solutions. Unfortunately, it does not guarantee an optimal solution is found, but the goal is to see relatively quick convergence of the particles towards a solution. In multi-objective problems, Pareto dominance is taken into account when moving the particles, and non-dominated solutions are stored as to approximate the Pareto front (Cura, 2009).

Using PSO to solve a discrete optimization problem, the PSO is initialized with a group of random particles (mixed-integer variable solutions). The algorithm searches for optima by updating the generations of particles. In each iteration, the particles are updated by two "best" values. First, we record the best solution (fitness, objective function value) that has been achieved so far. The objective value is also stored as p_{best}. Second, the algorithm records the best value obtained so far by any particle in the population, known as a global best and stored as g_{best}. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called l_{best}. The formulation of the swarm is determined by the specific problem, and in this case, each particle represents the complete set of portfolios selected for all of the airports. Therefore, each particle of a swarm (denoted by index *i*) must include all of the variables r_{ikj} and z_{idk} are the variables denoting the quantity of each security measure assigned to each airport.

After finding the two best values, the particle updates the velocity and positions of its variables with the set of equations below (3.41) - (3.43), as discussed previously in Chapter 2. Both ω_1 and ω_2 denote uniform random numbers between 0 and 1. *t* denotes the iteration number while vz_{idk}^{t} denotes the velocity of variable *z* within particle *i*, and vr_{ikj}^{t} denotes the velocity of variable *r* within particle *i*. vz_{idk}^{t} will be updated if security measure *d* is selected by the portfolio of security measures within particle *i* at iteration *t+1*. Thus particle *i* moves at iteration *t+1* as follows:

$$vr_{ikj}^{t+1} = vr_{ikj}^{t} + c_1\omega_1\left(r_{\text{pbest}} - r_{ikj}^{t}\right) + c_2\omega_2\left(r_{gbest} - r_{ikj}^{t}\right) \tag{3.41}$$

$$r_{pi}^{t+1} = round\left(\frac{1}{1+e^{-\theta}} - \alpha\right), where\ \theta = r_{ikj}^{t} + vr_{ikj}^{t+1}\ and\ \ \alpha \text{ is set to 0.06} \tag{3.42}$$

$$vz_{idk}^{t+1} = \begin{cases} vz_{idk}^{t} + c_1\omega_1\left(z_{pbest} - z_{idk}^{t}\right) + c_2\omega_2\left(z_{gbest} - z_{idk}^{t+1}\right) & if\ \ r_{ikj}^{t+1} = 1, \\ vz_{idk}^{t} & otherwise \end{cases} \tag{3.43}$$

For a given particle, if the velocity on the dimension r_{ikj}^{t} is zero, this particle will not move in that dimension at iteration $t+1$. Suppose $vr_{ikj}^{t} = 0$ and $\mathrm{r}_{ikj}^{t} = 0$, hence $1/(1+e^{0}) =$ 0.5 and *round(0.5) = 1*, which means that particle *i* will move in dimension r_i $\left(r_{pi}^{t+1} = 1\right)$ at iteration *t+1*. To avoid such an unwanted move, we can use α, as seen in equation (3.42).

The searching is a repeat process with stop criteria occurring when the maximum number of iterations has been reached, or the minimum error condition is satisfied. An advantage of PSO is not many parameters require tuning. The number of particles (solutions to record) is in the range of 20 to 40; difficult problems may require 100 – 200; however, we don't have that luxury due to the size of the actual model. The dimension of the particles (dimension of solution set) is prohibitively large in this case, requiring us to keep the number of particles to a minimum size. The range of particles is determined by the upper and lower bounds of the decision variables. $v_{\max}$ determines the maximum change one particle can take during one iteration. We require two $v_{\max}$ due to having binary and integer variables.

The multi-swarm PSO modification is a more recent popular approach. In the multi-swarm approaches, the population is divided into multiple sub-populations (sub-swarms) with different levels of communication. The benefit of this approach is that the population can maintain divergence, search for multiple promising regions, and partially converge into multiple optima. In (García-Nieto and Alba, 2012), the optimal swarm (sub-swarm) size is discussed in great detail. It is proposed that six particles per swarm might be the optimal number for PSO based algorithms.

Pluhacek demonstrates that the multi-warm performance was superior to the single swarm PSO in all cases (Pluhacek, 2016). Based on the comparative study of single swarm PSO versus multi-swarm PSO performed in (Pluhacek, 2016), we decided to utilize multi-swarm PSO, with five sub-swarms, with varying particle sizes from 5 to 10 particles per swarm. The control parameters were set as follows:

- Population Size: {5,6,7,8,9,10}
- Iterations: 5
- $v_{initial}$: 10% of the position
- w_{max}: 0.9
- w_{min}: 0.4
- c_1, c_2 = 1.49445 (learning factors)

The multi-swarm PSO is based on the local version of PSO with a new neighborhood topology. Many existing evolutionary algorithms require larger populations, while PSO needs a comparatively smaller population size. A population with three to five particles can achieve satisfactory results for simple problems. According to many reported results on the local version of PSO, PSO with small neighborhoods performs better on complex problems. Hence, to slow down convergence speed and to increase diversity to achieve better results on multimodal problems, in the MSPSO, small neighborhoods are used. The population is divided into small-sized swarms. Each sub-swarm uses its own members to search for better regions in the search space.

The multi-swarm optimization algorithm works as follows:

Input: MOP (1)
Swarm_size: number of the swarm particles
No_subswarms: number of subswarms
Step 1: Calculate Subswarm size= Swarm_size/No_subswarms
Step 2: For subswarm = 1 to No_subswarms do
For t=1 to Max_iterations do
Apply PSO algorithm as in Eqs. (3.41) - (3.43)

Update leaders archive
Update external archive
End For
Return final result in the external archive
Append the result to the results file
End For

3.9.2 Additional Heuristics

We applied column generation and additional solutions techniques to provide additional comparisons for speed and results. To expedite the column generation method, we again had to consider breaking apart the algorithm to accommodate a large number of options. This separation inspired the two heuristics mentioned below.

Heuristic 1, we took the full set of portfolio options, randomized the list, then broke them apart into buckets of 250 combinations each and optimized that subset of the portfolio options across all 440 airports. 250 portfolios was a suitably small subset that still allowed for the optimization to complete in a reasonably fast manner, especially compared to overall column generation. Not all fidelity is lost in using heuristics due to maintaining all 440 airports in each subproblem and keeping quantity assignment variables intact.

In Heuristic 2, we continued the theme of breaking down the problem into subproblems. We separated portfolios into randomized buckets but did the same with the airports as well. Each subproblem then represented a subset of both the airports and the possible combinations. In both randomized heuristics, optimization is performed at every iteration. The selected combinations (not the quantities of security measures) are placed into a pool of optimal combinations. The pool of possible portfolios is then used in a final optimization.

3.10 Empirical Results

The test problems/information used were collected from all of the articles that presented strong models ((Sewell et al., 2012), (Sewell et al., 2013), (Poole & Passantino, 2003), (Nie et al., 2009), (McLay et al., 2006), (Virta et al., 2003)). The fixed and installation costs are determined through the expected useful life of the device and on the amount of time the device would spend in operation over one year. Thus, these values reported in Table 1 reflect the yearly cost divided by the total number of hours spent in operation over the year, based on a peak 6 hours of operation per day, per device. Note that all cost values are in US dollars.

Passengers are assigned to a two-class system based on perceived risk information generated through a prescreening system (e.g., Secure Flight). This classifies passengers as being either high-risk (e.g., selectee) or low-risk (e.g., non-selectee), where the majority of passengers constitute the latter group. In the computational examples, 85% of passengers are deemed low-risk and assigned to Class 1, while the remaining 15% of passengers are assigned to the higher risk security Class 2.

The total number of passenger enplanements is actual enplanement data from 2016 collected from faa.gov (Transportation, n.d.). The hourly airport budget is based on an estimated annual budget value to be distributed across all airports. Individual airport budgets were simply distributed based on the proportion of passengers with a set minimum value. Next, the total number of passengers screened per hour at an airport is based on the average airport being operational 365 days a year and having 16 regular working hours per day. The operating cost of each security screening device or method is based on the annual operating cost of that method divided by the average hourly passenger screening rate. The maximum and minimum hourly screening rates per device are pulled from actual manufacturer device specifications. Last, the perceived risk values are generated from a normal distribution with mean 0.26 and standard deviation 0.12 for the low-risk passengers assigned to Class 1, and with mean 0.55 and standard deviation 0.12 for the high-risk passengers assigned to Class 2.

All combinations of all of the possible subsets of device types are generated for evaluation. The combinations of the security measures are grouped by screening group and are estimated by assuming which security measures should always be constant and which are optional. For example, as seen in Table 3.3 below, for the checked baggage screening, it was assumed that all checked luggage is screened by a CT scanner with additional screening performed by hand search. Therefore, all combinations must have both methods employed. Canine units and Explosive Trace Detection are both treated as secondary screening measures since they are not typically a primary line of defense at any airport, and there is no way to provide support to all airports. Based on this information, there are then four possible combinations of checked baggage security measures that can be employed. This same approach was conducted for the other screening measure groups.

Table 3.3 - Example of Security Measure Combination Restriction

	Disruption Rate	1-DR	Security Measure	1	2	3	4
SM1	50%	**50%**	Hand Search	1	1	1	1
SM2	80%	**20%**	Canine Unit (unit consists of two to four teams, 1 handler/2 Dogs per team)	0	0	1	1
SM3	70%	**30%**	Explosive Trace Detection (open bag trace)	0	1	0	1
SM4	80%	**20%**	Computed Tomography (CT) Scan (Electronic Detection System)	1	1	1	1

A potential combination of device types is chosen from these 1024 possible configurations for each passenger class for every airport, where each airport may have a different combination from any other airport. We obtain the number of device types used at each airport by dividing the hourly rate of passengers screened at that airport by the device hourly throughput rate.

3.10.1 Results and Analysis

This section describes the computational test problems used to evaluate the proposed solution approach. The binary integer programs in (3.35)-(3.40) were generated in Python 3.7.3 using the gurobipy module and solved with Gurobi 9.0. The Gurobi parameters were kept at their default values, apart from turning off the pre-solve option so that Gurobi would spend less time expanding the node structure. The computational experiments were conducted on a personal computer with an Intel dual-core processor, 2.4 GHz processor speed, and 16 GB of RAM. Sensitivity analysis was completed on the Georgia Institute of Technology High Throughput server cluster.

The data for all independent scenario instances remained consistent and incorporated all 440 airports. 1024^2 different combinations were produced, based on the security measures available. Below we report on the results from the solution methods described above. All of the results from the multi-swarm PSOs, heuristics solutions, and column generation are presented and compared below. Herein, we report 11 modeling methods to contrast the results.

- Model 1: Multi-Swarm PSO – 5 particles

- Model 2: Multi-Swarm PSO – 6 particles
- Model 3: Multi-Swarm PSO – 7 particles
- Model 4: Multi-Swarm PSO – 8 particles
- Model 5: Multi-Swarm PSO – 9 particles
- Model 6: Multi-Swarm PSO – 10 particles
- Model 7: Combined Solution MSPSO
- Model 8: Heuristic 1
- Model 9: Heuristic 2
- Model 10: Column Generation Pricing with Multi-Swarm PSO
- Model 11: Column Generation

Table 2 presents the computational results for 11 different model formulations. All results in the equally weighted outputs were generated after converting the objectives into a unitless scalar to improve the ability to compare values. The first six models are the multi-swarm PSO results with varying results due to changing the size of the population. The first four columns display the equally weighted multi-objective results. When comparing the results just from the MSPSO results, the population size does not appear to be significant. Heuristic 1 and Heuristic 2 are the slightly different greedy heuristics. The Combined MSPSO took all of the portfolio results from each of the MSPSOs and solved the optimization problem based on all of the options. The CG Price MSPSO model takes the column generation construct but solves the pricing problem using the MSPSO instead of having to solve the individual subproblems for each airport. CG Final is the full column generation solution using the standard column generation algorithm to achieve the optimal solution.

Table 3.4 - Summarized Model Results

	Model Results				Normalized Model Results			
	Obj 1	Obj 2	Obj 3	Total	Obj 1	Obj 2	Obj 3	Total
MSPSO 5	0.792	0.121	0.533	1.446	0.968	0.123	0.565	1.657
MSPSO 6	0.777	0.150	0.257	1.184	0.949	0.153	0.0	1.102
MSPSO 7	0.620	0.070	0.432	1.122	0.743	0.071	0.358	1.171
MSPSO 8	0.793	0.098	0.441	1.332	0.970	0.100	0.376	1.446
MSPSO 9	0.513	0.001	0.614	1.128	0.602	0.0	0.730	1.333
MSPSO 10	0.684	0.059	0.628	1.370	0.826	0.060	0.758	1.644
Combined MSPSO	0.676	0.475	0.734	1.885	0.816	0.485	0.976	2.278
Heuristic 1	0.603	0.480	0.746	1.828	0.720	0.490	1.0	2.210
Heuristic 2	0.598	0.467	0.696	1.761	0.713	0.477	0.898	2.089
CG Price MSPSO	0.535	0.519	0.675	1.729	0.631	0.531	0.855	2.016
CG Final	0.816	0.978	0.645	2.439	1.0	1.0	0.793	2.793

The triangle radar plot in Figure 1 displays the normalized results. What we look for in the radar plot is for the colored lines to reach as close to 1 in each corner as possible. If the model lines reach 1, then the objective has reached the maximum value amongst the various models. If a color is barely registering, then the objective value result was basically inconsequential in comparison.

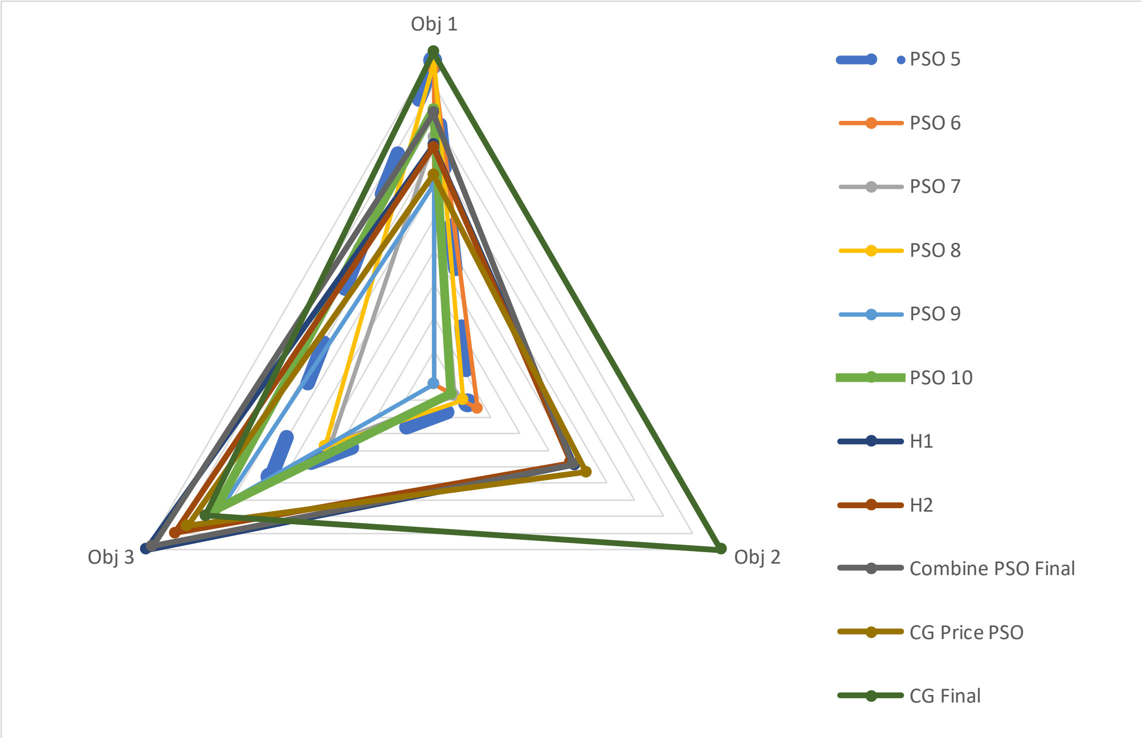

Figure 3.1 - Triangle Radar Plot, Performance Metric Comparison

These scenarios allow us to observe how security measure allocations differ when varying the number of inputs into the overall model. This technique gives us insight into determining whether or not it is beneficial to dedicate the time to find an optimal solution. The PSO methods take the least amount of time by far, and if the solutions are potentially just as strong, then it is possible that they can be utilized regularly. The decision-makers are also able to witness multiple options and consider what results remain consistent throughout the runs or what results change drastically depending on the model.

Model 1 (Table 3.5) displays the results of multi-swarm PSO with five particles.

Table 3.5 - Model 1: Multi-Swarm PSO – 5 particles

	PSO - 5																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	2	0	19	22	15	0	22	15	1	1	0	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	0	14	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	24	0
3	39	2	0	13	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
4	30	2	0	11	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
5	25	2	0	11	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
6	23	2	0	11	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
7	20	2	0	10	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
8	16	2	0	8	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
9	16	2	0	9	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
10	16	2	0	8	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0

Model 2 (Table 3.6) displays the results of multi-swarm PSO with six particles.

Table 3.6 - Model 2: Multi-Swarm PSO – 6 particles

	PSO - 6																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	2	0	19	0	15	48	22	15	1	0	3	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	0	14	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	24	0
3	39	2	0	13	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
4	30	2	0	11	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
5	25	2	0	11	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
6	23	2	0	11	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
7	20	2	0	10	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
8	16	2	0	8	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
9	16	2	0	9	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
10	16	2	0	8	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0

Model 3 (Table 3.7) displays the results of multi-swarm PSO with seven particles.

Table 3.7 - Model 3: Multi-Swarm PSO – 7 particles

	PSO - 7																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	2	2	19	0	15	48	22	15	1	1	0	15	15	15	19	15	19	0	22	22	0	0	0	37	0
2	42	2	2	14	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	24	0
3	39	0	2	13	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	22	11	0	0	0
4	30	2	2	11	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
5	25	2	2	11	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
6	23	2	2	11	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
7	20	2	2	10	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
8	16	2	2	8	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
9	16	2	2	9	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0
10	16	2	2	8	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	0	0	0	21	0

Model 4 (Table 3.8) displays the results of multi-swarm PSO with eight particles.

Table 3.8 - Model 4: Multi-Swarm PSO – 8 particles

	PSO - 8																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	0	2	19	0	15	48	22	15	1	0	3	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	2	14	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	24	0
3	39	0	2	13	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
4	30	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
5	25	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
6	23	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
7	20	2	2	10	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
8	16	2	2	8	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
9	16	2	2	9	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0
10	16	2	2	8	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	21	0

Model 5 (Table 3.9) displays the results of multi-swarm PSO with nine particles.

Table 3.9 - Model 5: Multi-Swarm PSO – 9 particles

	PSO - 9																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	0	0	19	0	15	48	22	15	1	0	0	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	2	14	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	9	0	0	0
3	39	2	2	13	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
4	30	2	2	11	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
5	25	2	2	11	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
6	23	2	2	11	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
7	20	2	2	10	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
8	16	2	2	8	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
9	16	2	2	9	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
10	16	2	2	8	0	14	44	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0

Model 6 (Table 3.10) displays the results of multi-swarm PSO with ten particles.

Table 3.10 - Model 6: Multi-Swarm PSO – 10 particles

	PSO - 10																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	2	2	19	22	15	0	22	15	1	0	0	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	2	14	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	0	27
3	39	2	0	13	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	11	0	0	0
4	30	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	0	21
5	25	2	0	11	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	11	0	0	0
6	23	2	0	11	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	11	0	0	0
7	20	2	0	10	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	11	0	0	0
8	16	2	2	8	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	0	21
9	16	2	2	9	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	0	21
10	16	2	2	8	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	0	0	0	21

Model 7 (Table 9) displays the results of the heuristic 1.

Table 3.11 - Model 7: Heuristic 1

	Heuristic 1																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	2	2	19	0	15	48	22	15	1	1	0	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	2	14	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	24	0
3	39	2	2	13	0	14	44	20	14	1	1	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
4	30	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
5	25	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
6	23	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
7	20	2	2	10	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
8	16	2	2	8	20	14	0	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
9	16	2	2	9	20	14	0	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
10	16	2	2	8	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0

Model 8 (Table 10) displays the results of the heuristic 2.

Table 3.12 - Model 8: Heuristic 2

	Heuristic 2																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	2	2	19	0	15	48	22	15	1	1	0	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	2	14	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	24	0
3	39	2	2	13	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
4	30	2	2	11	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
5	25	2	2	11	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
6	23	2	2	11	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
7	20	2	2	10	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
8	16	0	2	8	20	14	0	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
9	16	2	2	9	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
10	16	2	2	8	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0

Model 8 (Table 10) displays the results of the column generation.

Table 3.13 - Model 9: Column Generation

	Final Model																									
	Security Measures																									
Airports	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
1	49	1	2	19	0	15	48	22	15	1	1	0	15	15	15	19	15	19	0	22	22	24	0	0	37	0
2	42	2	2	14	0	14	44	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	24	0
3	39	2	2	13	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
4	30	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
5	25	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
6	23	2	2	11	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
7	20	2	2	10	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0
8	16	2	2	8	20	14	0	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
9	16	2	2	9	20	14	0	20	14	1	0	0	14	14	14	19	14	18	0	21	21	22	0	0	21	0
10	16	2	2	8	20	14	0	20	14	1	0	3	14	14	14	19	14	18	0	21	21	22	11	0	0	0

3.11 Conclusion

3.11.1 Summary

The classical portfolio optimization model uses the variance as the risk measure and relies on the covariance matrix. Without reliable estimates for the covariance/correlation, we needed to adapt network analysis techniques to make a pseudo correlation matrix. We constructed a network of interdependent risk factors that can be represented by a weighted adjacency matrix (Ravasz et al., 2002). This matrix is then combined with the TOM, similarity measure for biological networks (Yip & Horvath, 2007) and (Agliari et al., 2015), to define and quantify the topological and interdependent relationships between the security measures and the risk factors.

As a means to integrate the risk factors and security measures that are put in place by TSA, the method to calculate Risk Posture is developed. Risk Posture is calculated based on the optimal security measure portfolios selected and their interdependent relationship with the TSA risk taxonomy. With Risk Posture, we are trying to maximize the resilience of the system so that no matter the risk, we should be able to face it. There are no standard Risk Posture calculations, and the term has been typically associated with Cyber-security readiness. Our goal is to maximize the overall improvement in risk posture by minimizing our risk.

Nearly all security measures have been addressed in small groupings in past research over the past 16 years, but none all together in an optimization model. Stewart and Mueller (Stewart & Mueller, 2017) are the only publication/s that include all security measures. No prior optimization model has attempted to incorporate multiple screening areas into a single model. We were able to incorporate Stewart and Mueller's (Stewart & Mueller, 2017) reliability construct to include Checked baggage, Carry-on baggage, and Passenger screening. ERM portfolio optimization models are typically tied to the Insurance and Finance industries and follow a very traditional modeling approach. Sewell's SADM model and Nie's model are sub-models within our overall resource allocation model. This new model expands upon previous research and combines all models into a resource allocation optimization model with a new primary objective concentrated on Risk Posture.

The output of the model allocates limited quantities of security measures/screening devices across airports nationwide to

- Minimize the probability of false clears
- Maximize the total security level (probability of threat detection)

• Maximize the risk posture of the TSA (threat detection capability concerning the interdependent network of TSA risk elements)

3.11.2 Conclusions

The biggest knowledge gap in the research is any type of optimization model concerning enterprise risk management performs at an operational level. This would be the first model that covers a full multi-tier enterprise risk management (strategic, tactical, and operational levels). This would also be the first model to concentrate on risk posture. The device allocation problem, combined with a passenger risk assessment policy, can be used to structure a risk-based screening strategy to use limited screening resources effectively. The model continues to be very flexible and can comfortably accommodate different resources, new constraints, and additional objectives

This chapter extends the work by (Sewell et al., 2013) and (Nie et al., 2009) to present an optimal solution methodology for solving the security screening device allocation model across multiple airports. Given budget constraints, including the installation, operation, and fixed costs associated with screening devices and procedures in an airport checkpoint, the purpose of this model is to allocate limited quantities of new screening technology across airports nationwide to maximize the total security level (i.e., probability of threat detection) over all the airports under consideration. To accomplish this, we compute a risk factor for security classes using either the new or existing detection devices, based on the hourly throughput rate of each of the device types and the perceived risk of the passengers. The passenger risk is obtained using a prescreening system such as

CAPPS and allows security operations to partition passengers into high or low-risk categories for undergoing higher or lower intensity screening, respectively. This chapter presents a Dantzig-Wolfe decomposition approach to the nonlinear problem, where optimal solutions are shown to be obtained in several seconds through several computational examples. The device allocation problem, combined with a passenger risk assessment policy, can be used to structure a risk-based screening strategy that makes effective use of limited screening resources.

CHAPTER 4. CBP RESOURCE ALLOCATION

4.1 Introduction

The large influx of immigrants across the US-Mexico border has severely strained the government's capacity to handle border safety and protection. The situation is further exacerbated with thousands of immigrant children separated from their parents or family members and being held at Border Patrol facilities. Although there have been numerous debates regarding strategies and policies in securing border safety and in mitigating the risks and danger migrants go through to arrive in the United States in search of a better future, an effective unifying theme of border security and operational infrastructure has not materialized.

Immigration and security along the southern border have long been a topic of discussion and is a well-publicized struggle. We acknowledge that there are social and humanitarian issues present in this area of research, and we are exploring some of these in additional research papers. We, the authors, do care about illegal immigrants and their physical and social well-being, but this is not a political paper. This research focuses on constructing a mathematical model that aids the government in how they spend their budget on selecting resources for the U.S. operational security infrastructure. The simple fact is the U.S. does have a southern border where hundreds of thousands of immigrants attempt to enter the country illegally. The popular proposition is to construct additional wall segments and update existing structures along the border. This paper presents a mathematical model that will assist the CBP in determining whether the wall is a prudent investment to the safety and security of all people involved or if there are security investments that can aid in detection and aid in non-physical deterrence.

There is a large amount of academic research studying security between ports of entry, but almost none of it is mathematical. There is a host of research that describes security between ports of entry, and more than anything, it focuses on the leveraging of mortal danger the migrants face in navigating remote wilderness locations as the prime mechanism of deterrence. The leveraging of mortal danger as deterrence was an explicit part of Border Patrol's deterrence-based strategic planning. In large part, what brought the most considerable academic attention to a deterrence-driven border policy was that the ratio of deaths to CBP migrant apprehensions skyrocketed and had been steadily increasing into the latter 2010s, even as projected migration rates declined (Chambers, 2019). The misconception with non-mathematical usage and explanations of these numbers is that migrant deaths have been increasing for years across the border. The truth is the annual number of deaths is approximately 21% lower than the 20-year average, 26.5% lower than the number of deaths ten years ago, and 7.6% higher than the number of deaths 20 years ago (U.S. Border Patrol Southwest Border Deaths by Fiscal Year, 2019). The drastic difference/increase in the ratio of deaths to apprehensions is due to the 75% decrease in apprehensions in the last 18 years (U.S. Border Patrol Total Illegal Alien Apprehensions By Fiscal Year). It is correct that the number of deaths has increased in specific sectors due to the "Funnel Effect" or avoidance of enhanced border surveillance technology in other sectors, but the rates also drastically decreased in the monitored areas (Chambers, 2019). Overall, this tells us that deterrence-based strategic efforts are possibly contributing to the reduction in attempted illegal crossings.

There are arguments that deterrence based strategies increase injuries to immigrants. The literature presents the scenarios where immigrants attempt to cross the border over the large border fences and have injured themselves from falling off of them (Jusionyte, 2018). It is evident in the results presented in this paper that spending additional funds on the wall is not the best solution and that other more effective and less physical methods can deter immigrants. Obviously, any loss of life is tragic, but our mathematical model can be used to encourage funding of deterrence and detection methods, even in remote areas. Not only would this aid in decreasing migrant attempts in the dangerous routes, but it would also assist CBP agents and first responders in assisting those individuals that are injured in their crossing attempts.

The US-Mexico border spans approximately 1,933 miles long. As a result of the Secure Fence Act in 2006, hundreds of miles of physical fence were constructed along the border. Currently, 1,279 miles, 66% of the border is unfenced, with the Rio Grande River making up much of this unfenced border. The current position of pedestrian fence and vehicle barriers can be seen in Figure 4.1 - U.S. Mexico Border (Mark & Kiersz, 2019).

In the study, we present a dynamic systems modeling approach to analyze how best to establish effective border strategies in deterrence and detection through optimal security measures investment.

Figure 4.1 - U.S. Mexico Border

Within the field of portfolio investment research, there is a severe gap in incorporating ERM within the operational level. Chapter 3 analyzed strategies for security measure allocation for optimal aviation security and incorporates a computational framework for multi-tier risk taxonomy modeling and strategic assessment. In this chapter, we leverage the modeling framework to tackle the borders, taking into account the full multi-tier enterprise risk management (strategic, tactical, and operational levels).

4.2 CBP Prior Work

Little has been reported in the literature regarding the measures of the effectiveness of existing border security. Merely looking at the number of apprehensions could be very misleading. A decrease in the number of apprehensions could indicate either successful border enforcement or failed border enforcement. Success could be due to rising deterrence and fewer attempts. Failure could be due to more successful illegal entries. Hence some robust estimates of the likelihood that an unauthorized border crosser will be stopped and detained are much needed. Similarly, knowing what type of security measures are responsible for the identification of border crossers being apprehended is essential. Our objective is to establish a combination of security measures by sector to increase the apprehension rate or deterrence rate.

The current state of enforcement is a work in progress. Arizona is the first state to experience technology upgrades at its border. The original upgrade plans called for 52 Integrated Fixed Towers, underground sensors, night vision scopes for trucks, and remote video surveillance systems.

This study establishes a mathematical model that supports border security and includes both physical and technological security measures. The model determines the best combination of security measures based on their detection characteristics and capability (and potentially other factors). Security measures can be tailored to each sector and an area of coverage based on the average number of apprehensions per month and physical attributes of each sector. This could be further tailored to specific station requirements. The apprehension rates used in the model (Table 4.1) are based on the FY18 statistics. The monthly average sector rates are used to determine the quantity of each security measure required to ensure 100% apprehension.

Table 4.1 - Total Apprehensions by Sector per Month (U.S. Border Patrol Southwest Border Apprehensions by Sector (2018))

						Total Apprehensions								
Sector	**FY 2018**	**Oct**	**Nov**	**Dec**	**Jan**	**Feb**	**Mar**	**Apr**	**May**	**June**	**July**	**Aug**	**Sept**	**Avg**
Big Bend	8,045	819	828	802	543	838	703	808	743	375	456	585	545	670.42
Del Rio	15,833	1,046	1,186	1,113	1,083	1,306	1,466	1,451	1,486	1,462	1,365	1,506	1,363	1319.4
El Centro	29,230	2,194	2,123	2,110	2,052	1,954	2,697	2,790	2,683	2,327	2,531	2,821	2,948	2435.8
El Paso	31,561	1,489	1,647	1,713	1,607	1,737	2,782	2,671	3,510	3,560	2,890	3,585	4,370	2630.1
Laredo	32,641	2,451	2,283	1,982	2,296	2,671	3,652	3,370	3,210	2,586	2,600	2,785	2,755	2720.1
Rio Grande	162,262	9,722	11,726	11,668	9,484	9,611	14,140	15,993	17,491	14,703	13,238	16,744	17,742	13522
San Diego	38,591	2,377	2,760	2,764	3,171	3,107	4,101	3,644	3,418	3,014	3,098	3,507	3,630	3215.9
Tucson	52,172	3,854	4,562	4,400	3,925	3,824	5,785	5,012	4,760	4,146	3,241	3,627	5,036	4347.7
Yuma	26,244	1,536	1,970	2,443	1,814	1,618	2,064	2,504	3,038	1,916	1,880	2,364	3,097	2187
USBP SW Border Total	**396,579**	**25,488**	**29,085**	**28,995**	**25,975**	**26,666**	**37,390**	**38,243**	**40,339**	**34,089**	**31,299**	**37,524**	**41,486**	

There is very little academic research about border security between immigration ports. Bristow provided the only model that analyzed the border wall in Arizona (Bristow, 2017). The model focuses on infrastructure on the Arizona border and how to decide to upgrade infrastructure based on current effectiveness levels. There is a severe lack of mathematical models developed to support border decision-making strategy. It is a timely opportunity to analyze resource allocation across all sectors holistically to maximize global effectiveness.

4.3 Our Contributions

A quantitative construct for optimizing security measure investments is established to achieve the most cost-effective deterrence and detection capabilities for the CBP. A large-scale resource allocation optimization integer program was successfully modeled that rapidly returns good Pareto optimal results. The model incorporates the utility of each measure, the probability of success, along with multiple objectives. To the best of our knowledge, our work presents the first mathematical model that optimizes security strategies for the CBP and is the first to introduce a utility factor to emphasize deterrence and detection impact. The model accommodates different resources, constraints, and various types of objectives. The solution methodologies being put in place are complex, current state-of-the-art, and very effective.

We leverage our recent multi-objective resource allocation model developed for TSA airport security analysis (Leonard, Lee, Booker, 2019). Specifically, we introduced a large-scale integration and expansion of the work by Nie et al. and Sewell et al. (Nie et al., 2009, Sewell et al., 2012, Sewell, et al., 2013). The systems TSA model determines an optimal allocation of threat detection devices and measures for screening checked baggage, carry-on baggage, and passengers across a set of airports so as to 1) maximize risk posture, 2) maximize the number of threats detected, and 3) minimize the overall false alarm rate while considering passenger threat classification. Constraints are imposed on the time available at each check station, flow capacity at security stations, budget, as well as staffing needs at each check station. We employ the TSA construct for the CBP border security model herein but with a primary objective of maximizing the utility of the security measure portfolios employed in each sector of the border wall.

The chapter is organized as follows. Section "Mathematical Model" presents the CBP system resource allocation problem as a multi-objective nonlinear integer program. Section "Solution Methodology" first describes a Dantzig-Wolfe decomposition approach to handle the nonlinear constraints and objective and the binding constraint of allocating resources across the sectors. It also describes a column generation approach implemented to solve the model to optimality directly. Section "Computational Results" reports empirical results from several problem instances to demonstrate the Pareto optimal solutions and their respective trade-offs.

4.4 Requirements Analysis

We model the U.S. Customs and Border Protection ERM in 3 tiers. Tier 3 is comprised of satellites monitoring the geographic area of the border. This allows for 24/7 surveillance and data gathering, pinpointing high-frequency crossing areas, and addressing vulnerable locations. Tier 2 employs High Altitude Long Endurance (HALE) drones with high fuel capacity for extended surveillance. They provide higher resolution images compared to Tier 3 and are equipped with infrared capabilities to find hidden smuggling camps, etc. Tier 1 is the ground layer, which includes a variety of security surveillance systems and manned outposts. This operational layer is equipped with quadcopter drones, intermittent outposts along the border, and sensor technology in between. The outposts serve as command posts for drone swarms and also as home bases for analyzing information streams from all tiers. The sensors can identify border crossings as well as attempted tampering with existing wall structures and the ground below. Swarms of drones can be sent out as quick response teams to identify crossers further or interdict them. An illustrated diagram of the surveillance tiers, as seen below in Figure 4.2 (Lee E. K., 2019).

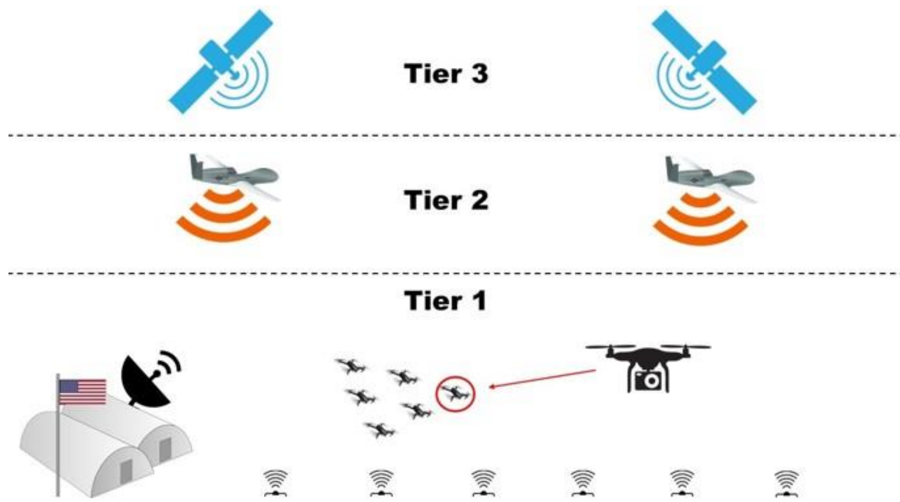

Figure 4.2 - A Three Tier Security Capability Architecture

CBP has made positive strides in protecting the border. However, multiple challenges remain. A critical element is to utilize the border agents' time more efficiently and effectively. The number of CBP agents has grown several times over in the past two decades, but the quantity of manpower is never quite enough. The requirements of a physical agent presence for identification, verification, or detection are continuously straining the manpower resources. Some of this pressure can be alleviated by employing "smart" security measures. For example, false alarms could be identified by drones or surveillance equipment. Risk levels and priority scores can be dynamically generated to better allocate different resources on a day to day basis with real-time information coming from the "Smart Wall." With the use of surveillance technologies and drones, illegal immigrants can be deterred from illegal crossings non-violently. Drones play a big part in surveillance, but can also be used to scare away potential smugglers. The loud sound of the rotors and drones flying low-overhead as well as noise emitting technology could be used to scatter or deter groups of smugglers. Detecting “unseen” threats are much easier with the use of advanced technology. Tunneling has become a common smuggling method and can go undetected until actual contraband reaches the other side of the border. With smart, highly sensitive sensors, the consistent vibrations of digging could be detected and separated from the interference (such as animals) using pattern detection technology. Agents can be alerted in real-time, allowing for pre-emptive security. 24/7 information flow will enable agents to monitor and learn smugglers' patterns. It may even be possible to determine the patterns of smugglers/smuggling (favorite combinations of routes, time of day, weather, etc.) through machine learning to be steps ahead.

In determining requirements, Tier 3 is omitted and assumes satellite systems are already in place. Tier 2 includes military-grade surveillance technology. For example, the cost of acquisition of HALE drones, such as the MQ-4C Triton, is roughly 20 million each. Tier 1 requirements include commercial drones. Commercial drone technology is developing at a rapid rate, and top of the line is constantly changing. We use the DJI Mavic Air as the base drone for our analysis. This drone has an effective range of 6.2 miles and costs 00 each. With a border length of ~1,934 miles long and an effective drone mission range of 6.2 miles, an outpost to act as the command center for these drones will need to be placed approximately every 12.4 miles to ensure 100% coverage. This leads to about 162 outposts along the border, which would need to be staffed accordingly. Unattended ground sensors would be located between outposts and would send alerts when suspected crossing or border tampering, such as digging under existing barriers, occurs.

CBP's primary objective is border security. There are over 60,000 employees, a third of which are border patrol agents. The southern continental border includes the border states of California, Arizona, New Mexico, and Texas. The border is split into nine "sectors," which are divided into 74 "stations," with each containing specific patrol zones, Figure 4.3, (Office of the Inspector General(1), 2017). In the current state, CBP only has effective (physical) control of 680 miles of the border, while the Rio Grande River serves as a natural barrier of over 1,000 miles, (S&T Impact: Borders & Ports of Entry).

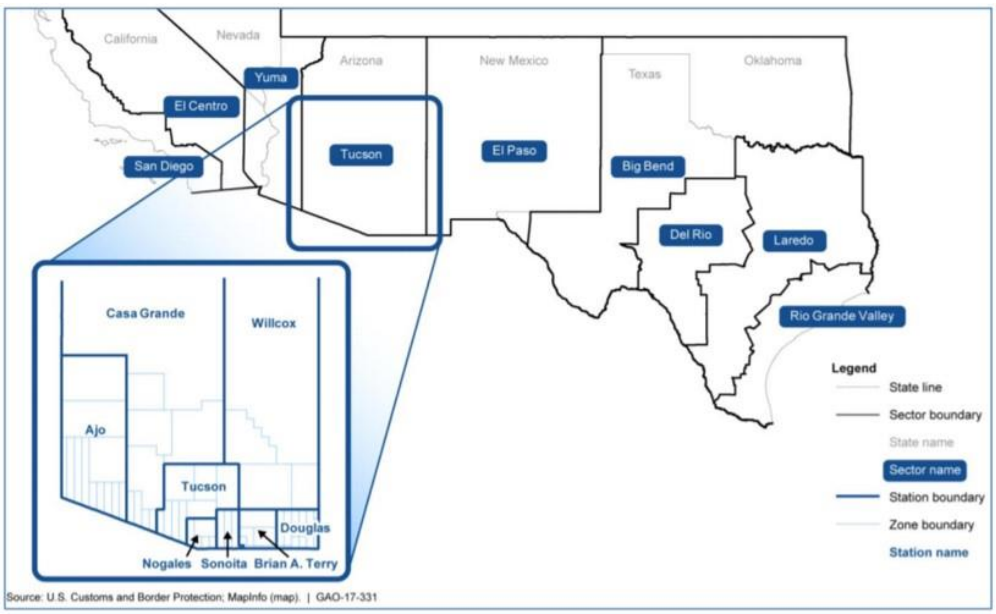

Figure 4.3 - Map of the Nine Border Sectors along the US-Mexico Border

CBP annually collects data from frontline border patrol agents and chiefs from each of the nine patrol sectors along the Southwest border. There are roughly 500,000 illegal entries per year. The data collected identifies vulnerability or "capability gaps." CBP then catalogs preliminary requests for solutions to address capability gaps that include infrastructure, technology, personnel, etc. The CBP Wall Decision Support Tool (WDST) identifies the relative priority of various segments along the border for the proposed border wall. The inputs are based on the feedback from the sector chiefs. (Committee of Homeland Security & Governmental Affairs (2018))

Focusing on CBP's highest priority vulnerabilities across all capability gaps (902 total), we observe the following key findings (Committee of Homeland Security & Governmental Affairs (2018))

- Less than 0.5% of the proposed solutions from CBP agents and sector chiefs included a request for a "wall."
- Less than 4% of proposed solutions from CBP agents and sector chiefs included a request for additional "fencing."
- Only one "Urgent and Compelling" request (out of 14) mentioned either a wall or fencing.
- 25% of vulnerabilities can be addressed using the man-made infrastructure of any kind.
- The remaining 75% indicate the need for technology and personnel approaches to advance border security.

The report presents a uniform opinion of CBP personnel's desire to integrate technology along the entire border to advance and improve border security. The physical border wall does exist and has been in place for many years, with some areas being modernized in the last five to seven years. The cost of upgrading the remaining legacy fence is cost-prohibitive at an average of .494 million per mile and would exhaust all available funds (Office of the Inspector General[(1)] (2017)). The remaining two-thirds of the open border contains terrain where technology is much more useful.

A modular multi-layered (tiered) system is desirable for achieving operational efficiency and strategic gains. Adding new technology and supplementing existing technology will facilitate this system's approach. New technology includes UAVs, tethered drones, unattended ground sensors, infrared detection, surveillance systems. They have generated a shift in security tactics that many believe can be very beneficial. (Office of the Inspector General (2017))

In this study, we apply risk-based modeling to determine the most cost-effective security measure investments. The effectiveness is based on reducing the likelihood of attack (increasing detection and/or deterrence). Such models empower policymakers to make sound and informed decisions in allocating funds. (Lavender, 2017)

4.5 Data Collection

Geographically, approximately 90% of the primary border fencing on the SW border is in the five western-most sectors, with the remaining 10% of primary fencing located in the four eastern-most sectors where the Rio Grande River delineates the majority of the border. The current percentage estimates of legacy and modern fencing are shown in Figure 4.4 below (Office of the Inspector General(1), 2017).

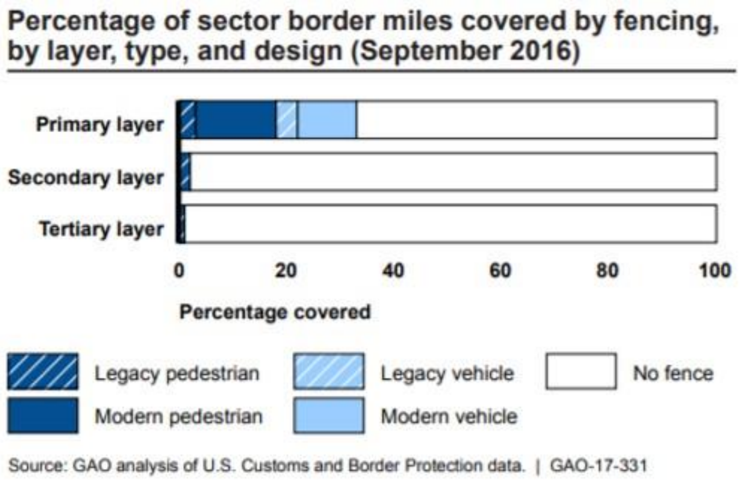

Figure 4.4 - Existing US-Mexico Border Fencing

CBP employs a system of interconnected security layers to deter and detect illegal immigration and criminal activity. Table 4.2 summarizes a sample list of existing security measures and the estimated values across the nine sectors. Some of these values are derived from existing documentation, while others are estimates based on public announcements of new installations (Office of the Inspector General[(1)] (2017)). Specifically, the physical and Tier 3 security measures are known quantities and locations. Tier 2 and Tier 1 security measures do exist; however, their exact values or positions are uncertain. We demonstrate the use of our system by inputting the Tier 3 measures and allowing the model to determine the initial purchase and assignment quantities across the sectors and stations. Other security measures include aircraft, UAVs, etc. with estimated values/locations/cost.

Table 4.2 - CBP Sample Existing Security Measures

Tier	Security Measure\Sector	California		Arizona		Texas					Purchase Cost per Item (in Millions)
		1	2	3	4	5	6	7	8	9	
		El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley	
Physical	New Bollard Wall (miles) (30 ft tall)	42	12	144.1	98.28	0	4.2	90	0	54.6	$ 5.50
	Primary Fence (miles) (10-20 ft tall)	16.1	33	65.5	8.82	2.55	0	18	0.855	0	$ 2.00
	Secondary Fence (miles)	0	13.2	0	10.08	2.55	0	14.4	0	0	$ 1.00
	Tertiary Fence (miles)	0	1.2	0	10.08	0	0	1.8	0	0	$ 0.50
Tier 3	Tactical Aerostats	0	0	1	1	1	1	1	0	1	$ 8.90
Tier 2	Commercial Drones	0	0	0	0	0	0	0	0	0	$ 0.00
	IFT	0	0	0	3	0	0	0	0	0	$ 18.48
	Remote Video Surveillance System (RVSS)	0	0	0	0	0	0	0	0	0	$ 4.42
Tier 1	RVSS Upgrades	0	0	0	0	0	0	0	0	0	$ 4.42
	Mobile Surveillance Capability (MSC)	0	0	0	0	0	0	0	0	0	$ 2.23
	Agent Portable Surveillance System (APSS)	0	0	0	0	0	0	0	0	0	$ 0.77
	Mobile Video Surveillance System (MVSS)	0	0	0	0	0	0	0	0	0	$ 3.15
	Thermal Imaging Device (TID)	0	0	0	0	0	0	0	0	0	$ 0.33
	Unattended Groud Sensors (UGS)	0	0	0	0	0	0	0	0	0	$ 0.00
	Imaging Sensors (IS)	0	0	0	0	0	0	0	0	0	$ 0.00

The border surveillance systems are comprised of combinations of surveillance technologies that are designed or utilized to assist the CBP in enforcing U.S. laws and to detect, identify, apprehend, and remove persons and illegal contraband. Since the 2014 BSS assessment, CBP has deployed new technologies, including mobile, fixed, and other technologies. (Luck (2018))

Mobile surveillance technology includes Tactical Aerostats (TAS), lightweight Counter-Mortar Radar/Lightweight Surveillance and Target Acquisition Radar (LCMR/LSTAR), and Man-Portable Aerial Radar System-Kits (MARS-K). (Luck (2018)). The TAS units are significant to the program and provide a low-cost, low-flying satellite system. The TAS units include wireless transmitters, are capable of detecting all aircraft within a 200-mile range, and all data is downloaded and integrated to the Air and Marine Operations Center (AMOC). (Long) The LSTAR radars provide 360 degrees 3D electronic scanning capabilities for detecting and tracking airborne targets. The LCMR systems provide continuous 3D 360-degree surveillance and 3D rocket, artillery, and mortar location using a non-rotating electronically steered antenna. Integrated Fixed Towers (IFTs) are integrated with the Tracking and Signcutting Modeling (TSM). The IFTs include day and night cameras, radar, and laser illuminator sensors that can be monitored from local sector facilities. (Luck (2018))

Other surveillance technology includes the cross-border tunnel threat (CBTT) program, the border tunnel activity detection system-point (BTADS-P), linear ground detection systems (LGDS), and unattended ground sensors (UGS). The CBTT program employs tunnel detection technology to enhance the tunnel activity monitoring capabilities. It is a network of subterranean ground sensors collecting seismic information that includes people walking near the border, climbing over fences, digging near the sensors, vehicles or animals near the border, and low flying aircraft. The BTADS-P, LGDS, and UGS are all different types of sensors used in the CBTT that are useful for detecting when a tunnel is actively being constructed or provide long-term physical intrusion detection. (U.S. Customs and Border Protection (2015))

4.6 Mathematical Model

We establish here the mathematical programming model to determine the optimal allocation of security devices and measures such that we can maximize the utility of the applied portfolio, maximize the probability of detection, and minimize cost.

To construct this model, we leveraged the same resource allocation model developed for TSA airport security analysis in Chapter 3. Specifically, we introduced a large-scale integration and expansion of the work by Nie et al. and Sewell et al. (Nie et al., 2009; Sewell et al., 2012; Sewell et al., 2013). The systems TSA model determines an optimal allocation of threat detection devices and measures for screening checked baggage, carry-on baggage, and passengers across a set of airports to 1) maximize risk posture, 2) maximize the number of threats detected, and 3) minimize the overall false alarm rate while considering passenger threat classification. Constraints are imposed on the time available at each check station, flow capacity at security stations, budget, as well as staffing needs at each check station. We employ the TSA construct for the CBP border security model herein. Specifically, a single large scale mixed integer programming portfolio optimization problem is constructed with a primary objective of maximizing the utility of the security measure portfolios employed in each sector of the border wall.

The parameters and decision variables used in the model are as follows:

Parameters	**Description**
T	The total number of border sectors
k	Index for border sector $k=1,2,\ldots,T$
d	Index for detection device type $d=1,2,\ldots,D$
B_k	Number of apprehensions to resolve per month by sector k
I_d	Installation cost ($/device) associated with security measure type d
C_d	Maximum throughput (apprehensions/month) of security measure type d
E_{dk}	Number of existing devices of security measure type d at sector k
K_{dk}	Capacity of the quantity of security measure type d at sector k
P_d	Conditional probability of detecting a threat given there is a threat for device type d
U_d	Number of device type d available for installation

Decision Variables	**Description**

x_{dk}	Binary variable where $x_{dk} = 1(0)$ if security measure type d is (not) used to deter/detect apprehension station k
y_{dk}	Number of security measure type d to be used at sector k (integer)
s_{dk}	Number of security measure type d to be installed at sector k (integer)

Our constraint development begins with assigning the device types, d, and numbers, y_{dk}, of detection devices to each sector in constraint (4.1). To accomplish this, the number of devices of type d to be installed in each sector, s_{dk}, is found by subtracting the number of devices of type d currently existing from the number of devices of type d used in total at each airport, in constraint (4.2). Therefore,

$$y_{dk} = \left\lceil \frac{B_k x_{dk}}{C_d} \right\rceil, \qquad \text{(ResourceCapacityConstraint)} \qquad (4.1)$$

$$s_{dk} = \max\{y_{dk} - E_{dk}, 0\}, \qquad \text{(DeviceInstallationConstraint)} \qquad (4.2)$$

for $d = 1,2,\dots,D$ and $k = 1,2,\dots,T$.

The sector installation budget constraint, (4.3), can be found by summing up the installation costs of each security measure at each sector, and verifying the total sum is less than or equal to the overall installation budget.

$$\sum_{k=1}^{T}\sum_{d=1}^{D} I_d s_{dk} \le \text{Budget}, \qquad \text{(SectorInstallationBudgetConstraint)} \qquad (4.3)$$

Next, the number of new security measures installed in all sectors must be less than or equal to the total number of new devices available (4.4), and so, the device resource availability constraint becomes

$$\sum_{k=1}^{T} s_{dk} \le U_d, \qquad \text{(ResourceAvailabilityConstraint)} \qquad (4.4)$$

for device type $d = 1,2,\dots,D$.

Lastly, the sector resource capacity (4.5) is defined by the number of new security measures less than or equal to the number of available billets within each sector.

$$s_{dk} \leq K_{dk},\ \forall\ d \in D \text{ and } k \in T \qquad \text{(Sector Resource Constraint)} \tag{4.5}$$

There are three objectives. The first (primary) objective is to maximize the utility of the applied portfolios (4.6). T_i is the utility value of each sector that is equivalent to $\frac{(\#\text{Agents})\times(\text{BorderMiles})\times(\#\text{Stations})}{(\text{SqMiles})}$. This calculation is then normalized to prevent overly large objective values. The weighted, adjusted Risk Posture, covering all sectors, all countermeasures, and all risk areas is calculated as follows

$$\text{Utility} = \sum_{k=1}^{T} \sum_{d=1}^{D} x_{dk} T_k \tag{4.6}$$

The second objective is to maximize the total probability of detection (4.8) and is based on both the probability of a device correctly detecting a threat and the rate of apprehensions generated by each security measure. P_d is the conditional probability that a threat is detected by security measure type d, given that a threat is present. L_k, (4.7), is the probability that at least one of the security measures used detects the threat.

$$L_k = 1 - \prod_{d=1}^{D} (1 - x_{dk} P_d) \tag{4.7}$$

$$\text{Probability of Detection} = \sum_{k=1}^{T} L_k \tag{4.8}$$

The third objective, (4.9), is to minimize cost. Even though there is a budget in place, the total number of dollars spent is still essential. Placing a limit on the budget allows for effective yet fiscally responsible portfolios to be selected.

$$\text{Cost} = \sum_{k=1}^{T} \sum_{d=1}^{D} I_d s_{dk} \tag{4.9}$$

In short, the multi-objective problem is defined by the objectives in (4.10)-(4.12)

$$z_1 = \max \text{Utility} = \max \sum_{k=1}^{T} \sum_{d=1}^{D} x_{dk} T_k \tag{4.10}$$

$$z_2 = \max \text{Probability of Detection} = \max \sum_{k=1}^{T} L_k \tag{4.11}$$

$$z_3 = \min \text{Cost} = \max \quad -1 \times \left(\sum_{k=1}^{T} \sum_{d=1}^{D} I_d s_{dk} \right) \tag{4.12}$$

4.7 Solution Methodology

The primary objective of this problem is to maximize the utility associated with improving the security posture of the border sectors. The utility improves by adding stronger security measures to a sector that sees larger rates of apprehensions on average and having a larger region of coverage with less manpower. Each security measure has a probability of detecting a threat, with the system as a whole having an overall threat detection probability. Since the system is layered, this is a conditional probability that at least one of the measures/devices in place will detect a threat given there is a threat. From here, a system reliability analysis can be performed with the intent of maximizing risk reduction or threat detection. Since all of the security measures/devices currently in use or proposed use are independent, this is modeled as a series system, as shown in (Leonard & Lee, 2020) and (Stewart & Mueller, 2017).

$$\text{Risk} = 1 - \begin{cases} \left(1 - P(Threat\ disrupted\ by\ physical\ security\ measure)\right) \\ \times \left(1 - P(Threat\ disrupted\ by\ surveillance\ device)\right) \\ \times \left(1 - P(Threat\ disrupted\ by\ maritime\ assets)\right) \\ \times \left(1 - P(Threat\ disrupted\ by\ air\ assets)\right) \\ etc. \end{cases} \quad (4.13)$$

Our problem is a mixed-integer program based on the three different decision variables. We can reduce the number of decision variables by eliminating y_{dk}, which are the total number of security measures to be put in place in each sector. $y_{dk} = \lceil B_k x_{dk}/C_d \rceil$ and $y_{dk} \in \mathbb{Z}^+$. We can relax the integrality requirement with the following steps.

- $C_d y_{dk} \geq B_k x_{dk} (1)$
- $C_d y_{dk} < B_k x_{dk} + C_d (2)$
- let $y'_{dk} \in \mathbb{R}$, then y'_{dk} satisfies both equations (1) and (2)
- Then $y'_{dk} = \frac{B_k x_{dk}}{C_d}$ $\forall\, d$ and k
- Now y_{dk} can be obtained directly from $y'_{dk} \leq y_{dk} \leq y'_{dk} + 1$

If y_{dk} is obtained directly, then we can also determine s_{dk}. However, we still want to decide how many security measures to purchase and distribute so we retain the decision variables s_{dk}.

Even though we are considering additional constraints, the foundation of the problem is a direct derivation of the TSA allocation model from Chapter 3. Now that we have reduced the decision variables to s, we can enumerate the combinations that satisfy the constraints and store them in a binary array. $X_k = x\text{isfeasibleforsector}k = x_{i1}^{kJ} =$

$\left(x_{11}^{kJ}, x_{12}^{kj}, \dots, x_{1D}^{kj}, x_{21}^{kj}, x_{22}^{kj}, \dots, x_{M_k D}^{kj}\right)$. We can then further define our y_{dk} arrays as $y^{kj} = \left(y_1^{kj}, y_2^{kj}, \dots, y_D^{kj}\right)$.

Define n_k as the number of feasible solutions for sector k. Then notice that all of the feasible solutions for sector k can be generated by generating all of the binary arrays for x_d^{kj} and then computing y_d^{kj}. Next, we will define a binary variable r_{kj} for each feasible solution for each sector, where r_{kj} = 1(0) if solution x_{kj} is (not) selected to be used in sector k. The master problem can now be written as the following binary integer program seen in the equations (4.14) - (4.21) below:

BORDER_RESOURCE_IP

Maximize

$$z_1 = \max \sum_{k=1}^{T} \sum_{j=1}^{n_k} T_k r_{kj} \tag{4.14}$$

$$z_2 = \max \sum_{k=1}^{T} \sum_{j=1}^{n_k} L_k r_{kj} = \max \sum_{k=1}^{T} \sum_{j=1}^{n_k} r_{kj} \left(1 - \prod_{d=1}^{D} (1 - x_{dk} P_d)\right) \tag{4.15}$$

$$z_3 = \min \text{Cost} \ = \max \quad -1 \times \left(\sum_{k=1}^{T} \sum_{j=1}^{n_k} \sum_{d=1}^{D} I_d s_{dk} r_{kj}\right) \tag{4.16}$$

subject to

$$s_{dk} \geq y_{dk} - E_{dk}, \qquad \forall d = 1,2,\dots,D, \quad k = 1,2,\dots,T \tag{4.17}$$

$$\sum_{k=1}^{T} \sum_{j=1}^{n_k} s_{dk} r_{kj} \leq U_d \qquad \forall d = 1,2,\dots,D \tag{4.18}$$

$$\sum_{j=1}^{n_k} s_{dk} r_{kj} \leq K_{dk} \qquad \forall d = 1,2, \dots, D, \;\; k = 1,2, \dots, T \tag{4.19}$$

$$r_{kj} \in 0,1 \tag{4.20}$$

$$s_{dk} \in Z^+ \tag{4.21}$$

Although the model has been decomposed, the two remaining decision variables are now being multiplied by one another in one of the objective functions as well as in multiple constraints. We will introduce a new decision variable into the model to be represented by the equation $z = s \times r$ where s is a positive integer variable, and r is binary. If s is bounded below by zero and above by any large value, M, then we can add the following constraints to the model:

- $z \leq M \times r$
- $z \leq s$
- $z \geq s - (1 - r) \times M$
- $z \geq 0$

We can then substitute any expressions of $s \times r$ within the model with the new integer variable z.

4.8 Empirical Results

The **BORDER_RESOURCE_IP** ((4.14) - (4.21)) was generated in Python 3.7.3 and solved with Gurobi 9.0. The Gurobi parameters were kept at their default values, apart from turning off the pre-solve option so that Gurobi would spend less time expanding the node structure. We conducted the majority of the computational experiments on a personal computer with an AMD Ryzen 7 quad-core processor, 3.8 GHz processor speed, and 16 GB of RAM. Sensitivity analysis was completed on the Georgia Institute of Technology High Throughput server cluster.

4.8.1 Scenario Analysis

We design multiple experiments to gauge the interplay and tradeoffs of the objective functions and the constraints. All initial models solved in this section were solved using the standard discrete programming method found in Gurobi. Herein, we report eight scenarios to contrast the outcome.

- Model 1: Maximize Utility (Obj. 1)
- Model 2: Maximize the Probability of Detection (Obj. 2)
- Model 3: Minimize Cost (based on a lower bound of $2.5 B) (Obj. 3)
- Model 4: Maximize Cost (based on an upper bound of $5 B) (Obj. 3)
- Model 5: Tri-Objective Model (Max Obj 1 & 2, Min Obj. 3)
- Model 6: Tri-Objective Model (Max Obj 1 & 2, Max Obj. 3)
- Model 7: Tri-Objective Model (Max Obj 1 & 2, Min Obj. 3 with lower weight)
- Model 8: Tri-Objective Model (Max Obj 1 & 2, Max Obj 3 with lower weight)

These scenarios allow us to observe how security measure allocations differ when different primary objectives are emphasized. We can also observe the tradeoffs – how different primary objectives impact the other objectives (positively or negatively). This allows the decision-makers to see multiple options and to consider what results remain consistent throughout the scenarios or what results change drastically depending on the focus.

Model 1 (Table 4.3) displays the results of maximizing the utility function (Objective 1). Maximizing the utility sees upgrading or installing a modern bollard wall in several sectors. Commercial drones, IFTs, and Imaging sensors are critical for surveillance.

Table 4.3 - Model 1 Results

	Maximizing Utility Function Only								
Security Measure\Sector	**El Centro**	**San Diego**	**Tucson**	**Yuma**	**Big Bend**	**Del Rio**	**El Paso**	**Laredo**	**Rio Grande Valley**
New Bollard Wall (30 ft)	19	48	39	21	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	0	0	0	3	1	0	2	2
Commercial Drones	11	9	23	18	74	31	29	25	37
IFT	9	0	38	18	74	0	39	0	0
Remote Video Surveillance System (RVSS)	4	0	13	6	26	0	14	0	0
RVSS Upgrades	4	0	13	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	27	35	0	0
Agent Portable Surveillance System (APSS)	0	3	0	5	0	9	12	0	11
Mobile Video Surveillance System (MVSS)	0	1	0	0	0	4	0	3	0
Thermal Imaging Device (TID)	0	0	3	0	0	0	0	3	5
Unattended Groud Sensors (UGS)	5	4	15	8	33	0	18	11	16
Imaging Sensors (IS)	26	22	95	46	187	77	98	63	92

Model 2 (Table 4.4) displays the results of maximizing the probability of detection (Objective 2). Maximizing the detection capability alone provides a lesser solution due to focusing strictly on probability values.

Table 4.4 - Model 2 Results

	Maximizing Probability of Detection Only								
Security Measure\Sector	El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley
New Bollard Wall (30 ft)	19	43	39	21	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	0	0	0	3	0	0	2	2
Commercial Drones	11	9	23	18	0	31	29	25	37
IFT	11	0	0	0	0	0	39	0	0
Remote Video Surveillance System (RVSS)	4	0	13	6	26	0	0	0	0
RVSS Upgrades	4	0	13	0	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	27	35	0	0
Agent Portable Surveillance System (APSS)	0	3	0	5	0	9	12	0	11
Mobile Video Surveillance System (MVSS)	0	0	0	0	0	4	0	3	0
Thermal Imaging Device (TID)	0	0	3	0	0	0	0	3	5
Unattended Groud Sensors (UGS)	5	4	15	8	33	0	18	11	16
Imaging Sensors (IS)	0	22	0	0	0	0	0	0	0

Model 3 (Table 4.5) reports the results of minimizing cost based on a lower bound of $2.5 billion (Objective 3). Minimizing cost alone places slightly more emphasis on upgrading or installing new portions of the bollard wall

Table 4.5 - Model 3 Results

	Minimizing Bounded Cost								
Security Measure\Sector	El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley
New Bollard Wall (30 ft)	18	48	39	21	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	1	0	0	3	1	0	2	1
Commercial Drones	8	9	23	18	74	31	29	25	37
IFT	11	0	0	18	32	0	0	0	0
Remote Video Surveillance System (RVSS)	4	0	0	0	26	0	0	0	0
RVSS Upgrades	4	0	0	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	26	0	0	0
Agent Portable Surveillance System (APSS)	0	0	0	0	0	0	11	0	11
Mobile Video Surveillance System (MVSS)	0	0	0	0	0	0	0	0	0
Thermal Imaging Device (TID)	0	0	0	0	0	0	0	0	0
Unattended Groud Sensors (UGS)	0	0	0	0	33	0	0	0	16
Imaging Sensors (IS)	0	0	0	0	0	0	98	46	0

Model 4 (Table 4.6) displays the results of maximizing cost based on an upper bound of $5 billion (Objective 3). Maximizing cost concentrates on remote and mobile surveillance systems versus introducing low-cost commercial drones.

Table 4.6 - Model 4 Results

	Maximizing Bounded Cost								
Security Measure\Sector	El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley
New Bollard Wall (30 ft)	19	48	39	20.99999784	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	1	0	0	3	1	0	2	2
Commercial Drones	9	0	0	0	0	0	0	0	37
IFT	11	0	38	18	74	0	39	0	0
Remote Video Surveillance System (RVSS)	4	0	13	6	26	0	14	0	0
RVSS Upgrades	4	0	13	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	27	35	0	0
Agent Portable Surveillance System (APSS)	0	0	0	0	0	1	0	0	0
Mobile Video Surveillance System (MVSS)	0	0	0	0	0	4	0	2	0
Thermal Imaging Device (TID)	0	0	0	0	0	0	0	0	4
Unattended Groud Sensors (UGS)	0	0	0	0	0	0	0	2	0
Imaging Sensors (IS)	0	0	0	0	0	0	0	0	0

Model 5 (Table 4.7) displays the results of the full triple-objective model while minimizing objective 3, using equal weights. The optimization, including minimal cost, concentrates resources on commercial drones, and IFTs.

Table 4.7 - Model 5 Results

	Tri-Objective Model (Min Cost)								
Security Measure\Sector	El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley
New Bollard Wall (30 ft)	18	48	39	21	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	1	0	0	3	1	0	2	1
Commercial Drones	8	9	23	18	74	31	29	25	37
IFT	11	0	0	18	32	0	0	0	0
Remote Video Surveillance System (RVSS)	4	0	0	0	26	0	0	0	0
RVSS Upgrades	4	0	0	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	26	0	0	0
Agent Portable Surveillance System (APSS)	0	0	0	0	0	0	11	0	11
Mobile Video Surveillance System (MVSS)	0	0	0	0	0	0	0	0	0
Thermal Imaging Device (TID)	0	0	0	0	0	0	0	0	0
Unattended Groud Sensors (UGS)	0	0	0	0	33	0	0	0	16
Imaging Sensors (IS)	0	0	0	0	0	0	98	46	0

Model 6 (Table 4.8) displays the results of the full triple-objective model, maximizing three objectives using equal weights. The optimization, maximizing cost, focuses on the IFTs, but concentrates on remote and mobile surveillance instead of small drones.

Table 4.8 - Model 6 Results

	Tri-Objective Model (Max Cost)								
Security Measure\Sector	**El Centro**	**San Diego**	**Tucson**	**Yuma**	**Big Bend**	**Del Rio**	**El Paso**	**Laredo**	**Rio Grande Valley**
New Bollard Wall (30 ft)	19	48	39	20.999998	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	1	0	0	3	1	0	2	2
Commercial Drones	9	0	0	0	0	0	0	0	37
IFT	11	0	38	18	74	0	39	0	0
Remote Video Surveillance System (RVSS)	4	0	13	6	26	0	14	0	0
RVSS Upgrades	4	0	13	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	27	35	0	0
Agent Portable Surveillance System (APSS)	0	0	0	0	0	1	0	0	0
Mobile Video Surveillance System (MVSS)	0	0	0	0	0	4	0	2	0
Thermal Imaging Device (TID)	0	0	0	0	0	0	0	0	4
Unattended Groud Sensors (UGS)	0	0	0	0	0	0	0	2	0
Imaging Sensors (IS)	0	0	0	0	0	0	0	0	0

Model 7 (Table 4.9) displays the results of the full triple objective model by minimizing cost and with a lower weight. The optimization, while minimizing cost, shows a very good distribution of technologies.

Table 4.9 - Model 7 Results

	Tri-Objective Model (Min weighted Cost)								
Security Measure\Sector	El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley
New Bollard Wall (30 ft)	0	0	39	1	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	0	0	0	3	0	0	0	1
Commercial Drones	11	9	23	18	74	31	29	25	37
IFT	0	0	0	0	73	0	0	0	0
Remote Video Surveillance System (RVSS)	0	0	13	6	26	0	14	0	0
RVSS Upgrades	0	0	13	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	15	0	27	35	0	0
Agent Portable Surveillance System (APSS)	0	3	0	6	0	9	12	0	11
Mobile Video Surveillance System (MVSS)	0	0	0	0	0	4	0	3	0
Thermal Imaging Device (TID)	0	0	3	0	0	0	0	3	5
Unattended Groud Sensors (UGS)	5	4	15	9	33	0	18	11	16
Imaging Sensors (IS)	26	22	95	46	187	77	98	63	92

Model 8 (Table 4.10) displays the results of the full triple objective model, by maximizing cost and with a lower weight. The optimization, while maximizing cost, shows a very good distribution of technologies.

Table 4.10 - Model 8 Results

	Tri-Objective Model (Max weighted Cost)								
Security Measure\Sector	El Centro	San Diego	Tucson	Yuma	Big Bend	Del Rio	El Paso	Laredo	Rio Grande Valley
New Bollard Wall (30 ft)	19	47	39	21	0	0	26	0	0
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0	0
Tacical Aerostats	0	1	0	0	3	1	0	2	2
Commercial Drones	11	9	23	18	74	31	29	25	37
IFT	9	0	38	18	74	0	39	0	0
Remote Video Surveillance System (RVSS)	4	0	13	6	26	0	14	0	0
RVSS Upgrades	4	0	13	6	26	0	14	0	0
Mobile Surveillance Capability (MSC)	0	0	0	16	0	27	35	0	0
Agent Portable Surveillance System (APSS)	0	2	0	5	0	9	12	0	11
Mobile Video Surveillance System (MVSS)	0	1	0	0	0	4	0	3	0
Thermal Imaging Device (TID)	0	0	3	0	0	0	0	3	5
Unattended Groud Sensors (UGS)	5	4	15	8	33	0	18	11	16
Imaging Sensors (IS)	26	22	95	46	187	77	98	63	92

Models 7 and 8 (Table 4.9 and Table 4.10) both provide two robust options in terms of optimizing multiple critical criteria while meeting different budgetary options. When only optimizing a single objective, certain security measures are left out that may be important to some missions.

Table 4.11 provides a consolidated model overview of how the allocation values change as we iterate through the eight models discussed above. The values in each cell represent the total units of security measures to be distributed across all sectors (sum of the rows from Table 4.3 - Table 4.10. Although each solution presented is Pareto optimal for its specific model, each model provides a trade-off solution that might be of importance to the decision-maker. We observe how different detection measures are directly impacted by emphasizing different objectives over others or using equal weights among them. Important values to acknowledge are if concentrating on minimizing cost as in Model 7, installing new Bollard Wall along the border becomes the least essential security measure to focus on while maximizing the number of commercial drones available is still a priority. In every model, it is vital to install IFTs within the sectors. This is interesting since there are hardly any IFTs operational at the moment, and they are the most expensive security measure to put into place. The model, in this case, determines that IFTs are a critical security element. Another interesting observation, if possible, it seems prudent to allocate as many drones and IFTs as possible. However, when the cost is an issue, it is important to install as many drones as possible and reduce the number of IFTs or vice versa. We see that having one or the other is critical, but having many of both is the best-case scenario.

Table 4.11 - Overall Model Comparison

	Comparison between Various Objective Models							
	Total number of each security measure to install							
SM\Different Model Results	**Model 1**	**Model 2**	**Model 3**	**Model 4**	**Model 5**	**Model 6**	**Model 7**	**Model 8**
New Bollard Wall (30 ft)	153	148	152	153	152	153	66	152
Pedestrian Fence (miles) (10-20 ft)	0	0	0	0	0	0	0	0
Secondary Fence (miles)	0	0	0	0	0	0	0	0
Tertiary Fence (miles)	0	0	0	0	0	0	0	0
Tacical Aerostats	8	7	8	9	8	9	4	9
Commercial Drones	257	183	254	46	254	46	257	257
IFT	178	50	61	180	61	180	73	178
Remote Video Surveillance System (RVSS)	63	49	30	63	30	63	59	63
RVSS Upgrades	63	57	50	63	50	63	59	63
Mobile Surveillance Capability (MSC)	78	78	42	78	42	78	77	78
Agent Portable Surveillance System (APSS)	40	40	22	1	22	1	41	39
Mobile Video Surveillance System (MVSS)	8	7	0	6	0	6	7	8
Thermal Imaging Device (TID)	11	11	0	4	0	4	11	11
Unattended Groud Sensors (UGS)	110	110	49	2	49	2	111	110
Imaging Sensors (IS)	706	22	144	0	144	0	706	706
Total Cost per Model Plan	**$ 4,996.66**	**$ 2,500.55**	**$ 2,500.00**	**$ 5,000.00**	**$ 2,500.00**	**$ 5,000.00**	**$ 2,502.50**	**$ 4,999.29**
	Note: Totals are cumulative across all 9 sectors							

4.8.1.1 Measures of Performance

Purely comparing the three objectives (Utility, Detection, Cost) in Table 4.12, we see that Models 1, 6, and 8 are the three strongest models. The three objectives are almost the same and at peak points in these solutions.

Table 4.12 - Measures of Performance Results

Measures of Performance	**Model**							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Objective 1: Utility	2458.335	919.1804976	1143.309129	832.4948118	2291.727179	2457.712656	2292.992739	245.02
Objective 2: Detection	8.999957	8.999928	8.986102	8.999588	8.999957	8.999957	8.999957	8.999957
Objective 3: Cost	4996.662656	2500.550143	2500	5000	2500.258805	4999.716503	2502.495497	4999.29

Table 4.13 summarizes the measures of performance when normalized between 0 and 1. This emphasizes the equivalence among Models 1, 6, and 8. The triangle radar plot in Figure 5 displays the normalized results. What we look for in the radar plot is for

the colored lines to reach as close to 1 in each corner as possible. If the model lines reach 1, then the objective has reached the maximum value amongst the various models. If a color is barely registering, then the objective value result was inconsequential in comparison.

Table 4.13 - Normalized Model Results

Measures of Performance	Normalized Model Results							
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
Objective 1: Utility	1	0.05333789	0.19124471	0.05072685	0.90304958	1	0.90775159	1
Objective 2: Detection	1	0.99794377	0	0.99995897	1	1	0.99999997	1
Objective 3: Cost	0.99866506	0.00022006	0	1	0.31791367	1	0.31856118	1

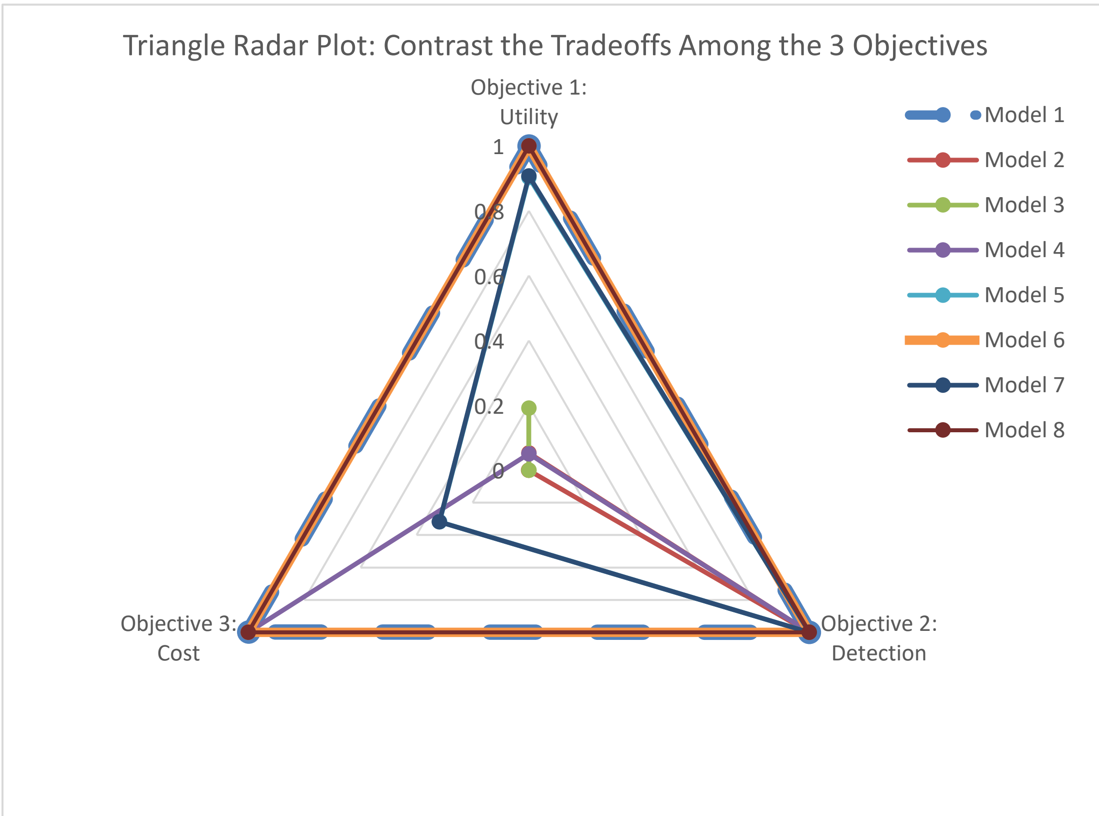

Figure 4.5 - Triangle Radar Plot Contrasting Models

4.8.1.2 Pareto Optimal Solutions

Pareto optimality is a state of resource allocation from which it is impossible to reallocate to make any single objective improve without making at least another objective worse off. The efficient frontier represents the set of Pareto optimal portfolios that offer the highest expected return for a defined level of risk. The efficient frontier in this problem displays the tradeoff among the multiple objectives and offer a Pareto optimal solution. The decision-maker can follow the efficient frontier and select Pareto optimal alternatives that provide the same overall level of return but emphasize different levels or values of each objective. Identifying potential combinations of assets is a long researched concept originally introduced by Markowitz (Markowitz, 1952). Typically, the objectives represented in a multi-objective portfolio optimization problem are competing. In this CBP analysis with multiple objectives that are being maximized (utility and probability of detection) versus one minimization (cost), there are several potentially reliable alternative solutions. Any portfolios that exist along the efficient frontier have equivalent optimal values but offer up varying combinations and quantities of security measures to be allocated amongst the border sectors.

We run the optimization instances thousands of times, varying the individual weights of the objective functions while ensuring they sum up to 1. In Figure 4.6 below, diagram A shows the variations in overall objective value while adjusting Objectives 1 and 2. What we would look for is for both objectives to be maximized, so we refer to the upper right corner of the graph for the Pareto portfolio combinations. Figure 4.6 diagram B compares Objective 1 and Objective 3, comparing maximizing utility and minimizing cost. Here we look to the bottom right-hand corner to achieve the highest utility while using the least amount of funding. For Figure 4.6 diagram C, we lastly compare Objectives 2 and 3, again looking to the bottom right corner for the best combinations of portfolios that achieve the highest level of detection while minimizing cost.

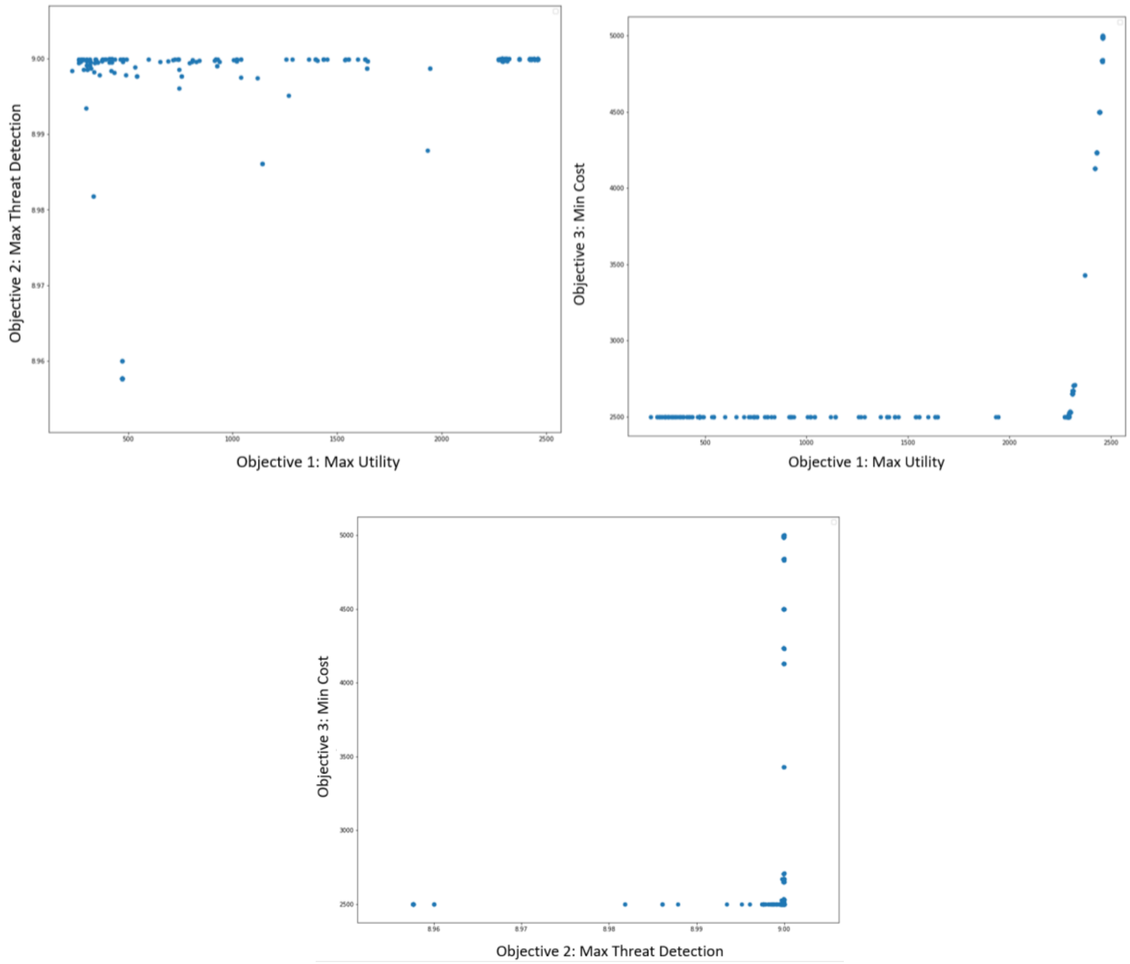

Figure 4.6 - Comparison of Objective Functions (A – top left, B – top right, C – bottom middle)

Several methods have been proposed for generating the complete Pareto efficient frontier for multi-objective optimization problems with greater than two objectives. The challenge in finding the efficient frontier comes from the number of Pareto optimal points growing with the number of objective functions. Finding a Pareto optimal point involves solving an IP, and the number of IPs to solve grows rapidly through the process. Table 4.6**Error! Reference source not found.** shows the behavior and interaction of the three fluctuating objective values together. It becomes obvious that the more funding that is available, the higher utility and threat detection capabilities are. If we reduce the budget, which is the goal, the maximum utility decreases drastically. Fortunately, the efficient frontier displays the range of possible solutions that are available for consideration to achieve acceptable good values for all three objectives simultaneously.

4.8.2 Multi-Swarm Particle Swarm Optimization Sensitivity Analysis

The purpose of this section is to answer questions about the performance of MSPSO when varying the different iteration parameters. We looked specifically at the number of sub swarms, iterations, runs, and the population size.

- Sub swarms: 5, 6, 7, 8, 9, 10
- Iterations: 10, 15, 20
- Runs: 10, 15, 20
- Population Size: 5, 6, 7, 8, 9, 10

Fluctuating these parameters impact the overall solution time, the number of full solution sets running through each iteration, the amount of time spent adjusting the solutions sets, and the number of completely different solutions that are considered. Overall, we compare 324 MSPSO instances to the exact solution of the mixed-integer program provided by the solver.

In Table 4.14 below, we show a subset of the top 26 solutions generated from the analysis. We observed these solutions primarily because their combinations of parameters were able to select portfolios that would generate solutions within 0.01% of the optimal solution. Aside from noting that the MSPSO solutions did get very close to the optimal solution, we can quickly point out two main results. First, the larger the number of iterations, the better the solution. Half of the “best” scenarios used 20 iterations, and 21/26 of the scenarios used 15 or more iterations. Half of the “best” scenarios used 15 runs. The best population size was the smallest, meaning using five sets of random solutions generated the best values. Lastly, the larger the number of sub swarms, the better. 40% of the “best” scenarios used ten sub swarms, and 22/26 using 8, 9, or 10 sub swarms. This may not appear to mean much, but more is not always better, especially when it increases memory usage and computation time. We now know that we can keep the population size and number of runs to a minimum as long as we increase the number of iterations and the number of sub swarms.

Table 4.14 - MSPSO Sensitivity Analysis

% Gap	Iterations	Runs	Population Size	Number Swarms	Obj1	Obj2	Obj3	Overall	Scenario #
0.00%	0	0	0	0	0.89	0.63	0.90	0.61200000	**Optimal Solution**
99.9926%	20	10	5	10	0.89	0.62	0.90	0.61195474	**222**
99.9926%	15	15	6	9	0.89	0.62	0.90	0.61195474	**155**
99.9926%	15	15	5	6	0.89	0.62	0.90	0.61195474	**146**
99.9926%	20	10	7	5	0.89	0.62	0.90	0.61195470	**229**
99.9926%	15	20	6	8	0.89	0.62	0.90	0.61195470	**190**
99.9926%	10	20	9	10	0.89	0.62	0.90	0.61195470	**102**
99.9924%	20	20	6	10	0.89	0.62	0.90	0.61195374	**300**
99.9924%	20	15	7	10	0.89	0.62	0.90	0.61195374	**270**
99.9922%	20	15	6	9	0.89	0.62	0.90	0.61195199	**263**
99.9922%	20	10	7	10	0.89	0.62	0.90	0.61195199	**234**
99.9922%	15	15	9	8	0.89	0.62	0.90	0.61195199	**172**
99.9922%	15	15	5	8	0.89	0.62	0.90	0.61195199	**148**
99.9922%	15	10	5	10	0.89	0.62	0.90	0.61195199	**114**
99.9921%	20	20	8	8	0.89	0.62	0.90	0.61195195	**310**
99.9921%	20	15	6	10	0.89	0.62	0.90	0.61195195	**264**
99.9921%	20	15	5	9	0.89	0.62	0.90	0.61195195	**257**
99.9921%	10	10	9	9	0.89	0.62	0.90	0.61195195	**29**
99.9921%	10	10	8	7	0.89	0.62	0.90	0.61195195	**21**
99.9920%	20	15	5	8	0.89	0.62	0.90	0.61195102	**256**
99.9920%	15	15	5	9	0.89	0.62	0.90	0.61195102	**149**
99.9920%	15	10	9	10	0.89	0.62	0.90	0.61195102	**138**
99.9920%	10	20	6	10	0.89	0.62	0.90	0.61195102	**84**
99.9920%	20	15	7	9	0.89	0.62	0.90	0.61195098	**269**
99.9920%	20	15	7	8	0.89	0.62	0.90	0.61195098	**268**
99.9920%	20	15	5	5	0.89	0.62	0.90	0.61195098	**253**
99.9920%	10	20	8	10	0.89	0.62	0.90	0.61195098	**96**

4.9 Conclusion

A large-scale resource allocation optimization integer program was successfully constructed that easily runs to optimality and provides results using Gurobi 8.1 run in Python 3.7.3. This model was directly influenced by the current TSA security screening research (Leonard, Lee, Booker 2019). The overall model continues to be very flexible and can easily accommodate different resources, new constraints, and additional objectives. The solution methodologies that are being put in place are complex, current, and effective. They will allow further development of a mathematically supported decision analysis computational tool for the CBP to provide further justification for their capability gaps and develop smart investments.

With a strong model foundation in place, this formulation is very flexible and can easily accommodate additional and/or different objectives and constraints. We acknowledge our model estimates the following input:

- False alarm detection rate for surveillance devices
- List of new and potential technologies to be considered
- Different measures of performance that can be included
- Accurate list of current devices that are employed and their locations

Working with CBP domain experts is critical to ensure realistic data is being used for analyzing the results.

The biggest knowledge gap in the research is any type of optimization model concerning enterprise risk management performs at an operational level. The TSA model is the first, which results in close to ½ billion decision variables. This CBP model is more manageable with 13,888 integer variables (448 of those are binary) and is capable of covering a full multi-tier enterprise risk management (strategic, tactical, and operational levels). To the best of our knowledge, this is the first model to mathematically determine security strategies for the CBP, as well as to introduce a utility factor to emphasize deterrence/detection impact. The model continues to be very flexible and can easily accommodate different resources, new constraints, and additional objectives.

(Leonard et al., 2019) offers an application to the large-scale system developed for TSA, and determines an optimal solution methodology for solving the security measure resource allocation model across multiple border sectors. Under physical/cyber/resource/logistics constraints, this model optimizes the allocation of limited quantities of deterrence and detection security measures across the entire southern continental U.S. border so as to maximize the total utility of the measures utilized, maximize the probability of deterrence and/or detection, and minimize cost. A utility factor is introduced to rating the impact of a security measure. The Dantzig-Wolfe decomposition approach is used to solve the nonlinear problem MIP problem instances, where optimal solutions are shown to be obtained in several seconds through several computational examples. Working with CBP, there is an opportunity to integrate a multi-tier risk taxonomy framework (Lee et al., 2019), e.g., incorporating migrants, cargos, materials, etc. and their risk interdependencies within the resource allocation framework problem to structure a risk-based screening strategy that makes effective use of limited screening resources.

CHAPTER 5. TRAUMA RESOURCE ALLOCATION

5.1 Introduction

The purpose of this chapter is to propose a model that facilitates the allocation and utilization of resources by the statewide trauma system. The primary objective was developed from the Georgia Trauma Center Network Commission's (GTCNC) objectives. The GTCNC desired to maintain and expand Georgia's trauma centers, strengthen emergency medical services in certain regions, and develop a statewide transfer system (Commission, 2009). The goal of the model that we created is to replicate the Georgia Trauma Network and run scenarios reflecting the objectives of the GTCNC to determine where the trauma system funding would have the most positive impact. We develop a mathematical model, and computational framework to (1) create the set of all feasible and Pareto-efficient portfolios where limited available funding is allocated among several requests (investments) from trauma centers, hospitals, and EMS providers, (2) quantitatively analyze the impact of each feasible portfolio on the system's performance measures via the Trauma System Simulator and (3) conduct sensitivity analysis to determine the best decision making policy to transport/transfer patient and to observe how possible changes in the system inputs affect the returns and resource utilization.

5.2 Emergency Trauma Care Prior Work

The study of trauma care systems and trauma policy development began after the Vietnam War (Nathens et al., 2004). The efforts to designate trauma centers and build regional trauma systems have continued for the last four decades. Many studies have demonstrated that the implementation of a statewide trauma system reduces the frequency of hospitalizations and death (Hulka et al., 1997). The literature covers a broad set of issues to improve the quality of trauma care and to improve patient care. One group of articles addresses the issues in the transportation of patients (Rittenberger & Callaway, 2009), (Blackwell et al., 2003), and (Cameron & Zalstein, 1998) and the impact of improvements in patient outcomes. The primary focus of these papers to determine the impact of transport time on patient survival rates. It was concluded that the actual transport time from the scene of the incident to the hospital does not have an impact. However, the time that it takes for the emergency responders to arrive at the scene once they have received the call does have an effect (Rittenberger & Callaway, 2009). Several studies that are more patient-centric focus on treatment and intervention methods (Hamilton & Breakey, 1995), (Haukoos et al., 2011), and injury evaluation methods to correctly detect the patient's condition (Vles et al., 2004). There is an abundance of retrospective reviews of trauma patient data and related statistics to further understand what variables affect the mortality and morbidity of trauma patients (Veenema & Rodewald, 1995). Most trauma-related literature concerns the operational and tactical levels of trauma care and trauma systems. However, there is a gap in the strategic level (top-down) approach to trauma systems from a financial perspective. In this paper, we take the Georgia state trauma system and model it as a network of trauma facilities, hospitals, and EMS providers. We are interested in designing a long-term development model of the network, considering the demands of each component in the trauma network and focusing on strategies that will eventually lead to improving patient outcomes. The management of finances in a statewide trauma system network that aims to maximize the quality of patient care without creating a heavy burden on trauma centers, EMS and hospitals is a critical but so far untouched task.

The range of assessed subjects is broad and distinct. There are mainly four different approaches to improve patient outcomes: (1) Patient perspective, (2) emergency medical service (EMS) perspective, (3) hospital/trauma facility perspective, and (4) policy/systems perspective. Literature that focuses on the patient perspective usually addresses how the survival, mortality, and morbidity rates are affected by the characteristics of patients (Hefny & Idris, 2013), (Prin & Li, 2016) and clinical decision making for specific types of trauma patients (Palmer, 2007). Published papers that concern the emergency medicine perspective consists of studies that assess how the length of emergency response time affect patient outcomes (Rogers et al., 2015), and the effect of prehospital trauma care on the survival rates of trauma patients (Vles et al., 2004). One class of literature focused on developing new rules to predict emergency intervention in trauma patients to improve triage effectiveness and efficiency (Haukoos, et al., 2011). Patient transportation strategies to trauma facilities or emergency departments (Veenema & Rodewald, 1995), (Brathwaite et al., 1998) and trauma system effectiveness are also extensively studied.

5.2.1 Emergency Trauma Care Problems

This project originated from Georgia's desire to improve its trauma care system statewide. The original plan was created in 2009, and the Georgia Trauma Care Network Commission (GTCNC) created a 5-year strategic plan designed to address existing deficiencies and future developments (Commission, 2009). The team explored various techniques, and findings have been recommended to the Trauma Commission leaders. The methodologies and the results and implications of the work presented are relevant not only in Georgia but in many other states throughout the U.S.

In the literature, discrete-time simulation modeling of emergency medical service systems (Wu & Hwang, 2009) has been developed extensively, but there are none that focus on trauma systems with an additional feature that analyzes and incorporates future investments into the network. In the current state, each component of the network (the trauma center) submits their requests to a central agency, or decision-maker, where each request has a cost and return. The central agency is responsible for allocating its limited budget among the requests. The cost of a request is in dollars, but returns are in terms of improvements in patient outcomes as a result of the investments made in the trauma network. The impact of any set of investments on the system can be captured via performing a simulation of the trauma system. In this chapter, possible quantitative and computational methods to handle this problem will be examined and investigated.

It is important to see the impact of the investments on the quality of care and level of trauma system infrastructure. The selection of investments that provide the best patient, hospital, and EMS outcomes will benefit both the government finances and public health. There is no conflict of interest in distributing taxpayers' money to healthcare systems the best way possible, since the shareholders, which are the taxpayers, will benefit from improvements in healthcare infrastructure to the maximum extent. Also, this model provides the opportunity for decision-makers to observe how sensitive the system outcomes are to the tactical and strategic level decisions. We believe that constructing a complete picture of trauma systems from a dynamic and strategic point of view spanning tactical-level decisions would be a unique contribution to strategic decision-making literature in healthcare systems.

5.3 Our Contributions

Our research contributes to a framework of investment allocation for emergency trauma networks. We constructed a simulation to allow a thorough analysis and systematic update of the system with given investments and to facilitate the decision-making process. The simulation modeled the potential portfolio options and generated the measures of performance. In line with the previous chapters, a MIP was developed to solve for an optimal solution. The MIP was formulated as a multi-region, multi-depot, vehicle routing pickup, and delivery problem with time windows. We were then able to include hospital and patient constraints to formulate the ambulance routing problem. Finally, investment allocation options are layered onto the ambulance routing problem to reach the full-scale model.

The chapter is organized as follows. Section “Solution Methodology” describes the full design of the simulation. Section “Simulation Results” provides the results of the sensitivity analysis and the description of the measures of performance.

5.4 Data Collection

For the simulation, we developed a theoretical layout of the Georgia trauma system. Arrival rates of patients, coordinates of TC’s, hospitals, and EMS stations, injury-related statistics, number of ambulances, hospital and TC capacities, and population and regional statistics of counties of Georgia are found from various public resources, including Centers for Disease Control and Prevention, GTCNC and the Georgia Association of Emergency Medical Services. The relevant parameters used in the simulation model are given in Table 5.1 below.

Table 5.1 - Value of Parameters Used in the Model

Parameter	**Value**	**Unit**
Patient arrival rate per year (P) (Increase by increments of 5,000)	25,000	patients
Budget to allocate for investment requests (B)	5 million	$
Parameters about trauma centers, hospitals and ambulances.		
Number of Level I TC's (T_1)	5	TC
Number of Level II TC's (T_2)	9	TC
Number of Level III TC's (T_3)	6	TC
Number of Level IV TC's (T_3)	5	TC
Number of non-trauma hospitals (H)	110	hospital
Number of EMS stations (N)	285	station
Number of ambulances (M)	2300	ambulance
Radius of circle to scan TC's to decide destination TC (r)	30	miles
Probabilistic parameters		
Probability that an ambulance is busy with other type of patients at any time	90%	
Probability that patient type I's injury severity level reduces to level II	5%	
Probability that patient type II's injury severity level reduces to level III	10%	
Probability that patient type III's injury severity level reduces to level IV	15%	
Process times distributions		
Duration between incident and first call to EMS (call time)	~Normal(3.5, 1)	minutes
Duration between call time and ambulance departure from hospital (preparation time)	~Normal(3, 1)	minutes
Duration to carry the patient to ambulance at emergency scene (carry time)	~Normal(5, 1)	minutes
Duration of patient stays in the hospital (treatment time)	~Normal(480, 60)	minutes
Number of submissions by type		
Level II TC ->Level I TC	9	
Level III TC->Level II TC	6	
Level IV TC->Level III TC	5	
Reduce incidents by 11%	10	
Can treat patient 6% faster at that site	50	
Add ambulance	9	

The exact incident times of patients are generated following a Poisson distribution because this type of data could not be accessed due to patient confidentiality issues. Process times throughout the simulations are assumed to follow a normal distribution with certain parameters, where negative random variables are omitted. Also, the number of submissions, by type, has been given as well. The TC ambulance requests are submitted by the TC's, EMS stations, and the EMS regions. It is not important which element of the system submits the request; it is the methodological framework that is being developed where one can apply data from anywhere to it to perform an analysis that matters.

5.5 Solution Methodology

5.5.1 Theoretical Model of the Problem

The first model presented is a simulation. Given that the objectives are concerning financial investments, we could have considered optimizing a constrained mathematical model to maximize patient impact. However, based on the initial discussion with the committee and information provided, a simulation was much more fitting. In our follow-up study, we derive and analyze a risk-driven resource allocation optimization model. Our simulation system presented herein allows us to evaluate a large number of alternative and realistic trauma investment plans that were identified by the decision-makers in the GTCNC. We will not be focusing on generating a single best strategy, but instead supporting the decision-makers by evaluating the numerous predefined options with a high degree of realism so that the decision-makers can genuinely understand the outcome and impact of their investments/decisions on the trauma network and most importantly on the patients. Furthermore, there is a high degree of uncertainty involved in the model, which the simulation can handle quite easily by employing properly fitted probability distributions.

Consider a network where there are *T* trauma centers (TC), *H* hospitals, *N* EMS providers, and *M* ambulances. What we refer to as hospitals are hospitals without a trauma center. There are T1 Level I TC's, T2 Level II TC's, T3 Level III TC's, and T4 Level IV TC's. The statewide EMS system consists of *R* EMS regions, and each region may cover a number of TC's, hospitals, and EMS stations. TC's, EMS regions, and EMS providers submit upgrade requests to a central decision-maker. The exact cost and benefit of the approval of a request are not known, and each request differs in their impact on the system performance measures. The central decision-maker evaluates all of the requests and selects a portfolio of investments (the requests to be approved and those to be refused) subject to a limited budget such that selection will yield the best patient outcome. The question of how to measure the best patient outcome is rather complicated since it is necessary to define measures that ensure making quantitative comparisons. The selection of quantitative performance measures that give the best representation of the system will be discussed later and are shown in Table 5.2. We will define the return of a portfolio as the percentage change in the performance measures defined for the system if that portfolio of investments is chosen. Since the selection of requests is not independent of the selection of other requests, the value of returns for each portfolio will be different from each other. A sample description of the submissions, with their costs and returns, are shown in Table 5.2.

Table 5.2 - Description of Request Types Submitted to Central Decision Maker

Request		**Cost**
Request Type I: Trauma center level upgrade		
Upgrade 1	Level II TC ⇒ Level I TC	$500k
Upgrade 2	Level II TC ⇒ Level II TC	$250k
Upgrade 3	Level II TC ⇒ Level III TC	$150k

Request Type II: A preventative plan for one region		
Prevent 1	Reduce incidents by 11%	$90k
Request Type III: An upgrade of equipment		
Equipment 1	Can treat patient 6% faster at that site	$50k
Equipment 2	Can treat patient 6% faster at that site	$50k
Request Type IV: Purchase of ambulance		
Ambulance 1	Add ambulance	$260k

As seen in Table 5.2, we will focus on four types of submissions. The first, a trauma level upgrade request, is among the most expensive, and its effect on the system is not known explicitly. A higher-level TC is typically better equipped to provide sufficient trauma care for the patients than a lower level TC. However, the overall change in the system depends on which TC is upgraded from which level due to the number of trauma patients within each region. Naturally, TC upgrades cause the pre-hospital patient flow to change. For example, some patients who are in serious condition and transported to another TC initially can now be transported to the upgraded TC. The change in the patient flow may affect the arrival rate, utilization of other TC's, ambulance assignments, and patient outcomes throughout the system. Measuring the difference in the overall performance of the trauma system is not straightforward due to the interdependency of system components if the TC upgrade takes place.

The second type of submission is to deploy a preventive measure for a specific region. A preventive measure plan includes initiatives that reduce the frequency of trauma incidents. Examples include preventing child maltreatment, preventing motor vehicle injuries, preventing falls among older adults, etc. (National Center for Injury Prevention and Control, 2018). It concentrates on preventing injuries before they happen rather than focusing on improving patient care after the injury occurs. According to the American

College of Surgeon Committee on Trauma, trauma systems must develop prevention strategies that help reduce injury occurrence as part of an integrated, coordinated, and comprehensive trauma system (Trauma, 2008). In our case, we classify a single type of preventive plan, with a similar effect on the region where it is adopted. It is assumed that the impact of a preventive plan in a region is the percentage reduction of the injuries in that region.

The third type of request is a TC equipment upgrade. Some TCs may lack specific equipment and resources to provide the highest quality trauma care. Hence, TCs submit requests for equipment upgrades in their facilities, which leads to a reduction in the treatment time of the patients and enables higher quality and safer treatments. Similar to previous types of requests, upgrades of equipment differ in their costs and impact on the trauma facilities. Since there are numerous types of equipment that can be purchased for different amounts, there is not a single cost that accounts for everything. We will consider equipment upgrades in terms of units, so a TC may request a unit of equipment upgrade funding, that if approved, would provide $50,000 for them to put towards their equipment purchases. It is presumed that the impact of an upgrade in equipment in a certain TC is known as the percentage reduction in the treatment time of patients. We note that equipment upgrades could impact other variables than the percent reduction in treatment time. For example, it can improve the accuracy of diagnosis, hence a better outcome. We caution that speed is not the only variable impacting patient outcomes.

Lastly, for the EMS submissions, EMS providers may request the purchase of a new ambulance to their fleet, particularly if they have a hard time satisfying incoming demand or patients. Usually, expanding the fleet of ambulances is a good solution to overcome this problem, especially if it is proven to be cost-effective. If the request is approved, the number of ambulances is increased by one for the region where its EMS provider is responsible for. It will yield improvements in the response times of patients; however, the question of how much improvement is achieved is implicit in the system.

Given the detailed explanation of requests, it is analytically hard to track what a certain portfolio of requests will produce in the system if it is approved by the central decision-maker. First, we need to understand the structure of the trauma system, which can be modeled as an integrated organization of trauma facilities, hospitals, and EMS providers. Designing a simulation model allowed us to represent a working trauma system realistically and is a useful tool to make quantitative observations on certain characteristics of the system. This is a top-down approach in strategic decision making of investments: the central decision-maker evaluates all the requests and forms the set of all possible investment portfolios; then, the effect of the approval of each portfolio using the trauma system simulation is observed. Therefore, the problem is divided into two stages: First, given a limited budget, select the feasible set of portfolios, such that the total cost of any portfolio in this set does not exceed the given budget, and none of the portfolios are Pareto dominant to each other. Second, given the set of feasible portfolios, build a simulated trauma system where users update the resources and attributes accordingly so that it is possible to see how the system works with different portfolios. This feature allows the decision-maker to evaluate all the possibilities of investments and to make a quantitative analysis of which investments should be approved or not. In the subsequent sections of this paper, both stages of the problem will be analyzed thoroughly, and helpful conclusions will be made.

5.5.2 Finding the Feasible Investment Request Set

Let $i = 1, \ldots, I$ be the indices of type of submissions and a_i be the number of submissions for each type. Given a_i for all i, the number of total submissions for the central decision-maker is $A = \sum_{i=1}^{I} a_i$. Let $j = 1, \ldots, A$ denote the indices for each submission, where if $a_i > 1$, then it means at least two of the submissions are the same type. Finally, let c_j be the cost of submission j and B be the size of the budget that the central decision-maker wants to allocate among the requests.

Obviously, if the total cost of submissions $\sum_{j=1}^{A} c_j$ does not exceed the size of budget B, then all the submissions can be approved. If not, then we have to consider all the combinations where the total cost does not exceed B. Suppose that there are K different combinations of investments, which we call them 'portfolios.' Let k = 1, . . ., K denote the indices of all feasible portfolios and

$$y_{kj} = \begin{cases} 1 & \text{if submission j of portfolio k is approved} \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

be the binary variable that specifies whether a request in a portfolio is approved or not.

Although it is feasible to find all the portfolios where the total cost is less than or equal to budget size, the number of possibilities is exponential. However, the decision-makers want to spend as much of their budget as possible; therefore, we can eliminate the feasible portfolios in which there is adequate funding in the budget to spend on at least one additional request. By doing so, we can reduce the overall number of portfolios. The problem can now be described as finding all the portfolios that have a cost of at most B, with no portfolio having any remaining fund that can support a potential submission. The number of portfolios is further constrained by a requirement to satisfy at least one request from each region. Let z_k denote a portfolio of investments [y_{k1}, $y_{k2}, \ldots, y_{k3}$], where $k=1, \ldots, K$. Define U_k as the set of submissions that are not selected in portfolio k. Given A investments and their costs, find all portfolios z_k subject to the following constraints:

$$\sum_{j=1}^{A} y_{kj} c_j \leq B \quad \forall\, k \tag{5.2}$$

$$B - \sum_{\mathrm{j}=1} y_{kj} c_{\mathrm{j}} \leq \min_{\mathrm{j} \in U} \mathrm{c}_{\mathrm{j}} \quad \forall\, k \tag{5.3}$$

Equation (5.2) models the budget constraint. Equation (5.3) satisfies the requirement that the remaining funding from the budget must always be less than the minimum cost of unselected submissions. This ensures that it is not possible to fund any additional investments. Any portfolio that satisfies these conditions is called a feasible portfolio, and the set of such portfolios is called the feasible portfolio set.

In general, the number of portfolios K is too large to examine. If there are A submissions to take into consideration, then the number of distinct portfolios is 2^A, which means it is impossible to evaluate the feasibility of all portfolios for larger values of A. Therefore, it is necessary to find an efficient algorithm that generates all portfolios that satisfy these three conditions. With such an algorithm, the size of the set of portfolios that one has to evaluate will be reduced to a surmountable number.

It is worth pointing out that we treat each submission equivalently in terms of their impacts. For some submissions, such as upgrading level of TCs, the impact of implementation is not known, and even if the impact of one submission is known, the output obtained in the system through interaction of several submissions to be implemented is unknown. In other words, it is not possible to prioritize the investments due to each investment yielding improvements in different metrics of the system. Furthermore, there is intrinsic uncertainty in the overall patient impact. Hence, it is necessary to evaluate all the possible combinations that a feasible portfolio can take.

We propose an algorithm that takes the costs of all submissions and the size of the budget as inputs and produces the feasible portfolio set. It does not give the best feasible portfolio due to the complexity outlined previously. The following recursive algorithm performs the task of obtaining the feasible portfolio set:

The idea of the algorithm is, to begin with the array 'sortedCost,' and add additional investments to the array 'base' until the total cost exceeds the budget limit B. This must be done on a systematic way so that all the feasible portfolios are achieved, and there is no repetition of portfolios. Algorithm 2 is essentially a procedure that gets the index number, base array, and remaining budget as inputs and produces the set of feasible portfolios associated with the inputs. Algorithm 1 iterates the procedure that Algorithm 2 performs, for all index numbers between 0 and A. This ensures that all possibilities are covered, and all the feasible portfolios are added to the list.

Algorithm 1: Procedure to iterate all cases

```
Result: The set of feasible portfolios
Initializations
for i ← 1 to A do
|  base(i) ← 0
end
k ← 0
// Loop function IterateOne(k,B,base) over k
while k < A do
|  IterateOne(k,B,base)
|  k++
end
```

Algorithm 2: *IterateOne(cc,B,base)*

```
Input: Cost of each investment(cost) and size of budget (B)
Result: Feasible portfolios for one iteration are added to the set of feasible portfolios
Initializations
// Sort investments from smallest to largest cost
sortedCost ← sort(cost)
begin
|  if sortedCost[cc] > B then
|  |  addToFeasibleInvestmentSet (base)
|  else
|  |  remainingBudget ← B – sortedCost[cc]
|  |  newbase ← updateBase (base,cc)
|  |  for j ← cc+1 to A do
|  |  |  nextBudget ← remainingBudget
|  |  |  - sortedCost[j]
|  |  |  if remainingBudget ≥ 2 ×sortedCost[j] then
|  |  |  |  IterateOne(j,remainingBudget,newbase)
|  |  |  end
|  |  |  else if (nextBudget < sortedCost[j] and
|  |  |     (nextBudget < min(sortedCost)) and
|  |  |     (nextBudget ≥ 0 then
|  |  |                    newbase2 ← updateBase(newbase)
|  |  |                    addToFeasibleInvestmentSet(newbase2)
|  |  |  end
|  |  end
|  end
end
```

The essential part of the algorithm is the procedure named Algorithm 2. It provides the following: Given an index number, it checks whether the cost of submission at that index is greater than the remaining budget. If it is, then we stop and add the 'base' array to the feasible investment set. If not, then we need to continue to investigate all other investments with a greater index independently. Add the investment on index *j* if *cost[j]* < *remaining budget*, where *j* > *index*. Also, the remaining budget is updated by extracting the cost of the investment that was added last if it satisfies three main conditions:

1) If the updated remaining budget > cost of investment that is added last, then continue to iterate the procedure with given inputs (j, updated remaining budget, updated base array). This means if we decide to add an investment to the portfolio, then we must continue to apply the same procedure starting from the index where we arrived last. Recursively, we check all of the possibilities for unarrived indices and add investments to the portfolio if there is a budget available for that specific investment and then update the remaining amount of the budget.

2) If the remaining budget < cost of investment that is added last AND the remaining budget > cost of the cheapest investment, then do not add the investment at that index to avoid repetition of the same investment in the portfolio. The algorithm does not proceed to search for new investments if the remaining budget is less than the cost of the investment that is added last because the costs of investments with greater indices are greater than

the remaining budget. At that point, the ’base’ array must be the portfolio. However, it is a Pareto dominated portfolio if the second condition, remaining budget, is greater than the cost of the cheapest investment, is satisfied. The reason is since the array of costs have been sorted, the procedure will arrive at the cheapest options at some point in the procedure, and if the remaining budget is greater than the cost of the cheapest investment, it means that the cheapest option has not yet been added to the portfolio. Also, if we add the cheapest investment to the portfolio, it will be the same as a portfolio that was created before with the recursive method in Condition (1). This condition always leads the algorithm to the correct path due to the sorting of costs from smallest to largest.

3) If the remaining budget < cost of investment that is added last AND remaining budget < cost of the cheapest investment, then stop. Do not add the next investment because there is no longer enough remaining budget to add an investment to the portfolio. We then add the array ’base’ to the set of feasible portfolios.

The feasible set of portfolios consists of an array of 0-1 variables, where 1 indicates approval, and 0 indicates the refusal of submission at that index. For our problem, the inputs are summarized in Table 18. The size of a portfolio is 89, and given this cost structure, the number of feasible portfolios in the set is greater than 2 million. In other words, there are greater than 2 million possible combinations of approvals and refusals of submissions given the financial information in Table 1. The effect of each combination on the trauma system will be evaluated using the Trauma System Simulator that we have built and described below.

5.5.3 Designing the Trauma System Simulator

Trauma incidents happen at random times following a Pois- son process with rate P/year. This implies that each year, P trauma incidents occur that require a patient to be transported to a trauma facility. The moment a traumatic incident occurs, it is assumed that this incident happens to only one person, and the location of the patient is generated by the steps described next. First, we determine the county where the incident occurred. The probability that an incident happens in a county is proportional to the ratio of injuries in one county to the injuries in the state. A county has been modeled as a square with a center, and patient coordinates are generated uniformly in a square where its area is equal to the area of the county. Patient arrival is also determined by patient transfers from a lower level TC or non-trauma hospitals. In addition, it is assumed that patients' conditions may differ, and each patient is assigned an injury severity level according to the ratio of patients who have been treated in Level I, Level II, Level III, and Level IV TCs in the past.

As it has been stated before, there are N EMS providers and M ambulances belonging to the EMS providers, where $M > N$. The exact coordinates of EMS stations are known. It is assumed that ambulances are evenly distributed among the EMS stations. Ambulances are not only busy with trauma patients but also other types of patients, such as cardiac patients. Therefore, in the simulation at any point in time, ambulances are available with a certain probability. This is realistic since the EMS resources are not only allocated for trauma patients but also for other types of patients.

The location and capacities of all TCs and hospitals are known. Capacity is defined as the number of trauma units in a TC and the number of trauma beds allocated in hospitals. We did not specifically consider the number of human resources or special equipment to define the capacity because the usage of those resources is very complicated in the hospital environment. The topic of efficient management of resources in the hospital is out of the scope of this paper. A TC or hospital has to admit the patients who arrive. Once they are admitted, if there are any available trauma units, the patient treatment begins. The patients stay in the hospital for a period of time, the length of which is determined by a random variable, and they are either moved from the emergency department or transported to another facility to receive better treatment if necessary.

The simulation is intended to model the trauma system with its components, interactions, and decisions. The system works as follows: An incident occurs, a trauma patient is created, then emergency services are called to transport the patient to a hospital. EMS assigns the closest available ambulance to the address where the call has been made. An ambulance responds to the assignment and quickly drives to the address. Once they arrive at the incident point, they carry out the first intervention and transport the patient to the ambulance. The target hospital or TC is decided according to a procedure, which is summarized in Algorithm 3. The ambulance transports the patient to the most appropriate TC or hospital (based on the trauma incident level) and returns to its station once the patient is delivered to the hospital. A patient's treatment starts if there are any trauma units or beds available; if there are none, then the patient enters a priority queue, where he/she waits until the next trauma unit becomes available for treatment. For trauma beds to become available, the patient who is being treated at that trauma unit must be either moved from the emergency department to a non-trauma bed or transported to another emergency facility for a trauma bed to become available.

Algorithm 3: Procedure to decide a patient's TC or hospital
Input: Patient's coordinates and location of all TCs and hospitals
Result: Decision of TC or hospital where the patient must be transported
Check all the TCs within radius of *r* miles from the patient's location.
Define set S as the set of trauma centers within the radius.
case If there are any TCs with higher level in *S* **do**
| ⇒ Send the patient to the closest TC with higher or
| equal level.
case If there are only equal AND lower level TCs in *S* **do**
| ⇒ Send the patient to the closest equal level TC
case If there are only equal level OR lower level TCs in *S* **do**
| ⇒ Send the patient to the closest TC
case If no TC within the radius. S=∅. **do**
| ⇒ Send the patient to the closest hospital, since there is | no TC within the radius.
end

If the patient has been transported to a lower level trauma facility or an ordinary hospital, this patient is considered to be a candidate for transfer to a higher-level TC to receive better and/or more suitable treatment for his/her injury. With a certain probability, the patient's severity of injury may reduce, and there is no need to transfer the patient to another facility. However, if the injury remains too severe and requires the patient to be treated at another TC facility, then the process of patient transfer is initiated. The patient transfer process is almost identical to the process of responding to a first-time patient. In a transfer situation, the location of the patient is the hospital where he/she is treated. Similarly, the EMS is called, and the closest available ambulance is assigned for patient transfer; when the ambulance arrives at the hospital, the patient is picked up and transported to the most appropriate TC. There is a decision-making process to determine which trauma center the patient should be transferred to. In this case, we ignore whether there is any TC within a radius *r*; instead, we simply send the patient to the closest available TC, where the level of TC is greater than or equal to the patient's injury severity. If there is no available higher-level TC due to capacity limitation, then the patient is sent to the closest available TC, where the TC level is higher than the current facility's level. If there is none in this case as well, which is quite exceptional, then the patient is left in the current hospital. Treatment continues for a certain time, and then the possibility of patient transfer is considered again.

Figure 1 shows the patient flow schema within the TC simulation.

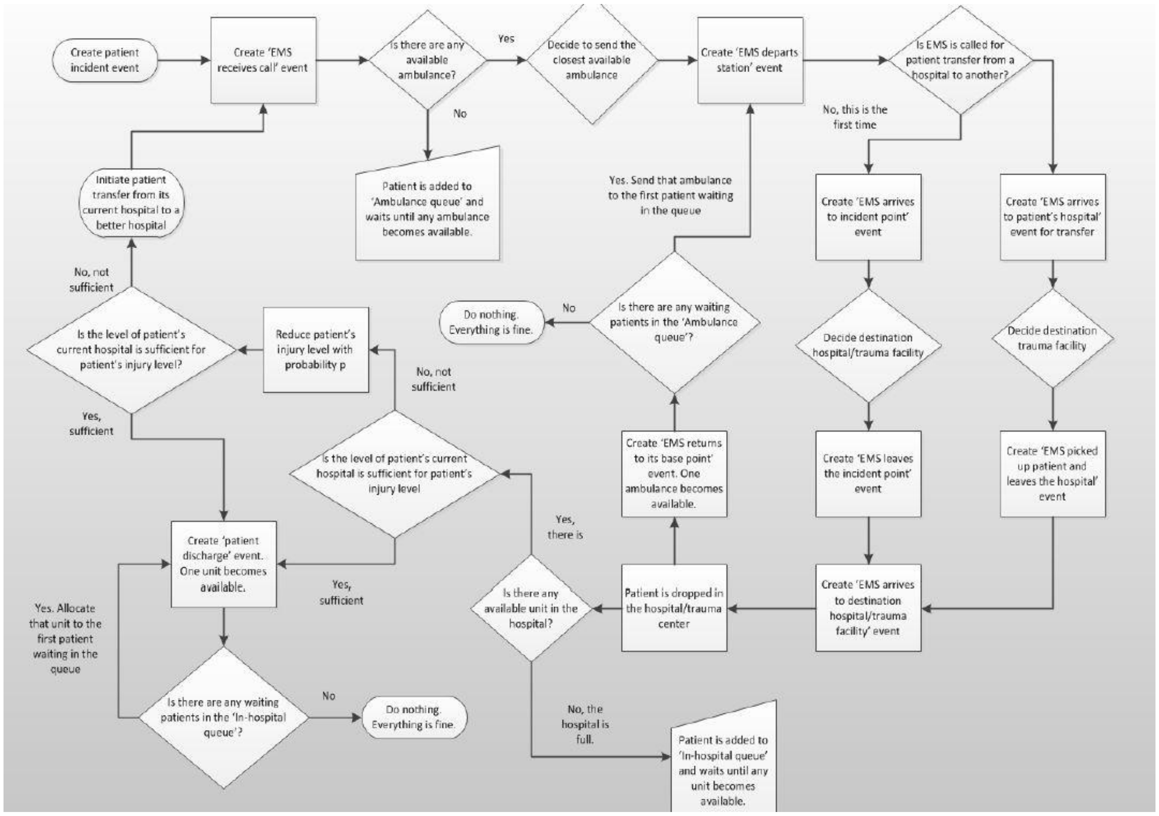

Figure 5.1 - Flowchart of a working trauma system

5.5.4 Outcome Measures

The outcome measures can be categorized into three classes: (1) Patient-related statistics, (2) EMS related statistics, and (3) TC or hospital-related statistics. Some measures can be an element of more than one class. Patient-related statistics include the number of patients transported in a year, the number of patients transferred from one facility to another facility, average time between the incident and the patient's arrival to the designated TC, the average waiting time for a trauma unit of a patient in the hospital, etc. EMS related statistics consist of average deadhead miles per ambulance, average response times per ambulance, and the average time that an ambulance is active. Lastly, TC or hospital-related statistics include the average number of patients waiting in the queue of the trauma unit, the proportion of patients who have waited in the queue to the total number of patients who have arrived, and average utilization. While the patient outcome is an important measurement, it is difficult to quantify objectively.

Three metrics among the outcome measures, each from one of these three classes, have been chosen to assess the overall performance of the trauma system. The chosen metrics are (1) average time between the incident and patient's arrival to the destinated TC, (2) average deadhead miles per ambulance, and (3) the proportion of patients who have waited in the queue to the total number of patients who have arrived. The first metric is patient-related due to the importance of transporting the patient to a TC or hospital as soon as possible. The second metric is ambulance-related since average deadhead miles give EMS an idea about ambulance utilization and is used in computing the financials of EMS. The last metric is TC-related since the patients who have waited in the queue do not receive sufficient quality and timeliness of care in the TC, and the proportion of that number to the total number of patients arrived at the TCs gives an idea about the quality of the trauma care and treatment in that TC. The basic results that are obtained via the Trauma Simulator consist of these three metrics. Comparisons among portfolios will be made with respect to those three metrics because it is believed that they are good indicators of the overall performance in the system.

5.5.5 Analysis

There is one major assumption modified in order to observe how modifications change the results of the system: the rate of trauma incidents. Initially, this rate is taken as 25,000 patients/year; we would like to analyze how the system responds if the patient arrival rates increase in increments of 5,000 up to 45,000 patients/year. These values were selected from the estimated number of trauma patients per year in Georgia. The primary reason to observe the effect of the arrival rate change is at first to notice whether any of the metrics chosen are in conflict with each other with the increasing or decreasing demand. If there is a tradeoff between the metrics, the question of how much it is must be addressed in order to understand the interaction between different components. However, since all the inputs are empirical, we will only present the results of 25,000 patients/year.

Initially, a second measure, scanning radius, was considered for sensitivity analyses due to how it alters the patient flow between scenes of emergency to the TCs. However, it quickly became clear that the smaller radius was always in favor of all scenarios. For the purpose of our analysis, we maintain a constant scanning radius throughout the simulation. There is a strong, positive correlation between the magnitude of radius for scanning and the values of all performance measures. This implies that during the decision-making process of patient transportation to a TC/hospital, EMS should only include closer TC/hospitals into the set of destination candidates. Including hospitals further from the area of incidence may result in the selection of TC/hospitals farther away, thus increasing the transportation time. Therefore, it is sufficient only to take into account the closest appropriate facilities to decide the assignment of a patient to a TC/hospital. Census results and geographic locations of all of the trauma centers in the U.S. have been analyzed and have shown that approximately 85% of the U.S. population is within 30 miles of the nearest trauma center. This includes both urban and rural populations. Given these results and how similar Georgia's specific proportion of Urban to Rural populations is to the U.S. proportions, we use 30 miles as our ambulance search radius from the location of the trauma incident.

5.6 Simulation Results

The Trauma System Simulator is run over a time interval of 2 years. Some results of the simulation run for parameters P = 25,000 per year are given in Table 3, and the associated best portfolios list is given in Table 4. Each identified portfolio is the best considering at least one metric, but it is also possible that some portfolios are the best in multiple metrics. Given the number of feasible portfolios, results were collected by sets of 1,500 feasible portfolios, and best portfolios by metrics were first selected amongst the sets, and then the best of this cohort were selected for further analysis.

Table 5.3 - Best observed values of selected metrics by sets

Set #	Average time between incident and patient arrival to TC (minutes)	Average deadhead miles time per ambulance (minutes)	Proportion of trauma patients who received lower level of trauma care
1	0.9083104	35.25873	0.003879
2	0.9149609	35.246306	0.003634
3	0.9260064	35.519721	0.004540
4	0.9267546	35.437128	0.004423
.	.	.	.
9	0.9249691	35.438552	0.004566

10	0.9263578	35.525422	0.004623
Avg	0.9206751	35.385691	0.004196

Table 5.4 - Index of best portfolios for each metric in each set

Set #	Average time between incident and patient arrival to TC	Average deadhead miles time per ambulance	Proportion of trauma patients who received lower level of trauma care
1	160	160	359
2	2602	2919	2603
3	4002	4002	3598
4	4541	4596	4589
.	.	.	.
9	13074	13074	13262
10	13558	13558	13960

It is observed that for each metric, different portfolios turn out to be the best ones. We record which portfolios are marked as the best portfolio for each metric and count how many times they were observed as the best across their set of 1,500 portfolios. This is tabulated in Table 22, and it appears that some portfolios are dominant to others. However, counting does inform us that some portfolios are significantly dominant to others; therefore, we need to define a procedure such that we can decide on the best overall portfolio. This procedure is briefly described in Algorithm 4.

Table 5.5 - Values of performance metrics from Final Investment Selection

Portfolio #	m2	m3	m7
160	35.2539	54.8403	0.00236
7152	35.4406	55.2292	0.00161
9268	35.2692	54.8752	0.00253
11458	35.5353	55.5022	0.00168

m1: Average number of patients not yet transported to the hospital (min)
m2: Average time between the incident and patient arrival to TC (min)
m3: Average deadhead miles time per ambulance (min)

m4: Average number of ambulances in service
m5: Average number of people in all hospitals
m6: Total number of patients who receive lower-quality care
m7: Proportion of trauma patients who received lower level of trauma care

The metric outcomes for each portfolio have been plotted in 3-D graphs. Each dimension is represented by a metric, and each point represents the corresponding portfolio. The convex hull of points gives an idea about the feasible manifold of the best portfolios in three dimensions. Here, the corners of the convex hull are of interest. The set of corner points are examined in 2-D for each pair of metrics, and for each step, the Pareto efficient portfolios are marked. This reduces the number of candidate portfolios to a smaller number. Once Pareto efficient portfolios are found for all pairwise comparisons, the procedure aims to find a portfolio where it is observed in a maximum number of sets of pairwise comparisons. If there exists only one such portfolio, it is decided as the best portfolio. If there exist multiple portfolios, then the procedure focuses on common investments approved among portfolios. It finally reaches a conclusion, where it may be either one unique portfolio or a set of portfolios. The decision to select the final portfolio is left to the decision-maker.

A formal description of the procedure is as follows:

Indices:

V = {25000, 30000, 35000, 4000, 45000}: the set of arrival rates per year

M = {1, 2,…, 7}: the set of metric indices

I = {1, 2,…, 2304028}: the set of portfolio id's

Variables:

X_{vmi}: the value of metric m for portfolio i, obtained by running the simulator with $P=I_i$.

We index the three chosen performance metric indicators as $t = 1, 2, 3,$ and define $d_{vt} = \{i: \min_i\{ X_{vti} \}\} \ \forall\, v \in V, t = 1,2,3$ as the ID of the portfolio where the minimum value of metric m is attained when the simulator is run with parameter v. Also, let $d_V = [d_{V1}\ d_{V2}\ d_{V3}]^{\boldsymbol{T}}$

Define z_v as a 3x3 matrix, where it stores the values of each performance measure observed for the portfolio in which a minimum is attained for one of the performance metrics. In matrix formation,

$$z_v = \begin{bmatrix} X_{v1d_{v1}} & X_{v2d_{v1}} & X_{v3d_{v1}} \\ X_{v1d_{v2}} & X_{v2d_{v2}} & X_{v3d_{v2}} \\ X_{v1d_{v3}} & X_{v2d_{v3}} & X_{v3d_{v3}} \end{bmatrix} \tag{5.4}$$

What follows is to obtain the Pareto efficient frontier for all pairwise comparison of three metrics (dimensions). Since there are three metrics, it comes with $\binom{3}{2} = 3$ possible 2-D spaces and all the points on the Pareto efficient frontier are recorded. Define $E_v = E_{v(1-2)} \cup E_{v(1-3)} \cup E_{v(2-3)}$ as the union of elements on the Pareto efficient frontiers for all pairs of dimensions. Then, we define

$$EI_v = \left\{h\colon \max_{h\,\in\, E_v} \left\{1_{\{h\in E_{v(1,2)}\}} + 1_{\{h\in E_{v(1,3)}\}} + 1_{\{h\in E_{v(2,3)}\}}\right\}\right\} \ \forall u, \forall v \tag{5.5}$$

as the set of indices h, where any $h' \neq h \in E_v$ is observed in all the sets $E_{v(1-2)}$, $E_{v(1-3)}, E_{v(2-3)}$ not more than $h \in E_v$. Here we try to find portfolio(s) that is (are) the most common among the 2-D efficient frontier index sets. Once EI_v's are found for $\forall v \in V$, it is possible to track the process backward to find the best portfolios. We then look at the set of portfolio ID's corresponding to the indices that have been most commonly observed in all Pareto efficient sets of two dimensions. If the set size is one, it implies there is a unique best portfolio for the given parameters v. However, if the set size is greater than one, it indicates that there are multiple portfolios where they are non-dominated with respect to their impact on performance measures. In this case, we can stop ignoring the cost of portfolios. In the beginning, we assumed that since all feasible portfolios have almost identical costs, we treated them equally and only focused on their impact on performance metrics. Now since we performed a systematic reduction on the feasible portfolios, we can use the cost of the remaining portfolios as a final step to further reduce the size of this set. Hence, let

$$OF_v = \left\{ f\colon \min_{f \in EI_v} \{cost\ of\ portfolio\ f \ \in \ EI_v\} \right\} \tag{5.6}$$

be the set of the portfolio(s) in which minimum cost is attained in the set EI_v. It is noteworthy to emphasize that $s\ (OF_v) > 1$ is possible for any $v \in V$. In this case, the final decision is left to the priorities of the decision-maker. The application of this procedure to our problem produced the selected portfolios with their corresponding performance measures for each v ∈ V as tabulated in Table 5.6. The selection of investments has not been displayed simply due to the sheer size of the investment set.

The selected portfolios were then compared across their specific investment sets. We observed that all four of the final portfolios were consistent in requiring the reduction of trauma incidents within the Regions through prevention methods. Also, they all required that each trauma center be allocated both units of upgrade funding ($100,000 total) to put towards the purchase of new equipment. The best performing portfolio (160) did not require the purchase of any additional ambulances, whereas the other three portfolios purchased at least one ambulance. Lastly, portfolio 160 also focused on upgrading Level IV TCs to Level III TCs.

As previously mentioned, we looked at the interaction of all the metrics with one another to determine if there were any significant relationships. All combinations of the seven metrics were

Table 5.6 - Performance Metrics

m1	Average number of patients who not yet been transported to the hospital (minutes)
m2	Average time between incident and patient arrival to TC (minutes)
m3	Average deadhead miles time per ambulance (minutes)
m4	Average number of ambulances in service
m5	Average number of people in all hospitals
m6	Total number of patients who receive lower-quality care
m7	Proportion of trauma patients who received lower level of trauma care

The following general results were observed. Metric 1 and 2 have a strong positive correlation, and they behave in a similar manner with respect to all the other metrics. A similar positive correlation exists between Metrics 3 and 4. Metrics 1 and 2 have a strong positive correlation with Metric 3 and Metric 4, as seen in Figure 2. Meaning the longer it takes for an ambulance to respond to a call and get the patient to the TC, the more patients that will end up waiting for a response.

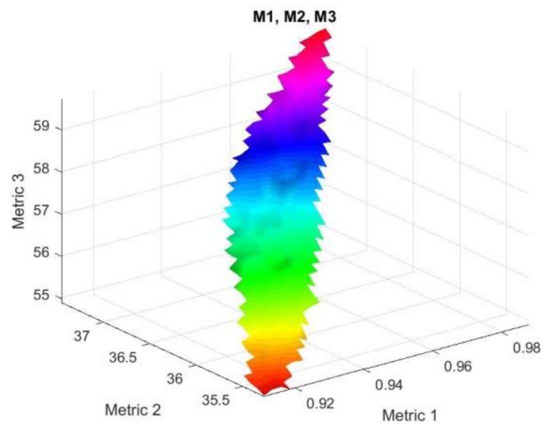

Figure 5.2 - Metric 1, Metric 2, Metric 3 Surface Plot

These values are also related to the average number of miles an ambulance must travel to pick up the patient and to get to the TC. The longer the distance, the longer the response, and the patient delivery take. All of these are again related to the number of ambulances in service. Metric 7 is negatively correlated with Metric 6. There is a slight positive correlation between Metrics 1 - 4 and Metric 5 and Metric 7. Metrics 1 - 4 appear to have no correlation with Metric 6. Metric 6 and Metric 7 also have a very strong positive correlation, but that is simply because Metric 7 is calculated using Metric 6. When Metric 6 is observed with the other metrics, as shown in Figure 3, we can easily observe that local minimums and maximums spread throughout the linearly increasing trend line. There is a slight positive correlation with Metrics 1 - 4 with Metric 6, but due to the spikes, we see that the number of patients receiving low-quality care is not always consistent with higher average times. Metric 5 does not appear to have any correlation with Metric 6 or metric 7, as shown in Figure 4, so the average number of TC patients in hospitals is not related to the number of TC patients receiving a lower level of care. The average number of people in all hospitals maintains a very tight range. Overall this means that regardless of the number of trauma patients, hospitals stay busy at the same rate, which is dependent on the number of beds. This is also due to trauma patients having priority over non-trauma patients.

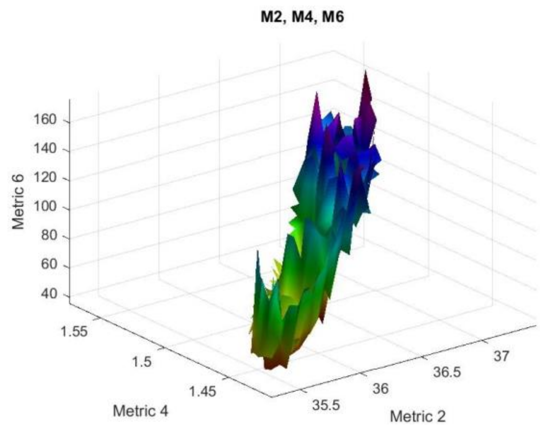

Figure 5.3 - Metric 2, Metric 4, Metric 6 Surface Plot

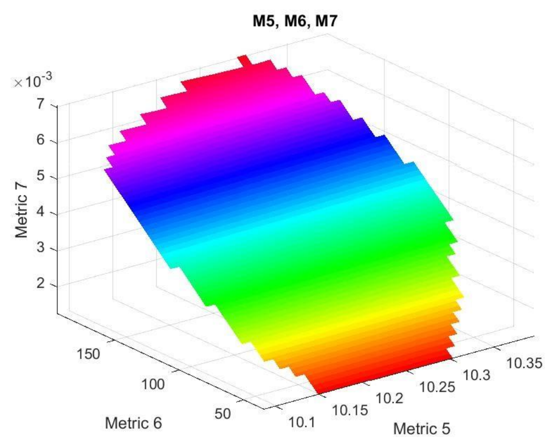

Figure 5.4 - Metric 5, Metric 6, Metric 7 Surface Plot

5.6.1 Simulation Limitations

Since we could not access the exact time of trauma incidents, it is assumed that the inter-incident times follow a Poisson distribution with rate P patients per year. In general, the trauma bed capacity for the TCs is correct, with one or two exceptions, in which case we assumed a minimum number of trauma beds. Since non-trauma care hospitals do not have designated trauma beds, we assume trauma patient capacity to be equal to the average trauma patient capacity of all facilities that are known. Also, there is a complication about the ratio of trauma patients who are Level I, Level II, Level III, and Level IV. We assign those numbers according to the injury severity score of the patients, but the decision of the destination hospital does not solely depend on this. However, we ignored the complexity of details during the decision making to simplify the model.

The next stage in the model development is to formulate the mixed-integer program. Although there are similarities in resource allocation, the foundation of the model relies on the patient incident arrivals. This naturally drove the formulation to pickup and delivery problems. The ambulances are the vehicles and are stationed at depots, which include fire stations, hospitals, and individual emergency stations. The patients are the clients to be picked up with a pickup time window based on their injury incident time. Pickup and delivery problems and related, adapted model formulation are described in the following sections.

5.7 Pickup and Delivery Problems

The general pick-up and delivery problem (GPDP) has been a popular research topic in the last two decades due in part to its practical importance. The principle of the GPDP is to construct a set of routes to satisfy transportation requests. Each request specifies the load size, origin, and destination locations, and each load must be transported by one vehicle without transshipment at other locations (Parragh et al., 2008).

The literature on single-region VRPDPs is quite extensive. (Savelsbergh & Sol, 1995) define the general delivery problem (GDP) as a vehicle routing problem where pickup and delivery customers must be served. In (Cordeau et al., 2008), PDPs, a special set of GDPs, are defined as routing problems where a set of vehicles that begin and end at a depot must satisfy a set of requests. A request typically consists of a pair of pickup and delivery locations between which goods must be transported. An in-depth review of VRPDPs and its variants and analysis on all problem variants can be found in the surveys in (Parragh et al., 2008) and (Berbeglia et al., 2007). The dial-a-ride problem (DARP), a PDP variant, in which loads and load size are represented by a single person (Guerriero et al., 2014).

Most of the previously mentioned studies assume a single depot (and a single region) setting, although multiple depots are relevant in real-world settings. (Nagy & Salhi, 2005) consider the VRPDP with mixed backhauls (also known as VRPMB in (Parragh et al. 2008)) from both a single and multi-depot perspective. The VRPMB is a special case of the VRP in which pickup and delivery customers do not need to be paired. To reach a solution, the authors propose a heuristic based on the application of different routines over an initial solution for the single depot case and adapt it to the multi-depot case. (Min et al., 1992) presented a multi-depot model similar to the VRPMB, but all delivery customers must be served prior to pickups. They proposed a three-phase algorithm that clustered and assigned customers to depots and routes prior to the route optimization. Last, Bettinelli presents a multi-depot heterogeneous PDP with soft time windows (Bettinelli et al., 2014).

Traditionally, heuristics run faster than metaheuristic methods, whereas metaheuristics usually outperform simple heuristics with respect to solution quality (Bruck et al., 2012). Authors in (Jaw et al., 1986), (Madsen et al., 1995), (Diana & Dessouky, 2004), (Lu & Dessouky, 2006) solve the PDPTW with a variety of insertion-based heuristics while (Nanry & Barnes, 2000) and (Cordeau & Laporte, 2003) developed Tabu search heuristics for the PDPTW. Simulated annealing, genetic algorithm, adaptive and large neighborhood search heuristic, and variable neighborhood search heuristic for solving the PDPTW are designed in (Parragh et al., 2010), (Li & Lim, 2001), (Pankratz, 2005), (Ropke & Pisinger, 2006) respectively. (Liu et al., 2013) proposes a genetic algorithm and tabu search method for a special simultaneous PDPTW.

In this section, we study a variant of the DARP that is multi-region, multi-vehicle, multi-depot, pick-up, and delivery problem with time windows (m-MRMDPDPTW) with paired pick-up and delivery locations. In (Alaia et al., 2015), they take the m-MDPDPTW and present it as a multi-criteria optimization problem. The objective is to define a set of solutions or routes that minimizes total travel distance, total tardiness, and the total number of vehicles. In their problem, the requests are transported by a single-vehicle between paired pick-up and delivery locations. The main contribution is the use of a genetic algorithm to rank and select solutions along the Pareto fronts with an elitist replacement strategy.

There are many evolutionary approaches to the MDVRP. In (Ombuki-Berman & Hanshar, 2009), they use a genetic algorithm (GA) and introduce a mutation operator to target the depot assignment to "borderline" customers, which are close to several depots, to solve the MDVRP. An algorithm named fuzzy logic guided genetic algorithms (FLGA) to solve VRPs with multiple depots, customers, and products is presented in (Lau et al., 2009). The authors combine GA search and fuzzy logic techniques to modify the crossover and mutation rates. In (Prins et al., 2014), an excellent survey of published papers, with more than 70 references, involving order-first, split-second methods are proposed for the MDVRP. Also, a solution to the VRP using heuristics methods is proposed in (Nagy & Salhi, 2005) to solve the simultaneous VRPPD for both single and multiple depots. Finally, (Wang, Xu, & Shang, 2008) designed a new genetic algorithm for MDVRPTW with heterogeneous vehicle limits.

Multiple region PDPs have yet to receive much attention in the literature. To the best of our knowledge, they are first presented in (Dragomir et al., 2018), where the authors discuss the application of multiple regions to logistic problems and present a mathematical model. The multi-region multi-depot pickup and delivery problem (MRMDPDP) is well defined in (Soriano et al., 2018). Here they define a region as an area where customers and depots are located. Requests are differentiated by whether or not they are within the same region. They decompose the problem into three subproblems and use an adaptive large neighborhood search algorithm to perform their computational studies.

In line with all the works summarized above, (Leonard & Lee, 2020) continue to expand on the concept of MRMDPDP. These problems consist of two or more geographically separated areas, with at least one depot in each of them. Typically there are a set of pickup and delivery requests to be serviced, whose origins and destinations are located in the same regions. Other requests with endpoints in the other regions can be part of the problem, too. A customer has considered any service point in a region being either a pickup or a delivery point. Customers are visited by ambulances performing tours on an intra-region basis. The goods, in this case, are the injured patients and, once picked up, are transported directly to a servicing hospital. Our problem is a variant of the general setting of the family of multiple regions problems previously described and is composed of more than two regions and two or more depots per region.

5.8 Ambulance Routing Problems

From the current state, we adapt the previous model into an ambulance routing problem (ARP). Many ARPs in prior literature concentrate on ambulance routing in disaster scenarios when a large number of injured people from various locations require medical attention (Tikani & Setak, 2019). Although we are not dealing with the same emergency state, it is still a critical issue to manage the fleet of ambulances to accommodate all trauma requests promptly.

There are several studies that address issues of locating, dispatching, and ambulance fleets. The main concern of EMS is immediate patient care prior to hospital arrival since any delay in treatment could affect the patients' conditions (Lam et al., 2015). The growing demands for EMS have made it a very active research area in transportation and health care management. Ambulance management problems can substantially improve healthcare systems by improving response times and assigning a suitable ambulance to injured patients. Besides these complicated factors, ambulance planning in disaster events is more complicated than normal circumstances due to increased injuries, casualties, and lack of appropriate vehicles. Despite the importance of this issue, few works study vehicle fleet routing problem in post-disaster states (Luis et al., 2012), (Pedraza-Martinez & Van Wassenhove, 2012), and (Talarico et al., 2015).

(Andersson & Värbrand, 2007) studied ambulance dispatching determined by the urgency of the call and the distance of the ambulance to the incident location. Other models are developed concentrating on capturing realistic planning situations like traffic-dependent traveling times and congestion. For example, (Schmid & Doerner, 2010) studied an ambulance location problem using varying travel times throughout the day. By considering these variations, ambulance deployment changes dynamically to fulfill coverage.

Knight (Knight et al., 2012) formulates a model to locate ambulances to maximize the overall expected survival probability of multiple patient classes. Schmid, (Schmid, 2012) considers emergency service providers that locate ambulances such that emergency patients can be reached in a time-efficient manner. In the proposed model, during the dispatching process, incoming emergency requests are assigned to ambulances, and a vehicle needs to be immediately dispatched to the patient's location. After serving the patient, the ambulance is relocated to its next waiting location. (Toro-DíAz et al., 2013) developed a mathematical model that integrates location and dispatching decisions. They incorporated queuing and traffic congestion with the dispatching decisions by considering a fixed priority list for each customer.

Zhang (Zhang et al., 2015) proposed a patient transportation problem formulated as a multi-trip dial-a-ride problem and provided a modified memetic algorithm for solving the problem. (Tlili et al., 2017) formulated the ambulance routing problem as an open VRP and a VRP with pickup and delivery. They proposed a cluster-first, route-second method based on the petal algorithm and the particle swarm optimization to improve the emergency response-time of medical service providers. In addition to the mentioned research, various studies in the literature address the transportation of patients and the planning of health care services in non-urgent situations in a DARP formulation. For example, they are transporting patients among hospitals, transporting a patient from home to a hospital, or transporting elderly people to their destination. Also, recently (Detti et al., 2017) formulated a multi-depot DARP in the healthcare realm under non-emergency situations and considering different features such as heterogeneous vehicles, vehicle-patient compatibility, etc. See (Parragh, 2011), (Parragh et al., 2012), (Coppi et al., 2013), (Marcon et al., 2017). We also refer to the study (Nable et al., 2016) for recent researches and trends in emergency medicine systems.

5.9 Mixed Integer Programming Trauma Network Optimization

In this section, we present a compact 2-index model and a 3-index partial path model for the VRP problem. The parameters and descriptions are listed in the table below.

Parameters **Description**

$P_{r,t}$	Set of patient nodes in region $r \in R$, of trauma level $t \in T$ (Set of pickup nodes, *{1…n}*)
$H_{r,t}$	Set of hospital nodes in region $r \in R$, of trauma level $t \in T$ (Set of delivery nodes, *{n+1..2n}*)
N_r	Set of all pickup and delivery nodes $N_r = P_r \cup D_r$, in region $r \in R$
$K_{r,d}$	set of all ambulances, K, in region $r \in R$, located at depot $d \in D$
$C_{r,h}$	The capacity of hospital $h \in H_r$ in $r \in R$
τ_k	Depot nodes that represent the start station of vehicle k, $k \in K$
τ'_k	Depot nodes that represent the end station of vehicle k, $k \in K$
V	Set of all nodes.
A	Set of (i, j) which is an arc from node i to node j, where $i, j \in V$
d_{ij}, t_{ij}	distance and travel time between node i and node j, for i and $j \in N$. Travel times satisfy the triangle inequality; $t_{ij} \leq t_{il} + t_{lj}$ $\forall\, i, j, l \in V$; and are nonnegative.
s_i	Fixed service time when visiting Patient i
e_i	Variable service time per item units of node i
$[a_i, b_i]$	Time windows when the visit at the particular location must start; a visit to node i can only take place between time a_i and b_i.
l_i	The quantity of goods to be loaded onto the vehicle at node i for $i \in P$

Decision Variables	**Description**
x_{ijk}	Binary variable where $x_{ijk} = 1(0)$ if vehicle k travels from node i to node j where $i, j \in V$, $k \in K$
S_{ik}	A nonnegative continuous variable that indicates when vehicle k starts the service at location i to node j where $i, j \in V$, $k \in K$
L_{ik}	A non-negative integer that is an upper bound on the quantity of goods on vehicle k after servicing node i where $i \in V$, $k \in K$. S_{ik} and L_{ik} are defined only when vehicle k visits node i.
z_i	A binary variable that indicates if request I is placed in the request bank where $i \in P$. The value is one if the request is placed in the request bank and zero otherwise.
$HC_{r,h}$	Integer variable counting the number of patients currently in hospital h, $h \in H_r$, in region $r \in R$.

5.9.1 2-index Formulation of the VRP

In the following model, let the cost of arc $(i, j) \in A$ be represented by both distance, d_{ij}, and time, t_{ij}. The objectives are to minimize the distance each client travels and the amount of time it takes each client to be served. This model combines the two-index model from (Bard, Kontoravdis, & Yu, 2002) with the constraints ensuring the time windows for the ATSP (Ascheuer et al., 2001) and is formulated as follows:

2-index formulation of the Model

Minimize

$$\boldsymbol{z_{LP}} = total\ travel\ time \tag{5.7}$$

subject to

$$\sum_{k \in K} \sum_{j \in N_r} x_{i,j}^{r,d,k} + z_i = 1 \qquad \forall\, r \in R, i \in P_r, d \in D_r \tag{5.8}$$

$$\sum_{j \in V_r} x_{i,j}^{r,d,k} - \sum_{j \in V_r} x_{j,n+1}^{r,d,k} = 0 \qquad \forall\, r \in R, i \in P_r, d \in D_r, k \in K_{r,d} \tag{5.9}$$

$$\sum_{j \in P_r \cup \{\tau'_{r,k}\}} x_{\tau_{r,k},j}^{r,d,k} = 1 \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d} \tag{5.10}$$

$$\sum_{j \in P_r \cup \{\tau_{r,k}\}} x_{j,\tau'_{r,k}}^{r,d,k} = 1 \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d} \tag{5.11}$$

$$\sum_{d \in D_r} \sum_{k \in K_{r,d}} x_{\tau_{r,k},i}^{r,d,k} \leq 1 \qquad \forall\, r \in R, i \in P_r \tag{5.12}$$

$$x_{i,j}^{r,d,k} = 1 \Rightarrow S_{r,d,i,k} + s_i + t_{i,j} \leq S_{r,d,j,k} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.13}$$

$$a_i \leq S_{r,d,i,k} \leq b_i \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.14}$$

$$S_{r,d,i,k} \leq S_{r,d,j,k} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.15}$$

$$x_{i,j}^{r,d,k} = 1 \Rightarrow L_{r,d,i,k} + l_i \leq L_{r,d,j,k} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.16}$$

$$L_{r,d,i,k} \leq 1 \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, i \in P_r \tag{5.17}$$

$$L_{r,d,\tau_k,k} = L_{r,d,\tau'_k,k} = 0 \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d} \tag{5.18}$$

$$x_{r,d,i,j,k} \in \{0,1\} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.19}$$

$$S_{r,d,i,k} \geq 0 \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, i \in V_r \tag{5.20}$$

$$L_{r,d,i,k} \in \{0,1\} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, i \in V_r \tag{5.21}$$

$$z_{r,i} \in \{0,1\} \qquad \forall\, r \in R, i \in P_r \tag{5.22}$$

The objective is to minimize the equally weighted objectives of the distance traveled, the total patient response time, and the penalty cost associated with the number of requests not scheduled.

Equation (5.8) ensures that each pickup location is visited or that the corresponding request is placed in the request bank. Equation (5.9) ensures that the delivery location is visited if the pickup location is visited and that the same vehicle is used. Equations (5.10) and (5.11) ensure that a vehicle leaves every start terminal, and a vehicle enters every end terminal. Together with equation (5.12), this ensures that consecutive paths between τ_k and τ'_k are formed for each vehicle $k \in K$. Equations (5.13) and (5.14) ensure that the variable, $S_{r,d,j,k}$, is tracking time correctly along the paths and that the time windows are obeyed. These constraints also ensure there are no sub tours. Equation (5.15) ensures that each pickup occurs before the corresponding delivery. Equations (5.16) - (5.18) ensure that the load variable is correctly determined along the paths and that the vehicle capacity constraints are enforced.

5.9.2 3-index Formulation of the VRP

(Petersen & Jepsen, 2009) present a solution methodology to the VRPTW by implementing bounded partial paths. The idea is to partition the problem such that the solution space is smaller than the original problem. This is done by splitting the larger tours into smaller segments and bounding the path length by the number of nodes. The number of visited customers is the bounding resource, which is particularly helpful since an ambulance can only visit a single patient at a time. For our scenario, the partial paths are restricted to an ambulance departing its depot, picking up a patient, arriving at the hospital, and returning to the original depot.

3-index formulation of the Model

Let $x_{i,j,l}^{r,d,k}$ be the variable indicating the use of arc $(i,j,l) \in A_r$. Problem (5.7) - (5.22) is rewritten to the following 3-index formulation in equations (5.23) - (5.38):

Minimize

$$z_1 = total\ travel\ time \tag{5.23}$$

subject to

$$\sum_{k \in K} \sum_{l \in L} \sum_{i \in N_k} x_{i,j,l}^{r,d,k} + z_i = 1 \qquad \forall\, r \in R, j \in P_r, d \in D_r \tag{5.24}$$

$$\sum_{k \in K} \sum_{(i,j) \in N_k} x_{i,j,l}^{r,d,k} \leq 1 \qquad \forall\, r \in R, d \in D_r, l \in L \tag{5.25}$$

$$\sum_{l \in L} \left(x_{i,i,l}^{r,d,k} - \sum_{j \in V_r} x_{i,j,l}^{r,d,k} \right) = \sum_{l \in L} \left(x_{i,i,l}^{r,d,k} - \sum_{j \in V_r} x_{i,j,l}^{r,d,k} \right) \quad \forall\, r \in R, i \in P_r, d \in D_r, k \in K_{r,d} \tag{5.26}$$

$$\sum_{(j,i) \in P_r \cup \{\tau'_{r,k}\}} x_{j,i,l}^{r,d,k} = \sum_{(i,j) \in P_r \cup \{\tau_{r,k}\}} x_{i,j,l}^{r,d,k} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, l \in L \tag{5.27}$$

$$\sum_{l\in L}\sum_{(i,j)\in A_r} x_{i,j,l}^{r,d,k} = len(P_r) \quad \forall\, r\in R, d\in D_r, k\in K_{r,d} \tag{5.28}$$

$$\sum_{i\in P_r}\sum_{(j,l)\in A_r} x_{i,j,l}^{r,d,k} \le 4 \quad \forall\, r\in R, d\in D_r, k\in K_{r,d} \tag{5.29}$$

$$x_{i,j,l}^{r,d,k} = 1 \Rightarrow S_{i,j,l}^{r,d,k} + s_i + t_{i,j} \le S_{i,j,l}^{r,d,k} \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, (i,j,l)\in A_r \tag{5.30}$$

$$a_i \le S_{i,j,l}^{r,d,k} \le b_i \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, (i,j,l)\in A_r \tag{5.31}$$

$$x_{i,j,l}^{r,d,k} = 1 \Rightarrow L_{r,d,i,k} + l_i \le L_{r,d,j,k} \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, (i,j,l)\in A_r \tag{5.32}$$

$$L_{r,d,i,k} \le 1 \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, i\in P_r \tag{5.33}$$

$$L_{r,d,\tau_k,k} = L_{r,d,\tau'_k,k} = 0 \quad \forall\, r\in R, d\in D_r, k\in K_{r,d} \tag{5.34}$$

$$x_{i,j,l}^{r,d,k} \in \{0,1\} \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, (i,j,l)\in A_r \tag{5.35}$$

$$S_{i,j,l}^{r,d,k} \ge 0 \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, (i,j,l)\in A_r \tag{5.36}$$

$$L_{r,d,i,k} \in \{0,1\} \quad \forall\, r\in R, d\in D_r, k\in K_{r,d}, i\in V_r \tag{5.37}$$

$$z_{r,i} \in \{0,1\} \quad \forall\, r\in R, i\in P_r \tag{5.38}$$

Constraints (5.24) ensures that all patients are visited exactly once, while the redundant constraints (5.25) ensures that no customer is visited more than once. Constraints (5.26) maintains flow conservation between the original nodes V. Constraints (5.27) maintain flow conservation with a layer. Constraints (5.28) ensure that enough partial paths are selected to service all of the patients, and constraints (5.29) limit the length of the partial path to at most four nodes. For this scenario, that allows an ambulance to depart the depot, arrive at the patient, deliver the patient to the hospital, and then return to the depot. Constraints (5.30) - (5.34) enforce resource limitations.

After the re-formulation, we use a Dantzig-Wolfe decomposition to reach the following MP and pricing problems. Although the pricing problems are identical for each region, the inputs are significantly different, so we cannot simplify the workload by combining it into a single pricing problem and must keep them separated by region.

Let λ_q be a binary variable indicating where partial path $q \in Q_r$ is used. We use Dantzig-Wolfe decomposition where the constraints (5.31) - (5.34) are kept in the MP. Since the vehicles are identical, we can aggregate over the sets to get the following MP:

Minimize

$$\boldsymbol{z}_{\mathbf{PP}} = total\ travel\ time \tag{5.39}$$

subject to

$$\sum_{q \in Q_r} \alpha_{i,j}^q \lambda_{r,q} + z_i = 1 \qquad \forall\, r \in R, (i,j) \in A_r \tag{5.40}$$

$$\sum_{q \in Q_r:\, \tau = i} \lambda_{r,q} - \sum_{q \in Q_r:\, \tau\prime = i} \lambda_{r,q} = 0 \qquad \forall\, r \in R, i \in V_r \tag{5.41}$$

$$\sum_{q \in Q_r} \lambda_{r,q} \leq len(P_r) \qquad \forall\, r \in R \tag{5.42}$$

$$\alpha_{i,j}^q \lambda_{r,q} = 1 \Rightarrow S_{i,j}^{r,d,k} + s_i + t_{i,j} \leq S_{i,j}^{r,d,k} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.43}$$

$$a_i \sum_{q \in Q_r} \alpha_{i,j}^q \lambda_{r,q} \leq S_{i,j}^{r,d,k} \leq b_i \sum_{q \in Q_r} \alpha_{i,j}^q \lambda_{r,q} \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, (i,j) \in A_r \tag{5.44}$$

$$\lambda_{r,q} \in \{0,1\} \qquad \forall\, r \in R, q \in Q_r \tag{5.45}$$

$$S_{r,d,i,k} \geq 0 \qquad \forall\, r \in R, d \in D_r, k \in K_{r,d}, i \in V_r \tag{5.46}$$

In this formulation, $\alpha_{i,j}^q$, is the number of times arc $(i,j) \in A_r$ is used on path $q \in Q_r$ and $\tau_{r,q}$ and $\tau'_{r,q}$ respectively indicate the start and end node of partial path $q \in Q_r$. Constraint (5.40) ensures that each customer is visited exactly once. Constraint (5.41) ensures flow conservation and links the partial paths together. Constraint (5.42) is the convexity constraint and ensures that the number of partial paths selected is equal to the number of patients. Constraints (5.43) and (5.44) enforce resource windows.

Before discussing the pricing problem, we review the following theorems and proofs about the tightness of the bounds obtained by the decomposition. The following theorems and proofs were presented in (Petersen, 2011) to demonstrate that the Dantzig Wolfe decomposition of the 3-index partial path formulation provides a higher quality bound than the partial path solution.

Theorem 1: Let z_{LP} (5.7) be an LP-solution to (5.8) - (5.22) and let z_{PP} (5.39) be an LP-solution to (5.40) - (5.45) then $z_{LP} \leq z_{PP}$ for all instances of VRP.

Proof: Since all solutions to (5.39) - (5.45) map to solutions to (5.7) - (5.22), then $z_{LP} \leq z_{PP}$, originally demonstrated in (Wolsey & Nemhauser, 1999).

Theorem 2: Let z_{PP} (5.39) be an LP-solution to (5.40) - (5.45), and z_{EP} is the LP-solution to the classical decomposition of VRP into an elementary route for each vehicle. Then instances exist where $z_{EP} < z_{PP}$.

Proof: This can be shown through a simple yet effective proof by example, again demonstrated in (Petersen, 2011). For this proof, Peterson constructs an instance with three customers, each with a demand of 1 and vehicle capacity Q = 2, see Figure 5.5 below. There are six feasible routes ({0, 1, 0}, {0, 2, 0}, {0, 3, 0}, {0, 1, 2, 0}, {0, 1, 3, 0}, {0, 2, 3, 0}) having the costs (4, 2, 4, 3, 4, 3). The LP solution is (0, 0, 0, .5, .5, .5) with objective $z_{EP} = 5$. Using the partial path formulation with max path length L=3 and K=1 we find the optimal solution ({0,1,3,0,2,0}) with objective $z_{pp} = 6$, thereby demonstrating $z_{EP} < z_{PP}$.

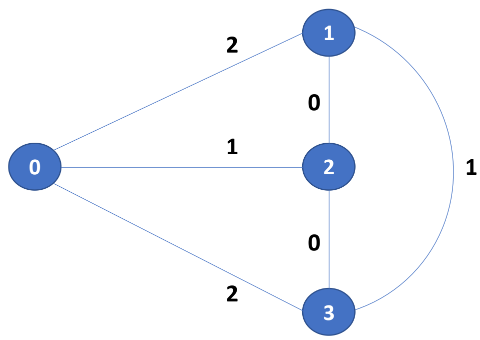

Figure 5.5 – There are three customers with a demand of 1 and vehicle capacity Q = 2 (Petersen, 2011).

Considering the column generation approach in (Baldacci et al., 2008), where columns are enumerated dependent on strong upper and lower bounds, it should be clear that the partial path approach should contain fewer enumerated columns due to the smaller solution space of the pricing problem. An additional improvement is provided by solving the decomposition of the partial path problem. A powerful strategy should be obtained by combining the relatively strong bound with the small solution space.

To adjust the 3-index VRP into the ambulance routing problem, we modify our existing parameters and notation and add additional constraints. For the ARP, the issue we face in the formulation is that the system evolves in continuous time. The amount of patients that can be delivered to a hospital at a particular point in time depends on the hospital's trauma patient capacity and inventory at that point in time (which depends on the initial number of patients, patient treatment time, and the time elapsed since the start of the planning period). Consequently, delivery times either must be scheduled carefully, or patients potentially have to rerouted to other hospitals.

This is contrasted with most other inventory routing problems (IRPs). The planning horizon is partitioned into periods, and deliveries take place at the start of the period, and consumption occurs at the end. However, the patient movement takes place continuously throughout the day, and a continuous-time variant of the IRP is most appropriate (Lagos et al., 2020). In our scenario, we are developing a MIP over a given, uniform, discretization of time. Like the CIRP presented in the article, vehicle routes/patient routes are not restricted to start and end in a single time period.

Our vehicles and patients rely on continuous-time movements, but for ease, our hospital inventory is tracked on a discrete-time interval. In this research, we will not set up a true continuous-time routing problem or even a partially time-expanded formulation. Future work will show that there does exist a discretization of time such that an optimal solution to a time-expanded network formulation using the discretization results in a continuous time-optimal solution. Instead, we define N as the full length of the time interval and δ as the length of the time interval. So $N = I \times \delta$ for some positive integer I, so I periods of length δ. Recall decision variable $HC_{vi}^{r,h}$ previously defined as integer variable counting the number of *patients* currently in hospital *h*, $h \in H_r$, in region $r \in R$, now in period *ni* $\in I$. We first modify constraint (5.9) above to account for specific patient trauma level requirements, now constraint (5.47). We add in the trauma level index t, $t \in [1..4]$ to the patients and to the hospitals.

Below we present the additional constraints to be joined with constraints (5.23) - (5.38) to form the 3-index partial path ARP.

$$\sum_{\substack{h \in H_r \\ \text{If } \delta(vi) < i\ incident \text{ time} \leq \delta(vi+1)}} \sum_{i \in P_r} \left(x_{d,i,h}^{r,d,k}\right) \leq K_{r,d} \quad \forall\ vi \in I, r \in R, d \in D_r, k \in K_{r,d} \tag{5.47}$$

$$HC_0^{r,h} = 0 \qquad \forall\ r \in R, h \in H_r \tag{5.48}$$

$$HC_{(vi-1)}^{r,h} + \underbrace{\sum_{k \in K_{r,d}} \sum_{i \in P_r} \sum_{d \in D_r} \left(x_{d,i,h}^{r,d,k}\right)}_{\text{If } i \text{ delivery time} \leq \delta(vi)} - \underbrace{\sum_{k \in K_{r,d}} \sum_{i \in P_r} \sum_{d \in D_r} \left(x_{i,h,d}^{r,d,k}\right)}_{\text{If } i \text{ treatment time} \leq \delta \times (vi)} \leq HC_{vi}^{r,h} \tag{5.49}$$

$$\forall\ vi \in I, r \in R, h \in H_r$$

$$HC_{vi}^{r,h} \leq C_{r,h,vi} \qquad \forall\ vi \in I, \in R, h \in H_r \tag{5.50}$$

$$HC_{vi}^{r,h} \in \{0,1\} \qquad \forall\ vi \in I, \in R, h \in H_r \tag{5.51}$$

Equation (5.9) is modified to (5.47) enforce trauma level requirements for the patient pickups by requiring certain trauma level patients to be sent to the appropriate facilities. In Equation (5.48), we initialize the hospital patient inventory variable at initial time 0. Equation (5.49) modifies the current period hospital patient inventory based on the number of patients that have arrived before the end of the time window subtract the number of patients that have been treated and moved to a non-trauma bed or have been discharged from the hospital before the end of the time window. Equation (5.50) ensures that the hospital patient inventory does not exceed the number of trauma beds available at that hospital.

From here, we perform the Dantzig-Wolfe decomposition, as previously mentioned. The formulation uses the same equations from (5.40) - (5.46). Again, let $\lambda_{r,q}$ be a binary variable indicating where partial path $q \in Q_r$ in region $r \in R$ is used.

$$\underset{\text{If } \delta(vi) < i\ incident \text{ time} \leq \delta(vi+1)}{\sum_{i \in P_r} \sum_{q \in Q_r} \alpha_{d,i}^{q} \left(\lambda_{r,q}\right)} \leq K_{r,d} \qquad \forall\ vi \in I, r \in R, d \in D_r \tag{5.52}$$

$$HC_0^{r,h} = 0 \qquad \forall\ r \in R, h \in H_r \tag{5.53}$$

$$HC_{(vi-1)}^{r,h} + \underset{\text{If } i \text{ delivery time} \leq \delta(vi)}{\sum_{i \in P_r} \sum_{q \in Q_r} \left(\alpha_{i,h}^{q} \lambda_{r,q}\right)} - \underset{\text{If } i \text{ treatment time} \leq \delta \times (vi)}{\sum_{i \in P_r} \sum_{q \in Q_r} \left(\alpha_{h,i}^{q} \lambda_{r,q}\right)} \leq HC_{vi}^{r,h} \tag{5.54}$$

$$\forall\ vi \in I, r \in R, h \in H_r$$

$$HC_{vi}^{r,h} \leq C_{r,h,vi} \qquad \forall\ vi \in I, \in R, h \in H_r \tag{5.55}$$

Lastly, we move on to the final formulation of the Resource Allocation ARP (RAARP). For the RAARP, we add an indicator variable representing whether or not an investment is selected, $rs_{r,n}\ \forall\ r \in R, n \in IN$, where *n* is a specific investment from the list of investments, *IN*, for that region *r*. The previous ARP constraints are then modified to include the respective changes if a resource is selected. An example of introducing investment 1, instituting a prevention program, is present below in equations (5.56) - (5.59).

$$\text{If } \left(rs_{r,1} = 0\right) \longrightarrow \sum_{q \in Q_r} \alpha_{i,j}^{q} \lambda_{r,q} + z_i = 1 \qquad \forall\ r \in R, (i,j) \in\ A_{r,All\ Patients} \tag{5.56}$$

$$\text{If } \left(rs_{r,1} = 1\right) \longrightarrow \sum_{q \in Q_r} \alpha_{i,j}^{q} \lambda_{r,q} + z_i = 1 \qquad \forall\ r \in R, (i,j) \in A_{r,Reduced\ Patients} \tag{5.57}$$

$$\text{If } \left(rs_{r,1} = 0\right) \longrightarrow\ \sum_{q \in Q_r}\ \lambda_{r,q} \leq len(P_{r,All\ Patients}) \qquad \forall\ r \in R \tag{5.58}$$

$$\text{If } (rs_{r,1} = 1) \rightarrow \sum_{q \in Q_r} \lambda_{r,q} \leq len(P_{r,Reduced\ Patients}) \qquad \forall\, r \in R \quad (5.59)$$

The same type of constraint modification is applied to the other constraints when accounting for the other types of resource allocation, such as adding an ambulance to a depot or upgrading equipment at a specific hospital. Any nonlinearities that are encountered from combining the resource allocation with the ARP can be resolved using the same technique discussed in Section 4.7.

5.10 Empirical Results

This section describes the computational test problems used to evaluate the proposed solution approach. The integer programs were generated in Python 3.7.3 and solved with Gurobi 8.1. The Gurobi parameters were kept at their default values. The computational experiments were conducted on the Georgia Institute of Technology High Throughput server cluster on CPUs with 64GB of RAM.

As in the previous chapters, we ran the multi-swarm PSO, the Column Generation with the MSPSO Pricing Problem, and the traditional Column Generation formulation. The objective solution results are displayed below in Table 5.7. Similar to the previous TSA and CBP models, we see the standard MSPSO performs the worst. Some of this performance could be improved by increasing the number of subswarms. The CGMSPSO performs much better than the MSPSO and shows significant improvements to the solution.

Table 5.7 - Minimum Total Distance Traveled and Travel Time

	Distance in miles		Time in minutes	
	Depot to Patient	Patient to Hospital	Depot to Patient	Patient to
MSPSO	9487.53	10074.45	25300.08	26865.21
CGMSPSO	6160.44	6318.96	16427.83	16850.57
CG	1442.56	4524.49	3846.827	12065.3

Lastly, we look at the investment selection variables. Although the minimum time and distance results were greatly different, the investment selection results were nearly the same across the methods, seen in Table 5.8. Regardless of the routes that became available during the PSO methods, the problem still selects those routes that minimize the travel time and travel distance and will select the resources that will allow them to reduce even more. This is reassuring because less solution time can be used to retrieve very similar results to the resource allocation or investment selection which is our focus. Both the full CG and CGMPSO chose to add ambulances and institute prevention methods and techniques in all regions. They also both chose to upgrade trauma centers in the same three regions. The only difference was one of the regions in which to add upgraded equipment. The standard MSPSO has a number of differences but still produces satisfactory resource allocation results.

Table 5.8 - Investment Selection Variable Results

	Upgrade	Prevention	Vehicle	Equipment	Upgrade	Prevention	Vehicle	Equipment
CG	0	1	1	0	$ -	$ 90.00	$180.00	$ -
$3,550.00	1	1	1	0	$250.00	$ 90.00	$180.00	$ -
Total Cost	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	1	1	1	1	$250.00	$ 90.00	$180.00	$ 50.00
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	1	1	1	0	$250.00	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	1	$ -	$ 90.00	$180.00	$ 50.00
CGPSO	0	1	1	1	$ -	$ 90.00	$180.00	$ 50.00
$3,550.00	1	1	1	0	$250.00	$ 90.00	$180.00	$ -
Total Cost	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	1	1	1	1	$250.00	$ 90.00	$180.00	$ 50.00
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	1	1	1	0	$250.00	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
PSO	0	1	1	1	$ -	$ 90.00	$180.00	$ 50.00
$3,170.00	1	1	1	0	$250.00	$ 90.00	$180.00	$ -
Total Cost	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	1	1	1	1	$250.00	$ 90.00	$180.00	$ 50.00
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	1	0	0	0	$250.00	$ -	$ -	$ -
	0	1	1	0	$ -	$ 90.00	$180.00	$ -
	0	1	0	0	$ -	$ 90.00	$ -	$ -
	1	1	0	0	$250.00	$ 90.00	$ -	$ -

5.11 Conclusions

The results are subject to change for a different trauma system, with different parameters, cost structures, and submissions. However, the most important contribution of this study is that it offers a framework of investment allocation for trauma systems and possible ways to evaluate the impact of the investments on the overall performance of the trauma system. It is basically a top-down approach on a strategic level, but it uses the tactical level decisions in order to evaluate several strategies to improve the system. Simulation is a powerful tool to perform a thorough analysis and systematic update of the system with given investments and facilitates the decision-making process of decision-makers in the trauma network.

The procedure described in the Results section reduces the number of candidate portfolios to a reasonable magnitude via exploiting the existence of Pareto efficient frontiers of the values obtained for specified metrics. This type of approach can be applied for a broad class of problems involving the selection of investments by considering their impact of certain performance measures. The procedure can be extended via increasing the number of performance measures (or in other words increasing the dimension size of the problem), via assigning a certain utility function dependent on the performance metrics for the patient and performing the elimination process with functions, or via increasing the time span of interest in the problem and adding a dynamical perspective to the selection of best portfolios. It is important to realize that this simulation was intended as a proof of concept to demonstrate that improvements to patient well-being can be measured in some manner and that alternative solutions can be analyzed. While most of the results from the simulation runs were fairly intuitive, there is a possibility that these arguments are not valid in a different setting (different distribution of location and time parameters). The validity and strength of these intuitive arguments can be tested through additional sensitivity analysis.

The RAARP model presented above allows us to approach the emergency trauma network problem to find an exact solution. We also implement the CGMSPSO solution technique to find good solutions in a shorter amount of time than both the traditional CG method and the trauma simulation.

Future research should involve the application of this modeling technique using real-world inputs for the portfolio options and the investment amount. This has the ability to produce potentially more interesting results based on limited funding and the different types of investment available. Other research should investigate such scenarios as the difference in portfolios dedicated to urban trauma system upgrades versus rural trauma system upgrades. Lastly, a significant area of interest is pediatric trauma care due to severely constrained resources. Dedicating a trauma simulation model to measuring how to improve pediatric trauma care could be very impactful.

CHAPTER 6. CONCLUSION

6.1 Summary

The focus of this dissertation is on public sector resource allocation problems and ow to solve them in the context of column generation and particle swarm optimization. The emphasis has been on large scale versions of the problem that lend themselves to Dantzig-Wolfe decomposition and column generation type algorithms.

We have demonstrated a consistent and practical solution methodology for resource allocation mixed-integer programs. In order to solve problems with an exponential number of decision variables, we employ Dantzig-Wolfe decomposition to take advantage of the special subproblem structures encountered in resource allocation problems. In each of the resource allocation problems presented, we concentrate on selecting an optimal portfolio of improvement measures. We explore utilizing multi-swarm particle swarm optimization to solve the decomposition heuristically. We also explore integrating multi-swarm PSO into the column generation framework to solve the pricing problem for entering columns of negative reduced cost.

We present a TSA problem to allocate security measures across all federally funded airports nationwide. This project establishes a quantitative construct for enterprise risk assessment and optimal resource allocation to achieve the best aviation security. We analyzed and modeled the various aviation transportation risks and established their interdependencies. The model selects optimal security measure allocations for each airport with the objectives to minimize the probability of false clears, maximize the probability of threat detection, and maximize the risk posture (ability to mitigate risks) in aviation security. The risk assessment and optimal resource allocation construct are generalizable and are applied to the CBP problem.

We optimize security measure investments to achieve the most cost-effective deterrence and detection capabilities for the CBP. A large-scale resource allocation integer program was successfully modeled that rapidly returns good Pareto optimal results. The model incorporates the utility of each measure, the probability of success, along with multiple objectives. To the best of our knowledge, our work presents the first mathematical model that optimizes security strategies for the CBP and is the first to introduce a utility factor to emphasize deterrence and detection impact. The model accommodates different resources, constraints, and various types of objectives.

We analyze the emergency trauma network problem first by simulation. The simulation offers a framework of resource allocation for trauma systems and possible ways to evaluate the impact of the investments on the overall performance of the trauma system. The simulation works as an effective proof of concept to demonstrate that improvements to patient well-being can be measured and that alternative solutions can be analyzed. We then explore three different formulations to model the Emergency Trauma Network as a mixed-integer programming model. The first model is a Multi-Region, Multi-Depot, Multi-Trip Vehicle Routing Problem with Time Windows. This is a known expansion of the vehicle routing problem that has been extended to model the Georgia trauma network. We then adapt an Ambulance Routing Problem (ARP) to the previously mentioned VRP.

There are no known ARPs of this magnitude/extension of a VRP. One of the primary differences is many ARPs are constructed for disaster scenarios versus day-to-day emergency trauma operations. The new ARP also implements more constraints based on trauma level limitations for patients and hospitals. Lastly, the Resource Allocation ARP is constructed to reflect the investment decisions presented in the simulation.

6.2 Conclusion

With the empirical results demonstrated with the scenarios, we have shown that the multi-swarm PSO is an effective solution technique for solving these large-scale resource allocation problems. We have also demonstrated that embedding the multi-swarm PSO into the column generation framework to solve the pricing problem is more effective still. The solution times for the new column generation multi-swarm pricing problem are typically much faster than the standard column generation due to reducing the number of subproblems to be solved.

6.3 Future Research

Many companies and industries face problems that can be defined in a resource allocation formulation. Often these problems are large in inputs, number of constraints, and decision variables. The practical nature of these problems often utilizes multiple objectives. Due to conflict between objectives, finding a feasible solution that simultaneously optimizes all objectives is usually impossible. Decision-makers also want to explore and understand the trade-off between objectives before choosing a suitable solution. Thus, we would like to explore generating many or all efficient solutions to these large-scale resource allocation problems.

REFERENCES

A Cameron, P., & Zalstein, S. (1998, 12). Transport of the critically ill. *The Medical Journal of Australia, 169*, 610-1.

Acosta, C. D., Kit Delgado, M., Gisondi, M. A., Raghunathan, A., D'Souza, P. A., Gilbert, G., . . . Wang, N. E. (2010). Characteristics of Pediatric Trauma Transfers to a Level I Trauma Center: Implications for Developing a Regionalized Pediatric Trauma System in California. *Academic Emergency Medicine, 17*, 1364-1373. doi:10.1111/j.1553-2712.2010.00926.x

Agliari, E., Barra, A., Galluzzi, A., Guerra, F., Tantari, D., & Tavani, F. (2015). Retrieval capabilities of hierarchical networks: from Dyson to Hopfield. *Physical review letters, 114*, 028103.

Agliari, E., Barra, A., Galluzzi, A., Guerra, F., Tantari, D., & Tavani, F. (2015). Topological properties of hierarchical networks. *Physical Review E, 91*, 062807. doi:10.1103/physreve.91.062807

Aguilar, D. (2018, 1 30). Border Security: Back to Basics with Personnel, Technology, Infrastructure. Retrieved from https://www.hstoday.us/subject-matter-areas/border-security/border-security-back-basics-personnel-technology-infrastructure/

Ai, J., Brockett, P. L., & Wang, T. (2017). Optimal enterprise risk management and decision making with shared and dependent risks. *Journal of Risk and Insurance, 84*, 1127-1169.

Ai, J., Brockett, P. L., Cooper, W. W., & Golden, L. L. (2012). Enterprise risk management through strategic allocation of capital. *Journal of Risk and Insurance, 79*, 29-56.

Ai, T. J., & Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research, 36*, 1693-1702.

Ai, T. J., & Kachitvichyanukul, V. (2009). Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem. *Computers & Industrial Engineering, 56*, 380-387.

Alaia, E. B., Dridi, I. H., Bouchriha, H., & Borne, P. (2015). Genetic algorithm for multi-criteria optimization of multi-depots pick-up and delivery problems with time windows and multi-vehicles. *Acta Polytechnica Hungarica, 12*, 155-174.

Alam, M. N. (2016). Particle swarm optimization: Algorithm and its codes in MatLab. *ResearchGate*, 1-10.

Alden, E. (2017). Is Border Enforcement Effective? What We Know and What It Means. *Journal on Migration and Human Security*. doi:10.14240/jmhs.v5i2.94

Anagnostopoulos, K. P., & Mamanis, G. (2010). A portfolio optimization model with three objectives and discrete variables. *Computers & Operations Research, 37*, 1285-1297. doi:10.1016/j.cor.2009.09.009

Anagnostopoulos, K. P., & Mamanis, G. (2011). The mean--variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms. *Expert Systems with Applications, 38*, 14208-14217. doi:10.1016/j.eswa.2011.04.233

Andersson, T., & Värbrand, P. (2007). Decision support tools for ambulance dispatch and relocation. *Journal of the Operational Research Society, 58*, 195-201.

Angelelli, E., Mansini, R., & Speranza, M. G. (2008). A comparison of MAD and CVaR models with real features. *Journal of Banking & Finance, 32*, 1188-1197. doi:10.1016/j.jbankfin.2006.07.015

Angelelli, E., Mansini, R., & Speranza, M. G. (2012). Kernel search: A new heuristic framework for portfolio selection. *Computational Optimization and Applications, 51*, 345-361.

Appelgren, L. H. (1969). A column generation algorithm for a ship scheduling problem. *Transportation Science, 3*, 53-68.

Ascheuer, N., Fischetti, M., & Grötschel, M. (2001). Solving the asymmetric traveling salesman problem with time windows by branch-and-cut. *Mathematical Programming, 90*, 475-506.

At border summit, CBP leaders describe strained resources. (n.d.). Retrieved from https://www.cbp.gov/newsroom/spotlights/border-summit-cbp-leaders-describe-strained-resources

Attagara, J. (2006). *The Explosive Scanning Devices Allocation Problem for Airport Security Systems.* Ph.D. dissertation, Texas Tech University.

Babu, V. L., Batta, R., & Lin, L. (2006). Passenger grouping under constant threat probability in an airport security system. *European Journal of Operational Research, 168*, 633-644.

Baldacci, R., Bodin, L., & Mingozzi, A. (2006). The multiple disposal facilities and multiple inventory locations rollon--rolloff vehicle routing problem. *Computers & Operations Research, 33*, 2667-2702.

Baldacci, R., Christofides, N., & Mingozzi, A. (2008). An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming, 115*, 351-385.

Banham, R. (2005). Enterprising views of risk management. *Articles of Merit Award Program for Distinguished Contribution to Management Accounting*, 14.

Bard, J. F., Kontoravdis, G., & Yu, G. (2002). A branch-and-cut procedure for the vehicle routing problem with time windows. *Transportation Science, 36*, 250-269.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., & Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations research, 46*, 316-329.

Barringer, M. L., Thomason, M. H., Kilgo, P., & Spallone, L. (2006). Improving outcomes in a regional trauma system: impact of a level III trauma center. *The American Journal of Surgery, 192*, 685-689. doi:https://doi.org/10.1016/j.amjsurg.2005.11.006

Belaid, I., & Eyraud-Dubois, L. (2015). Column generation integer programming for allocating jobs with periodic demand variations. *International Workshop on Algorithmic Aspects of Cloud Computing*, (pp. 35-48).

Berbeglia, G., Cordeau, J.-F., & Laporte, G. (2010). Dynamic pickup and delivery problems. *European journal of operational research, 202*, 8-15.

Berbeglia, G., Cordeau, J.-F., Gribkovskaia, I., & Laporte, G. (2007). Static pickup and delivery problems: a classification scheme and survey. *Top, 15*, 1-31.

Best, M. J., & Hlouskova, J. (2005, 11). An algorithm for portfolio optimization with transaction costs. *Management Science, 51*, 1676-1688. doi:10.1287/mnsc.1050.0418

Bettinelli, A., Ceselli, A., & Righini, G. (2014). A branch-and-price algorithm for the multi-depot heterogeneous-fleet pickup and delivery problem with soft time windows. *Mathematical Programming Computation, 6*, 171-197.

Bienstock, D. (1996). Computational study of a family of mixed-integer quadratic programming problems. *Mathematical programming, 74*, 121-140. doi:10.1007/bf02592208

Blackwell, T., Kellam, J. F., & Thomason, M. (2003). Trauma care systems in the United States. *Injury, 34*, 735-739. doi:https://doi.org/10.1016/S0020-1383(03)00152-9

Blanchfield, C., & Kolowratnik, N. V. (n.d.). Assessing Surveillance: Infrastructures of Security in the Tohono Oʻodham Nation. Retrieved from https://archinect.com/features/article/150049769/assessing-surveillance-infrastructures-of-security-in-the-tohono-o-odham-nation

Boland, N., Charkhgard, H., & Savelsbergh, M. (2017). The quadrant shrinking method: A simple and efficient algorithm for solving tri-objective integer programs. *European Journal of Operational Research, 260*, 873-885.

Bolton, J. D. (2017). *Airport Security Investment Model.* Master's thesis, University of Arkansas.

Border Patrol Sectors. (n.d.). Retrieved from https://www.cbp.gov/border-security/along-us-borders/border-patrol-sectors

Borhanifar, Z., & Shadkam, E. (2016). The new hybrid COAW method for solving multi-objective problems. *arXiv preprint arXiv:1611.00577*.

Brathwaite, C., Rosko, M., McDowell, R., Gallagher, J., Proenca, J., & Spott, M. A. (1998, 7). A Critical Analysis of On-Scene Helicopter Transport on Survival in a Statewide Trauma System. *The Journal of Trauma, 45*, 140-4; discussion 144.

Bristow, J. A. (2017). An Arizona border Wall Case Study.

Broyden, C. G. (1970). The convergence of a class of double-rank minimization algorithms 1. general considerations. *IMA Journal of Applied Mathematics, 6*, 76-90.

Bruck, B. P., Santos, A. G., & Arroyo, J. E. (2012). Hybrid metaheuristic for the single vehicle routing problem with deliveries and selective pickups. *2012 IEEE Congress on Evolutionary Computation*, (pp. 1-8).

Cavusoglu, H., Koh, B., & Raghunathan, S. (2010). An analysis of the impact of passenger profiling for transportation security. *Operations Research, 58*, 1287-1302.

Cavusoglu, H., Kwark, Y., Mai, B., & Raghunathan, S. (2013). Passenger profiling and screening for aviation security in the presence of strategic attackers. *Decision Analysis, 10*, 63-81.

Chambers, S. N., Boyce, G. A., Launius, S., & Dinsmore, A. (2019). Mortality, Surveillance and the Tertiary “Funnel Effect” on the US-Mexico Border: A Geospatial

Modeling of the Geography of Deterrence. *Journal of Borderlands Studies*, 1-26.

Chang, T.-J., Meade, N., Beasley, J. E., & Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research, 27*, 1271-1302. doi:10.1016/s0305-0548(99)00074-x

Chiodi, L., Mansini, R., & Speranza, M. G. (2003). Semi-absolute deviation rule for mutual funds portfolio selection. *Annals of Operations Research, 124*, 245-265. doi:10.1023/b:anor.0000004772.15447.5a

Chukwu, W. I., Uka, P. N., & Dike, A. O. (2014). Duality Theory in Multi Objective Linear Programming Problems. *International Journal of Engineering and Science, 3*.

Commission, G. T. (2009, 2). Five Year Georgia Trauma System Strategic Plan. Retrieved from http://www.gahighwaysafety.org/shsp/gtcncfiveyear.pdf

Coppi, A., Detti, P., & Raffaelli, J. (2013). A planning and routing model for patient transportation in health care. *Electronic notes in discrete mathematics, 41*, 125-132.

Cordeau, J.-F., & Laporte, G. (2003). A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological, 37*, 579-594.

Cordeau, J.-F., Laporte, G., & Ropke, S. (2008). Recent models and algorithms for one-to-one pickup and delivery problems. In *The vehicle routing problem: latest advances and new challenges* (pp. 327-357). Springer.

Council, N. R., & others. (2010). *Review of the Department of Homeland Security's approach to risk analysis.* National Academies Press.

Crama, Y., & Schyns, M. (2003). Simulated annealing for complex portfolio selection problems. *European Journal of operational research, 150*, 546-571. doi:10.1016/s0377-2217(02)00784-1

Cura, T. (2009). Particle swarm optimization approach to portfolio optimization. *Nonlinear analysis: Real world applications, 10*, 2396-2406. doi:10.1016/j.nonrwa.2008.04.023

Customs, U. S., & Protection, B. (2016). Cross-Border Tunnels and Border Tunnel Prevention. *Fiscal Year 2015 Report to Congress*.

Dantzig, G. B., & Wolfe, P. (1960). Decomposition principle for linear programs. *Operations research, 8*, 101-111.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation, 6*, 182-197. doi:10.1109/4235.996017

Detti, P., Papalini, F., & Lara, G. Z. (2017). A multi-depot dial-a-ride problem with heterogeneous vehicles and compatibility constraints in healthcare. *Omega, 70*, 1-14.

Diana, M., & Dessouky, M. M. (2004). A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows. *Transportation Research Part B: Methodological, 38*, 539-557.

Dillon, R. L., Burns, W. J., & John, R. S. (2018). Insights for Critical Alarm-Based Warning Systems from a Risk Analysis of Commercial Aviation Passenger Screening. *Decision Analysis*.

Dragomir, A. G., Nicola, D., Soriano, A., & Gansterer, M. (2018). Multidepot pickup and delivery problems in multiple regions: a typology and integrated model. *International Transactions in Operational Research, 25*, 569-597.

Dreżewski, R., & Doroz, K. (2017). An Agent-Based Co-Evolutionary Multi-Objective Algorithm for Portfolio Optimization. *Symmetry, 9*, 168. doi:10.3390/sym9090168

Dunn, M. J., Gwinnutt, C. L., & Gray, A. J. (2006, 9). Critical care in the emergency department: patient transfer. *Emergency Medicine Journal : EMJ, 24*, 40-44. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2658153/

Eberhart, R., & Kennedy, J. (1995). Particle swarm optimization. *Proceedings of the IEEE international conference on neural networks*, *4*, pp. 1942-1948.

Engelbrecht, A. P. (2010). Heterogeneous particle swarm optimization. *International Conference on Swarm Intelligence*, (pp. 191-202).

Fernández, A., & Gómez, S. (2007). Portfolio selection using neural networks. *Computers & Operations Research, 34*, 1177-1191. doi:10.1016/j.cor.2005.06.017

Fletcher, K. C. (2011). *Aviation security: a case for risk-based passenger screening.* Master's thesis, Monterey, California. Naval Postgraduate School.

Fletcher, K. C., & Abbas, A. E. (2018). A Value Measure for Public-Sector Enterprise Risk Management: A TSA Case Study. *Risk Analysis, 38*, 991-1008.

Ford Jr, L. R., & Fulkerson, D. R. (1958). A suggested computation for maximal multicommodity network flows. *Management Science, 5*, 97-101.

Framework, I. (2004). Enterprise Risk Management-Integrated Framework.

Garcia-Nieto, J., & Alba, E. (2012). Why six informants is optimal in PSO. *Proceedings of the 14th annual conference on Genetic and evolutionary computation*, (pp. 25-32).

Gholami-Zanjani, S. M., Pishvaee, M. S., & Torabi, S. A. (2018). OR Models for Emergency Medical Service (EMS) Management. In *Operations Research Applications in Health Care Management* (pp. 395-421). Springer.

Gillen, D., & Morrison, W. G. (2015). Aviation security: costing, pricing, finance and performance. *Journal of Air Transport Management, 48*, 1-12.

Gilli, M., & Këllezi, E. (2002). A global optimization heuristic for portfolio choice with VaR and expected shortfall. In *Computational methods in decision-making, economics and finance* (pp. 167-183). Springer.

Gilmore, P. C., & Gomory, R. E. (1961). A linear programming approach to the cutting-stock problem. *Operations research, 9*, 849-859.

Gilmore, P. C., & Gomory, R. E. (1963). A linear programming approach to the cutting stock problem—Part II. *Operations research, 11*, 863-888.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & operations research, 13*, 533-549. doi:10.1016/0305-0548(86)90048-1

Gray, A., Bush, S., & Whiteley, S. (2004). Secondary transport of the critically ill and injured adult. *Emergency Medicine Journal, 21*, 281-285. doi:10.1136/emj.2003.005975

Grazia Speranza, M. (1996). A heuristic algorithm for a portfolio optimization model applied to the Milan stock market. *Computers and Operations Research, 23*, 433-441. doi:10.1016/0305-0548(95)00030-5

Guerriero, F., Pezzella, F., Pisacane, O., & Trollini, L. (2014). Multi-objective optimization in dial-a-ride public transportation. *Transportation Research Procedia, 3*, 299-308.

Hamilton, S. M., & Breakey, P. (1995, 11). Fluid resuscitation of the trauma patient: How much is enough? *Canadian Journal of Surgery, 39*, 11-16. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3895117/

Harper, J. (n.d.). Going Underground: The U.S. Government’s Hunt for Enemy Tunnels. Retrieved from https://www.nationaldefensemagazine.org/articles/2018/1/2/going-underground-the-us-governments-hunt-for-enemy-tunnels

Haukoos, J. S., Byyny, R. L., Erickson, C., Paulson, S., Hopkins, E., Sasson, C., . . . Moore, E. E. (2011). Validation and Refinement of a Rule to Predict Emergency Intervention in Adult Trauma Patients. *Annals of Emergency Medicine, 58*, 164-171.

Hefny, A. F., Idris, K., Eid, H. O., & Abu-Zidan, F. M. (2013, 9). Factors affecting mortality of critical care trauma patients. *African Health Sciences, 13*, 731-735. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3824443/

Homeland Security, C., & Affairs, G. (2018). Border Security: Analysis of Vulnerabilities Identified by Frontline Agents.

Homeland Security, D. (2018). DHS Border Security Metrics Report.

Hulka, F., J Mullins, R., Mann, N. C., R Hedges, J., Rowland, D., H Worrall, W., . . . Trunkey, D. (1997, 4). Influence of a statewide trauma system on pediatric hospitalization and outcome. *The Journal of Trauma, 42*, 514-9.

Immigration Statistics, O. (2017). Efforts by DHS to Estimate Southwest Border Security between Ports of Entry.

Inspector General, O. (2017). Additional Actions Needed to Better Assess Fencing's Contributions to Operations and Provide Guidance for Identifying Capability Gaps. *Southwest Border Security, GAO-17-331*.

Inspector General, O. (2017). Border Patrol is Deploying Surveillance Technologies but Needs to Improve Data Quality and Assess Effectiveness. *Southwest Border Security, GAO-18-119*.

Inspector General, O. (2017). CBP's Border Security Efforts - An Analysis of Southwest Border Security Between the Ports of Entry. *Southwest Border Security, OIG-17-39*.

Jacobson, S. H., McLay, L. A., Kobza, J. E., & Bowman, J. M. (2005). Modeling and analyzing multiple station baggage screening security system performance. *Naval Research Logistics (NRL), 52*, 30-45. doi:10.1002/nav.20052

Jacobson, S. H., McLay, L. A., Virta, J. L., & Kobza, J. E. (2005). Integer programming models for deployment of airport baggage screening security devices. *Optimization and Engineering, 6*, 339-359. doi:10.1007/s11081-005-1743-5

Jaw, J.-J., Odoni, A. R., Psaraftis, H. N., & Wilson, N. H. (1986). A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research Part B: Methodological, 20*, 243-257.

Jusionyte, I. (2018). Called to “Ankle Alley”: tactical infrastructure, migrant injuries, and emergency medical services on the US--Mexico border. *American Anthropologist, 120*, 89-101.

Kachitvichyanukul, V., & others. (2009). A particle swarm optimization for vehicle routing problem with time windows. *Int. J. Oper. Res., 6*, 519-537.

Kean, T. (2011). *The 9/11 commission report: Final report of the national commission on terrorist attacks upon the United States.* Government Printing Office.

Kennedy, J. (1999). Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance. *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, *3*, pp. 1931-1938.

Kennedy, J., & Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. *1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation*, *5*, pp. 4104-4108.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science, 220*, 671-680. doi:10.1142/9789812799371_0035

Knight, V. A., Harper, P. R., & Smith, L. (2012). Ambulance allocation for maximal survival with heterogeneous outcome measures. *Omega, 40*, 918-926.

Konno, H., & Wijayanayake, A. (2001). Portfolio optimization problem under concave transaction costs and minimal transaction unit constraints. *Mathematical Programming, 89*, 233-250. doi:10.1007/pl00011397

Konno, H., & Yamamoto, R. (2005). Global optimization versus integer programming in portfolio optimization under nonconvex transaction costs. *Journal of Global Optimization, 32*, 207-219. doi:10.1007/s10898-004-2703-x

Konno, H., Akishino, K., & Yamamoto, R. (2005). Optimization of a long-short portfolio under nonconvex transaction cost. *Computational Optimization and Applications, 32*, 115-132. doi:10.1007/s10589-005-2056-5

Kos, M., Mikac, M., & Mikac, D. (2002). Topological planning of communication networks. *Journal of Information and Organizational Sciences, 26*, 57-68.

Lagos, F., Boland, N., & Savelsbergh, M. (2020). The Continuous-Time Inventory-Routing

Problem. *Transportation Science*.

Lam, S. S., Nguyen, F. N., Ng, Y. Y., Lee, V. P.-X., Wong, T. H., Fook-Chong, S. M., & Ong, M. E. (2015). Factors affecting the ambulance response times of trauma incidents in Singapore. *Accident Analysis & Prevention, 82*, 27-35.

Lau, H. C., Chan, T. M., Tsui, W. T., & Pang, W. K. (2009). Application of genetic algorithms to solve the multidepot vehicle routing problem. *IEEE transactions on automation science and engineering, 7*, 383-392.

Lavender, M. L. (2017). Decision Model for U.S,-Mexico Border Security Measures.

Le Thi, H. A., Moeini, M., & Dinh, T. P. (2009). DC programming approach for portfolio optimization under step increasing transaction costs. *Optimization, 58*, 267-289. doi:10.1080/02331930902741721

Lee, E. K. (2019). A Three Tier Security Capability Architecture. JPEG file.

Lee, E. K., & Mitchell, J. E. (2000). Computational experience of an interior-point SQP algorithm in a parallel branch-and-bound framework. In *High performance optimization* (pp. 329-347). Springer. doi:10.1007/978-1-4757-3216-0_13

Lee, E. K., Leonard, T. J., & Booker, J. (2019). Computational Framework for Multi-tier Risk Taxonomy Modeling and Assessment. *Working Paper*.

Lee, E. K., Leonard, T. J., Ozlu, & Wood. (2020). Strategic Decision Making in Trauma Systems.

Lenstra, J. K., & Kan, A. H. (1981). Complexity of vehicle routing and scheduling problems. *Networks, 11*, 221-227.

Leonard, T. J., & Lee, E. (2020). Building a Smarter Wall through Strategic Security Measure Allocation.

Leonard, T. J., & Lee, E. K. (2020). Computational Approaches in Solving Large-Scale Resource Allocation Problems.

Leonard, T. J., & Lee, E. K. (2020). Optimal Emergency Resource Allocation in Trauma Networks.

Leonard, T. J., Lee, E., & Booker, J. (2019). Risk Assessment and Security Measure Allocation for Optimal Aviation Security. *Preprint*.

Li, D., Sun, X., & Wang, J. (2006). Optimal lot solution to cardinality constrained mean--

variance formulation for portfolio selection. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics, 16*, 83-101. doi:10.1111/j.1467-9965.2006.00262.x

Li, H., & Lim, A. (2001). Pickup and delivery problem with time windows Instances.

Li, H., & Lim, A. (2003). A metaheuristic for the pickup and delivery problem with time windows. *International Journal on Artificial Intelligence Tools, 12*, 173-186.

Liang, J.-J., & Suganthan, P. N. (2005). Dynamic multi-swarm particle swarm optimizer. *Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005. SIS 2005.*, (pp. 124-129).

Lin, C.-C., & Liu, Y.-T. (2008). Genetic algorithms for portfolio selection problems with minimum transaction lots. *European Journal of Operational Research, 185*, 393-404. doi:10.1016/j.ejor.2006.12.024

Liu, R., Xie, X., Augusto, V., & Rodriguez, C. (2013). Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care. *European Journal of Operational Research, 230*, 475-486.

Long, D. (n.d.). CBP's Eyes in the Sky. Retrieved from https://www.cbp.gov/frontline/frontline-november-aerostats

Lu, Q., & Dessouky, M. M. (2006). A new insertion-based construction heuristic for solving the pickup and delivery problem with time windows. *European Journal of Operational Research, 175*, 672-687.

Lübbecke, M. E., & Desrosiers, J. (2005). Selected topics in column generation. *Operations research, 53*, 1007-1023.

Luck, S. (2018, 8 21). Privacy Impact Assessment Update for the Border Surveillance Systems (BSS). *DHS/CBP/PIA-022(a)*.

Luis, E., Dolinskaya, I. S., & Smilowitz, K. R. (2012). Disaster relief routing: Integrating research and practice. *Socio-economic planning sciences, 46*, 88-97.

Lwin, K., Qu, R., & Kendall, G. (2014). A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization. *Applied Soft Computing, 24*, 757-772. doi:10.1016/j.asoc.2014.08.026

Madsen, O. B., Ravn, H. F., & Rygaard, J. M. (1995). A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Annals of operations Research, 60*, 193-208.

Mansini, R., & Speranza, M. G. (1999). Heuristic algorithms for the portfolio selection problem with minimum transaction lots. *European Journal of Operational Research, 114*, 219-233. doi:10.1016/s0377-2217(98)00252-5

Mansini, R., & Speranza, M. G. (2005). An exact approach for portfolio selection with transaction costs and rounds. *IIE transactions, 37*, 919-929. doi:10.1080/07408170591007821

Mansini, R., Ogryczak, W., & Speranza, M. G. (2014). Twenty years of linear programming based portfolio optimization. *European Journal of Operational Research, 234*, 518-535. doi:10.1016/j.ejor.2013.08.035

Marcon, E., Chaabane, S., Sallez, Y., Bonte, T., & Trentesaux, D. (2017). A multi-agent system based on reactive decision rules for solving the caregiver routing problem in home health care. *Simulation Modelling Practice and Theory, 74*, 134-151.

Maringer, D., & Kellerer, H. (2003). Optimization of cardinality constrained portfolios with a hybrid local search algorithm. *Or Spectrum, 25*, 481-495. doi:10.1007/s00291-003-0139-1

Mark, G. S., & Kiersz, A. (2019). As the government shutdown over Trump's border wall rages, a journey along the entire 1,933-mile US-Mexico border shows the monumental task of securing it. Retrieved from https://www.businessinsider.com/us-mexico-border-wall-photos-maps-2018-5

Markowitz, H. (1952). Portfolio selection. *The journal of finance, 7*, 77-91. doi:10.1111/j.1540-6261.1952.tb01525.x

McLay, L. A. (2011). Risk-Based Resource Allocation Models for Aviation Security. In *Safety and Risk Modeling and Its Applications* (pp. 243-261). Springer. doi:10.1007/978-0-85729-470-8_9

McLay, L. A., Jacobson, S. H., & Kobza, J. E. (2006). A multilevel passenger screening problem for aviation security. *Naval Research Logistics (NRL), 53*, 183-197. doi:10.1002/nav.20131

McLay, L. A., Jacobson, S. H., & Kobza, J. E. (2007). Integer programming models and analysis for a multilevel passenger screening problem. *IIE Transactions, 39*, 73-81. doi:10.1080/07408170600729200

McLay, L. A., Jacobson, S. H., & Kobza, J. E. (2008). The tradeoff between technology and prescreening intelligence in checked baggage screening for aviation security. *Journal of Transportation Security, 1*, 107-126. doi:10.1007/s12198-007-0010-8

McLay, L. A., Jacobson, S. H., & Nikolaev, A. G. (2009). A sequential stochastic passenger screening problem for aviation security. *IIE Transactions, 41*, 575-591. doi:10.1080/07408170802510416

McLay, L. A., Lee, A. J., & Jacobson, S. H. (2010). Risk-based policies for airport security checkpoint screening. *Transportation science, 44*, 333-349. doi:10.1287/trsc.1090.0308

Min, H., Current, J., & Schilling, D. (1992). The multiple depot vehicle routing problem with backhauling. *Journal of Business Logistics, 13*, 259.

Mizushima, Y., Nakao, S., Idoguchi, K., & Matsuoka, T. (n.d.). Fluid Resuscitation of Trauma Patients. *The American Journal of Emergency Medicine, 35*, 842-845. doi:10.1016/j.ajem.2017.01.038

Nable, J. V., Lawner, B. J., & Brady, W. J. (2016). 2016: emergency medical services annotated literature in review. *The American journal of emergency medicine, 34*, 2193-2199.

Nagy, G., & Salhi, S. (2005). Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. *European journal of operational research, 162*, 126-141.

Nanry, W. P., & Barnes, J. W. (2000). Solving the pickup and delivery problem with time windows using reactive tabu search. *Transportation Research Part B: Methodological, 34*, 107-121.

Nathens, A., Brunet, F., & Maier, R. (2004, 5). Development of trauma systems and effect on outcomes after injury. *The Lancet, 363*, 1794-801. Retrieved from http://search.proquest.com/docview/198979654/

National Center for Injury Prevention and Control. (2018). *CDC Injury Center Research Priorities.* Atlanta, GA: Center for Disease Control.

Nebro, A. J., Durillo, J. J., Luna, F., Dorronsoro, B., & Alba, E. (2009). MOCell: A cellular genetic algorithm for multiobjective optimization. *International Journal of Intelligent Systems, 24*, 726-746. doi:10.1002/int.20358

Nie, X., Batta, R., Drury, C. G., & Lin, L. (2009). Passenger grouping with risk levels in an airport security system. *European Journal of Operational Research, 194*, 574-584.

Olson, D. L., Wu, D. D., & others. (2010). *Enterprise risk management models.* Springer. doi:10.1007/978-3-662-53785-5

Ombuki-Berman, B., & Hanshar, F. T. (2009). Using genetic algorithms for multi-depot vehicle routing. In *Bio-inspired algorithms for the vehicle routing problem* (pp. 77-99). Springer.

Ostadrahimi, L., Mariño, M. A., & Afshar, A. (2012). Multi-reservoir operation rules: multi-swarm PSO-based optimization approach. *Water resources management, 26*, 407-427.

Palmer, C. (2007). Major Trauma and the Injury Severity Score - Where Should We Set the Bar? *Annual Proceedings / Association for the Advancement of Automotive Medicine, 51*, 13-29. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3217501/

Pankratz, G. (2005). A grouping genetic algorithm for the pickup and delivery problem with time windows. *Or Spectrum, 27*, 21-41.

Parragh, S. N. (2011). Introducing heterogeneous users and vehicles into models and algorithms for the dial-a-ride problem. *Transportation Research Part C: Emerging Technologies, 19*, 912-930.

Parragh, S. N., Cordeau, J.-F., Doerner, K. F., & Hartl, R. F. (2012). Models and algorithms for the heterogeneous dial-a-ride problem with driver-related constraints. *OR spectrum, 34*, 593-633.

Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2008). A survey on pickup and delivery models part ii: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft, 58*, 81-117.

Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2010). Variable neighborhood search for the dial-a-ride problem. *Computers & Operations Research, 37*, 1129-1138.

Patrol, U. S. (2018). Privacy Impact Assessment Update for the Border Surveillance Systems (BSS). *DHS/CBP, PIA-022(A)*.

Pedraza-Martinez, A. J., & Van Wassenhove, L. N. (2012). Transportation and vehicle fleet management in humanitarian logistics: challenges for future research. *EURO Journal on Transportation and Logistics, 1*, 185-196.

Petersen, B. (2011). Shortest paths and vehicle routing.

Petersen, B., & Jepsen, M. K. (2009). Partial path column generation for the vehicle routing problem with time windows. *Proceedings of the 4th International Network Optimization Conference (INOC2009), pages paper TB4-3.*

Pirlot, M. (1996). General local search methods. *European journal of operational research, 92*, 493-511.

Pluhacek, M., Senkerik, R., Viktorin, A., & Zelinka, I. (2018). Single swarm and simple multi-swarm PSO comparison. *Proceedings of The 9th EUROSIM Congress on Modelling and Simulation, EUROSIM 2016, The 57th SIMS Conference on Simulation and Modelling SIMS 2016*, (pp. 556-560).

Poole, R. W., & Passantino, G. M. (2003). *Risk-based Airport Security Policy.* Tech. rep., Reason Public Policy Institute Los Angeles, CA.

Prin, M., & Li, G. (2016, 8). Complications and in-hospital mortality in trauma patients treated in intensive care units in the United States, 2013. *Injury Epidemiology, 3*, 18. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4974260/

Prins, C., Lacomme, P., & Prodhon, C. (2014). Order-first split-second methods for vehicle routing problems: A review. *Transportation Research Part C: Emerging Technologies, 40*, 179-200.

Qu, B. Y., Zhou, Q., Xiao, J. M., Liang, J. J., & Suganthan, P. N. (2017). Large-Scale Portfolio Optimization Using Multiobjective Evolutionary Algorithms and Preselection Methods. *Mathematical Problems in Engineering, 2017*. doi:10.1155/2017/4197914

Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N., & Barabási, A. L. (2002, 8). Hierarchical organization of modularity in metabolic networks. *Science (New York, N.Y.), 297*. doi:10.1126/science.1073374

Reifel, C. S. (2006). *Quantitative risk analysis for homeland security resource allocation.* Ph.D. dissertation, Monterey California. Naval Postgraduate School.

Renaud, J., Boctor, F. F., & Ouenniche, J. (2000). A heuristic for the pickup and delivery traveling salesman problem. *Computers & Operations Research, 27*, 905-916.

Rittenberger, J. C., & Callaway, C. W. (2009). Transport of Patients After Out-of-Hospital Cardiac Arrest: Closest Facility or Most Appropriate Facility? *Annals of Emergency Medicine, 54*, 256-257. doi:https://doi.org/10.1016/j.annemergmed.2009.01.009

Rogers, F. B., Rittenhouse, K. J., & Gross, B. W. (2015). The golden hour in trauma: Dogma or medical folklore? *Injury, 46*, 525-527. doi:https://doi.org/10.1016/j.injury.2014.08.043

Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation science, 40*, 455-472.

S&T Impact: Borders & Ports of Entry. (n.d.). Washington D.C.: Department of Homeland Security. Retrieved from https://www.dhs.gov/science-and-technology/st-impact-borders-ports-entry

Sadler, S. (2016, 4). TSA Secure Flight Program. TSA. Retrieved from https://www.tsa.gov/news/testimony/2014/09/18/tsa-secure-flight-program

Savelsbergh, M. W., & Sol, M. (1995). The general pickup and delivery problem. *Transportation science, 29*, 17-29.

Sawik, B. (2011). *Multi-objective portfolio optimization by mixed integer programming.* Ph.D. dissertation, AGH University of Science and Technology.

Sawik, B. (2013, 11). Survey of multi-objective portfolio optimization by linear and mixed integer programming. In *Applications of management science* (pp. 55-79). Emerald Group Publishing Limited. doi:10.1108/s0276-8976(2013)0000016007

Schmid, V. (2012). Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming. *European journal of operational research, 219*, 611-621.

Schmid, V., & Doerner, K. F. (2010). Ambulance location and relocation problems with time-dependent travel times. *European journal of operational research, 207*, 1293-1303.

Sewell, E. C., Attagara, J., Kobza, J. E., & Jacobson, S. H. (2012). Allocating explosive screening devices for aviation security. *Journal of Transportation Security, 5*, 141-155. doi:10.1007/s12198-012-0087-6

Sewell, E. C., Lee, A. J., & Jacobson, S. H. (2013). Optimal allocation of aviation security screening devices. *Journal of Transportation Security, 6*, 103-116. doi:10.1007/s12198-013-0106-2

Shadkam, E., Delavari, R., Memariani, F., & Poursaleh, M. (2015, 8). Portfolio selection by the means of cuckoo optimization algorithm. *Preprint, 5*. doi:10.5121/ijcsa.2015.5304

Shi, Y., & Eberhart, R. C. (1998). Parameter selection in particle swarm optimization. *International conference on evolutionary programming*, (pp. 591-600).

Singer, A. J., Quinn, A., Dasgupta, N., & Thode, H. C. (2017). Management and Outcomes of Bleeding Events in Patients in the Emergency Department Taking Warfarin or a Non–Vitamin K Antagonist Oral Anticoagulant. *The Journal of Emergency Medicine, 52*, 1 - 7.e1. doi:https://doi.org/10.1016/j.jemermed.2016.09.028

Solomon, M. M. (1987). “Vehicle Routing Problem with Time Windows Benchmark Problems.

Solomon, S., Thulasiraman, P., & Thulasiram, R. (2011). Collaborative multi-swarm PSO for task matching using graphics processing units. *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, (pp. 1563-1570).

Sombuntham, P., & Kachitvichayanukul, V. (2010). A particle swarm optimization algorithm for multi-depot vehicle routing problem with pickup and delivery requests. *World Congress on Engineering 2012. July 4-6, 2012. London, UK., 2182*, pp. 1998-2003.

Soriano, A., Gansterer, M., & Hartl, R. F. (2018). The two-region multi-depot pickup and delivery problem. *OR Spectrum, 40*, 1077-1108.

Stewart, M. G., & Mueller, J. (2014). Cost-benefit analysis of airport security: Are airports too safe? *Journal of Air Transport Management, 35*, 19-28. doi:10.1016/j.jairtraman.2013.11.003

Stewart, M. G., & Mueller, J. (2017). Risk and economic assessment of expedited passenger screening and TSA PreCheck. *Journal of transportation security, 10*, 1-22.

Surekha, P., & Sumathi, S. (2011). Solution to multi-depot vehicle routing problem using genetic algorithms. *World Applied Programming, 1*, 118-131.

Sweeney, D. J., & Murphy, R. A. (1979). A method of decomposition for integer programs. *Operations research, 27*, 1128-1141.

Syam, S. S. (1998). A dual ascent method for the portfolio selection problem with multiple constraints and linked proposals. *European Journal of Operational Research, 108*, 196-207. doi:10.1016/s0377-2217(97)00048-9

Talarico, L., Meisel, F., & Sörensen, K. (2015). Ambulance routing for disaster response with patient groups. *Computers & Operations Research, 56*, 120-133.

Tikani, H., & Setak, M. (2019). Ambulance routing in disaster response scenario considering different types of ambulances and semi soft time windows. *Journal of Industrial and Systems Engineering, 12*, 95-128.

Tlili, T., Harzi, M., & Krichen, S. (2017). Swarm-based approach for solving the ambulance routing problem. *Procedia Computer Science, 112*, 350-357.

Toro-DíAz, H., Mayorga, M. E., Chanta, S., & Mclay, L. A. (2013). Joint location and dispatching decisions for emergency medical services. *Computers & Industrial Engineering, 64*, 917-928.

Transportation, U. S. (n.d.). Retrieved from Federal Aviation Administration: faa.gov

Trauma, C. (2008). Regional Trauma Systems: Optimal Elements, Integration and Assessment.

TSA. (2014). *Transportation Security Administration Enterprise Risk Management: Emergency Risk Management Policy Manual.* TSA.

U.S. Border Patrol Southwest Border Apprehensions by Sector FY2018. (n.d.). Retrieved from https://www.cbp.gov/newsroom/stats/usbp-sw-border-apprehensions

Van den Bergh, F., & Engelbrecht, A. P. (2006). A study of particle swarm optimization particle trajectories. *Information sciences, 176*, 937-971.

Veenema, K. R., & Rodewald, L. E. (1995). Stabilization of Rural Multiple-Trauma Patients at Level III Emergency Departments Before Transfer to a Level I Regional Trauma Center. *Annals of Emergency Medicine, 25*, 175-181. Retrieved from http://www.sciencedirect.com/science/article/pii/S0196064495703209

Virta, J. L., Jacobson, S. H., & Kobza, J. E. (2003). Analyzing the cost of screening selectee and non-selectee baggage. *Risk Analysis: An International Journal, 23*, 897-908.

Virta, J., Jacobson, S. H., & Kobza, J. E. (2002). Outgoing selectee rates at hub airports. *Reliability Engineering & System Safety, 76*, 155-165.

Vles, W. J., Steyerberg, E. W., Meeuwis, J. D., & Leenen, L. P. (2004). Pre-hospital trauma care: a proposal for more efficient evaluation. *Injury, 35*, 725-733.

Wahde, M. (2008). *Biologically inspired optimization methods: an introduction.* WIT press.

Wang, X., Xu, C., & Shang, H. (2008). Multi-depot vehicle routing problem with time windows and multi-type vehicle number limits and its genetic algorithm. *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, (pp. 1-5).

What Is Network Topology? (2019, August 19). Retrieved from dnsstuff.com:

https://www.dnsstuff.com/what-is-network-topology

Wilhelm, W. E. (2001). A technical review of column generation in integer programming. *Optimization and Engineering, 2*, 159-200.

Wolsey, L. A., & Nemhauser, G. L. (1999). *Integer and combinatorial optimization* (Vol. 55). John Wiley & Sons.

Wright, P. D., Liberatore, M. J., & Nydick, R. L. (2006). A survey of operations research models and applications in homeland security. *Interfaces, 36*, 514-529. doi:10.1287/inte.1060.0253

Wu, C.-H., & Hwang, K. P. (2009). Using a Discrete-event Simulation to Balance Ambulance Availability and Demand in Static Deployment Systems. *Academic Emergency Medicine, 16*, 1359-1366. doi:10.1111/j.1553-2712.2009.00583.x

Xie, T., Wei, Y., Pan, L., Wang, T., & Chen, H. (2016). Modeling and Simulating for Emergency Medical Service System Optimizing Based on Discrete Event System Theory. *Open Journal of Social Sciences*, 133-136. doi:10.4236/jss.2016.47022

Xue, H.-G., Xu, C.-X., & Feng, Z.-X. (2006). Mean–variance portfolio optimal problem under concave transaction cost. *Applied Mathematics and Computation, 174*, 1-12. doi:10.1016/j.amc.2005.05.005

Yaakob, S. B., & Watada, J. (2010). A hybrid particle swarm optimization approach to mixed integer quadratic programming for portfolio selection problems. *International Journal of Simulation: Systems, Science and Technology, 11*, 68-74.

Yip, A. M., & Horvath, S. (2007). Gene network interconnectedness and the generalized topological overlap measure. *BMC bioinformatics, 8*, 22.

Zhang, Z., Liu, M., & Lim, A. (2015). A memetic algorithm for the patient transportation problem. *Omega, 54*, 60-71.

Zhao, S.-Z., Liang, J. J., Suganthan, P. N., & Tasgetiren, M. F. (2008). Dynamic multi-swarm particle swarm optimizer with local search for large scale global optimization. *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, (pp. 3845-3852).

Zuhal, L. R., & others. (2010). Resolving multi objective stock portfolio optimization problem using genetic algorithm. *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on*, *2*, pp. 40-44. doi:10.1109/iccae.2010.5451372