2014

Metaheuristic Approaches For Estimating In-Kind Food Donations Availability And Scheduling Food Bank Vehicles

III Luther Brock
North Carolina Agricultural and Technical State University

Metaheuristic Approaches for Estimating In-Kind Food Donations Availability and Scheduling Food Bank Vehicles

Luther G. Brock III

North Carolina A&T State University

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Department: Industrial & Systems Engineering

Major Professor: Dr. Lauren B. Davis

Greensboro, North Carolina

2014

The Graduate School
North Carolina Agricultural and Technical State University
This is to certify that the Doctoral Dissertation of

Luther G. Brock III

Has met the dissertation requirements of
North Carolina Agricultural and Technical State University

Greensboro, North Carolina
2014

Approved by:

Lauren B. Davis
Major Professor

Xiuli Qu
Committee Member

Abdollah Homaifar
Committee Member

Paul Stanfield
Committee Member

Steven X. Jiang
Committee Member

Tonya Smith-Jackson
Department Chair

Sanjiv Sarin
Dean, The Graduate School

Biographical Sketch

Luther G. Brock III was born to Luther and Rhonda Brock in Winston-Salem, NC. He earned his Bachelors and Masters of Science degrees in Industrial & Systems Engineering at North Carolina A&T State University. Luther's industry experience has allowed him to work for organizations such as the United States Postal Service, United Technologies, and Army Research Laboratories. His research interests include supply chain management, operations research, systems engineering, and data mining. He is particularly interested in using his skills to address challenges that affect the health and well-being of others. Luther is happily married to his wife LaMesha. They currently reside in Greensboro, NC.

Dedication

Lord, thank you for allowing me to complete this work. I acknowledge that every good idea, notion, and contribution presented in this dissertation is from You. It is only fitting that I dedicate this completed dissertation back to You. I pray that the strategies presented in this research lead to positive changes that improve the living conditions of individuals experiencing food insecurity. I also pray that this dissertation inspires all who read its contents to persevere through life's difficult processes, keeping the end in mind.

Acknowledgements

I would like to thank the outstanding staff of the Industrial and Systems Engineering Department for their encouragement and assistance while completing my degrees, as well as express my appreciation to committee members for providing helpful feedback for various aspects of my research. I would especially like to thank my dissertation advisor, Dr. Lauren B. Davis, for her patience, guidance, and support throughout my doctoral studies. I would also like to express my gratitude to Drs. Eui Park, Bala Ram, and Paul Stanfield for their direction, support, and encouragement during my undergraduate studies. Without your intervention at a crucial point in my life, I may very well not have discovered the passion that I have for this discipline. Although it is too many of you to name, thank you to the many graduate students and faculty who have provided sound advice, shared lessons learned, given constructive criticism, and shared words of encouragement. I am also appreciative to both the Title III HGBI Program and the Alfred P. Sloan Foundation for providing funding while completing my doctoral studies.

Most importantly, I want to thank my family members who have shown amazing confidence and support while I pursued my doctorate. To my wife, LaMesha, words do not express my gratitude to you for how you have been by my side throughout this process. Thanks for being my biggest supporter, providing words of encouragement at just the right time, and being patient with me in all aspects of life. I love you, and as I've said before, "you can't get rid of me even if you want to." I am blessed to have two sets of amazing parents. To my birth parents, Luther Jr. and Rhonda Brock, thank you for your support throughout my life. You guys taught me to never stop dreaming, regardless of who criticizes me. I'm following your advice. To my parents by marriage, Dexter "Dex" and Dorothy "Matrix" Stephens, thanks for your support as well. Dex, thanks again for proofreading my dissertation. To every person that I may parent or mentor, my dream is for you to know that you can accomplish your God-given goals. It was my honor to be an example by completing this process. To other family members and friends who have in some way inspired me but were not listed, I hope to thank each of you personally.

Table of Contents

List of Figures xiii

List of Tables xv

List of Acronyms xvii

Abstract 1

CHAPTER 1 Introduction 2

1.1. Problem Motivation 2

1.2. Operational Challenges and Opportunities for the CFSFSC 5

1.2.1. Constrained and Uncertain Supply 5

1.2.2. Routing Schedule Complexity 6

1.3. Contribution of Work 7

1.4. Key Findings 9

1.5. Organization of Dissertation 10

CHAPTER 2 Estimating Available Supermarket Commodities for Food Bank Collection in the Absence of Information 11

2.1. Introduction 11

2.2. Related Literature 14

2.3. Methodology 16

2.3.1. Assumptions 17

2.3.2. Variables and Representation 19

2.3.3. Traditional Forecasting Methods 24

2.4. Multi-Layer Perceptron Neural Network Models 26

2.4.1. Model Structure 26

2.4.2. Quasi-Greedy Algorithm for MLP-NN Model Selection 28

2.5. Computational Study 31

2.5.1. Data 31

2.5.2. Data Preprocessing 32

2.5.3. Experiments 33

2.6. Results 38

2.6.1. Performance of Forecasting Methods 39

2.6.2. Impacts on Transportation Costs 45

2.7. Managerial Impacts 47

2.8. Conclusion 48

CHAPTER 3 Formulations for the Periodic Vehicle Routing Problem with Backhauls 51

3.1. Introduction 51

3.2. Literature Review 54

3.2.1. PVRP 55

3.2.2. VRPB 61

3.2.3. Unaddressed Research Area 62

3.3. Model Formulation 63

3.4. Formulations for Variants for the PVRPB ... 67

3.4.1. Time Windows ... 68

3.4.2. Heterogeneous Fleet ... 68

3.5. Tour Limitation Constraints ... 69

3.6. Experimentation ... 70

3.6.1. Case Study: Charitable Food Distribution and Collection Challenges of Food Banks ... 71

3.6.2. Experimental Design ... 74

3.6.3. Equipment ... 76

3.7. Results ... 76

3.7.1. Experiment #1: Computational Complexity of PVRPB Extensions ... 76

3.7.2. Experiment #2: Tour Limitation Constraints ... 82

3.8. Managerial Insights ... 86

3.9. Conclusions ... 87

CHAPTER 4 A Hybrid Genetic Algorithm for Solving the Periodic Vehicle Routing Problem with Backhauls ... 89

4.1. Introduction ... 89

4.2. Literature Review ... 90

4.2.1. Metaheuristic Search Procedures for the PVRP ... 92

4.2.2. Hybrid Genetic Algorithms for Routing Problems ... 95

4.2.3. Limitations of Existing Metaheuristics for Research Problem 97

4.3. MHR Characteristics 98

4.3.1. Solution Space 100

4.3.2. Representation 102

4.3.3. Evaluation of Individuals 105

4.3.4. Initial Population 107

4.3.5. Mating Selection Reproduction Day Targeting 108

4.3.6. Crossover Operator 110

4.3.7. Mutation Operator 112

4.3.8. Learning Operator 113

4.3.9. Replacement Operator 117

4.4. Calibration of MHR Settings 118

4.5. Experimentation 119

4.6. Results 123

4.6.1. MHR Precision for Problem Variants 123

4.6.2. MHR Accuracy for Problem Variants 126

4.7. Managerial Insights 132

4.8. Conclusions 133

CHAPTER 5 Recommendations and Future Extensions 135

References 137

Appendix A: Pseudocodes for Heuristics 152

A.1. Pseudocode for the Quasi-Greedy Algorithm 152

A.2. Pseudocode for Initial Route Construction 153

A.3. Pseudocode for Crossover Operator 157

A.4. Pseudocode for Modified Local Search Operators 161

A.4.1. Modified Two Opt* Inter-Route Operator 161

A.4.2. Modified RELOCATE Inter-Route Move Operator 162

A.4.3. Modified CROSS Inter-Route Operator 165

A.4.4. Modified Two Opt Intra-Route Operator 166

A.4.5. Modified Or Operator 167

Appendix B: Food Bank Network Characteristics 171

B.1. Linehaul Customers 171

B.2. Backhaul Customers 172

B.3. Vehicle Characteristics by Fleet Type 175

B.4. Travel Time Between Locations 176

B.5. Travel Distance Between Locations 198

Appendix C: Schedules Associated with Best Solutions for Different Routing Problems 220

C.1. PVRPB Solutions Obtained Using Basic Model Formulation 220

C.2. PVRPB Solutions Obtained by Adding Tour Limitation Constraints to Model Formulation......226

C.3. PVRPB Solutions Obtained Using MULTI-HGA-ROUTE......236

C.4. PVRPBTW Solutions Obtained Using Basic Model Formulation......245

C.5. PVRPBTW Solutions Obtained Adding Tour Limitation Constraints to Model Formulation......251

C.6. PVRPBTW Solutions Obtained Using MULTI-HGA-ROUTE......259

C.7. HPVRPB Solutions Obtained Using Basic Model Formulation......270

C.8. HPVRPB Solutions Obtained by Adding Tour Limitation Constraints to Model Formulation......278

C.9. HPVRPB Solutions Obtained Using MULTI-HGA-ROUTE......289

List of Figures

Figure 1.1. Food Insecurity Percentages, 1998 - 2012 2

Figure 1.2. Diagram of Material and Information Flows in the CFSFSC 4

Figure 2.1. Description of Planning Horizon for Food Collections 19

Figure 2.2. A Visual Representation of an L-Layered MLP-NN 27

Figure 2.3. Diagram of the Quasi-Greedy Algorithm Model Selection Process 30

Figure 2.4. Data Pre-processing, Model Training, and Experimentation Processes Implemented for the Computational Study 33

Figure 2.5. Food Collections and Percentage Change in Estimated Transportation Costs for Greenville Branch 46

Figure 2.6. Food Collections and Percentage Change in Estimated Transportation Costs for Wilmington Branch 47

Figure 3.1. Transportation Costs Obtained for PVRPB, PVRPBTW, and HPVRPB in Test Scenarios 81

Figure 3.2. Optimality Gap Percentages for the BMF when Solving Test Scenarios for the PVRPB, PVRPBTW, and HPVRPB 82

Figure 4.1. Process Flow for MHR 98

Figure 4.2. The Representations for the First and Second Individual 105

Figure 4.3. Example of Two Parents and Offspring Created by the Crossover Operator 112

Figure 4.4. Visual Representation of Modified Shift Operator 113

Figure 4.5. Visual Representation for Two-Opt*, CROSS, and RELOCATE Inter-Route Operators 115

Figure 4.6. Visual representation of Two-Opt and Or-Opt Intra-Route Operators 116

Figure 4.7. Impact of different *nkeep* Values on Diversity 119

List of Tables

Table 2.1 Representation for System Variables........ 23

Table 2.2 Training Parameters for the Quasi-Greedy Algorithm........ 36

Table 2.3 Summary Statistics for Data Set after Preprocessing (By Food Type)........ 39

Table 2.4 Model Characteristics for Selected MLP-NN Models........ 40

Table 2.5 Model Approximation Performance (By Food Type)........ 42

Table 3.1 Problem Decomposition Used to Solve the PVRP........ 57

Table 3.2 Characteristics for each Test Scenario........ 72

Table 3.3 Vehicle Characteristics........ 73

Table 3.4 Results Obtained when Attempting to Solve the BMF of the PVRPB........ 76

Table 3.5 Results Obtained when Attempting to Solve the BMF of the PVRPBTW.... 78

Table 3.6 Results Obtained when Attempting to Solve the BMF of the HPVRPB........ 80

Table 3.7 Solutions Obtained when solving the BMF and TLC Formulations of the PVRPB........ 83

Table 3.8 Solutions Obtained when solving the BMF and TLC Formulations of the PVRPBTW........ 84

Table 3.9 Solutions Obtained when solving the BMF and TLC Formulations of the HPVRPB........ 85

Table 4.1 Metaheuristics Methods to Solving the PVRP........ 93

Table 4.2 Notation for Solution Space........ 101

Table 4.3 Example of Tabu List Selection........ 110

Table 4.4 Precision Measures for Feasible MHR Solutions to the PVRPB........ 123

Table 4.5 Transportation Cost Measures for Feasible MHR Solutions to the PVRPBTW 124

Table 4.6 Transportation Cost Measures for Feasible MHR Solutions to the HPVRPB 125

Table 4.7 Comparison of MHR to BMF and TLC when Solving PVRPB 126

Table 4.8 Comparison of MHR to BMF and TLC when Solving PVRPBTW 129

Table 4.9 Comparison of MHR to BMF and TLC when Solving HPVRPB 131

List of Acronyms

BMF: Basic model formulation for a PVRPB variant as a mixed integer linear program

CFSFSC: Charity-Focused Secondary Food Supply Chain

CVRP: Capacitated Vehicle Routing Problem

FA: Feeding America ©

GA: Genetic algorithm

HGA: Hybrid Genetic Algorithm

HPVRPB: Heterogeneous Fleet Periodic Vehicle Routing Problem with Backhauls

MHR: MULTI-HGA-ROUTE (an HGA proposed to solve PVRPB variants)

MILP: Mixed Integer Linear Program

MLP-NN: Multi-Layer Perceptron Neural Network

MLR: Multiple Linear Regression

PVRP: Periodic Vehicle Routing Problem

PVRPB: Periodic Vehicle Routing Problem with Backhauls

PVRPBTW: Periodic Vehicle Routing Problem with Backhauls and Time Window

PVRPBTW: Periodic Vehicle Routing Problem with Time Windows

SS: Scatter Search

TLC: Basic model formulation for a PVRPB variant as a mixed integer linear program when including tour limitation constraints

TSP: Traveling Salesman Problem

VRPB: Vehicle Routing Problem with Backhauls

VRPBTW: Vehicle Routing Problem with Backhauls and Time Windows

VRPTW: Vehicle Routing Problem with Time Windows

Abstract

Food banks provide services that allow households facing food insecurity to receive nutritious food items. Food banks, however, experience operational challenges as a result of constrained and uncertain supply and complex routing challenges. The goal of this research is to explore opportunities to enhance food bank operations through metaheuristic forecasting and scheduling practices. Knowledge discovery methods and supervised machine learning are used to forecast food availability at supermarkets. In particular, a quasi-greedy algorithm which selects multi-layer perceptron models to represent food availability is introduced. In addition, a new classification of the vehicle routing problem is proposed to manage the distribution and collection of food items. In particular, variants of the periodic vehicle routing problem backhauls are introduced. In addition to discussing model formulations for the routing problems, a hybrid genetic algorithm is introduced which finds good solutions for larger problem instances in a reasonable computation time.

CHAPTER 1

Introduction

1.1. Problem Motivation

Food insecurity is defined as the inability of individuals to obtain consistent access of adequate food (Maxwell and Frankenberger 1992). This condition affects a significant proportion of the U.S. population. In two reports, the US Department of Agriculture estimate that food insecurity affects at least 10 percent of all U.S. households (Nord et al. 2009, Coleman-Jensen et al. 2011). These reports make the connection between food insecurity and economic conditions, showing that food insecurity increased to 14.6% in 2008 with the onset of the recession. Figure 1.1 gives a visual representation of food insecurity since 1998.

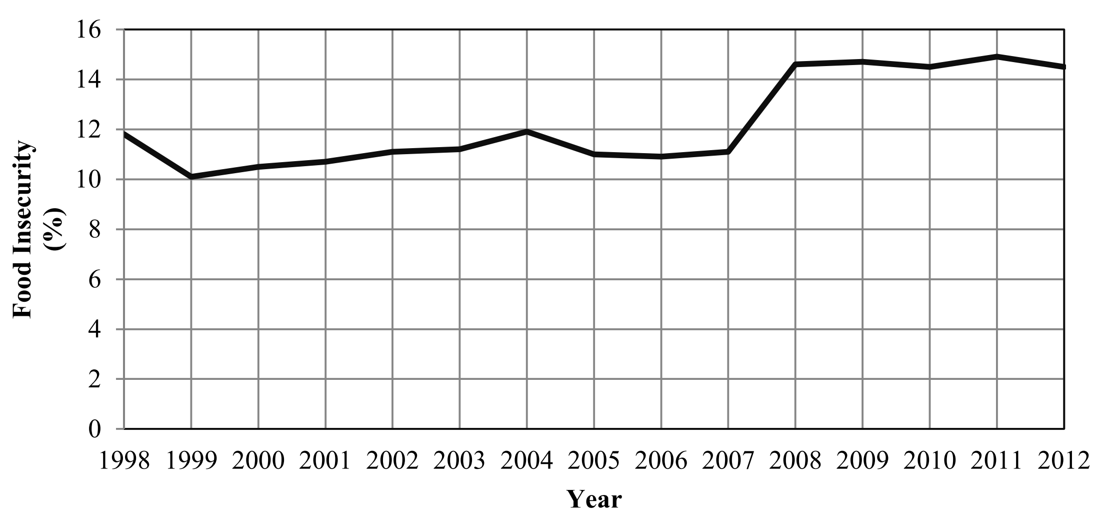


Figure 1.1. Food Insecurity Percentages, 1998 - 2012 (Coleman-Jensen et al. 2011)

The effects of food insecurity are counteracted through non-profit organizations. Some of them, including food banks, soup kitchens, food pantries, and shelters provide in-kind food to those that might otherwise not receive it. This research is motivated by the desire to improve the efficiency of non-profit food collection and distribution in relation to food bank operations.

The existence of food banks has been vital to sustainability in the United States. Food banks serve as central warehouses where donations can be inventoried and distributed to any number of charitable agencies. Many government-supported programs, (e.g. including The Emergency Food Assistance Program) depend on food banks to transport surplus commodities to emergency food programs in local communities (Cabili et al. 2013). Food banks are also involved with disaster relief, providing millions of pounds of food to individuals in affected areas (FA 2011b). Over 200 food banks operating in the United States are affiliates of Feeding America®, a network of food banks, corporations, and community-concerned groups whose objective is to end food insecurity in the United States.

The flow of in-kind foods, resources, funds, and information among government, local donors, and collaborating food banks is captured in Figure 1.2. In this research, this is referred to as the Charity-Focused Secondary Food Supply Chain (CFSFSC). Product flows are represented by forward arcs. Reverse flows represent the sharing of information and dollars that would promote improved coordination.

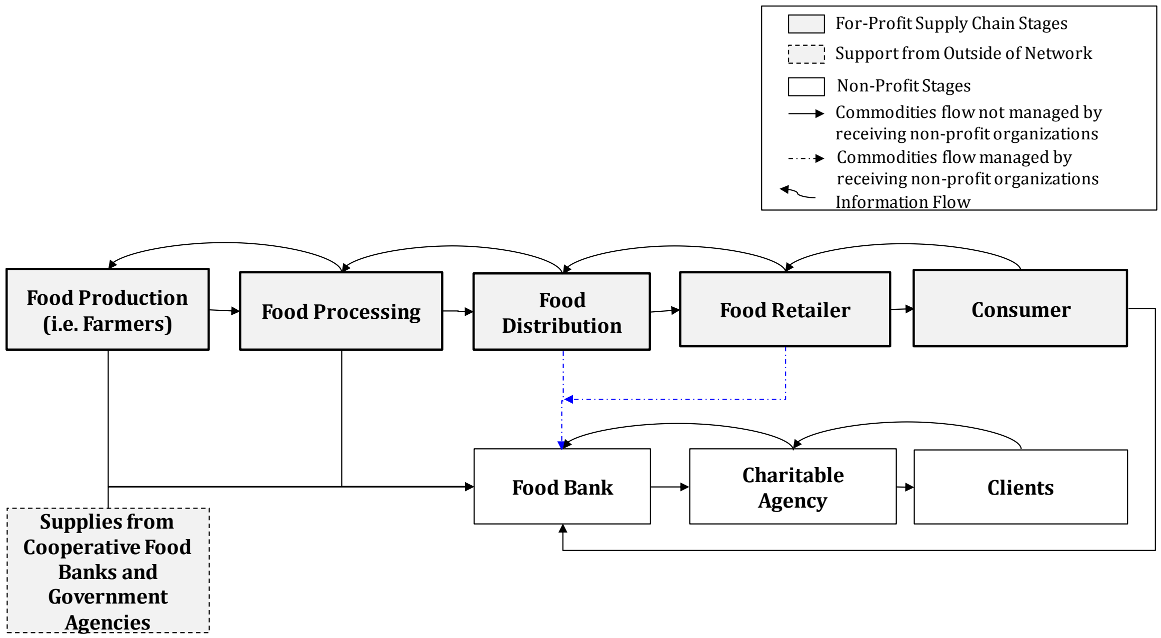


Figure 1.2. Diagram of Material and Information Flows in the CFSFSC (Adapted from FA 2011b)

Food banks represent one of three stages in the CFSFSC whose primary concern is to alleviate hunger at the local level. The others are charitable agencies and clients receiving food assistance. Food banks perform most of the warehousing and logistical operations. Commodities received by food banks are distributed to charitable agencies who distribute the commodities to their clients.

Local sources of supply for the CFSFSC can be generated at each stage of the for-profit food supply chain. Food producers (i.e. farmers) provide food donations that are leftover from harvested commercial crops. Food processors and food distributors provide items which are usable yet inappropriate to be sold in local markets. Examples of these commodities include dented canned goods and products with minor damage to labeling. Food retailers provide items which are approaching their manufacturer-recommended "sell-by" dates (see Teron and Tarasuk (1999), Tarasuk and Eakin (2005), FA 2011a). While the majority of high-volume donations are received from food processors and retailers, consumers also serve as a source of supply through local food drives.

1.2. Operational Challenges and Opportunities for the CFSFSC

1.2.1. Constrained and Uncertain Supply

Charitable agencies rely heavily upon food bank assistance. According to a report by Mabli et al. (2010), food banks account for 76% of the food received by pantries, 50% of the food received by soup kitchens, and 41% of food received by shelters. However, the ability to provide uninterrupted services to the charities they serve is challenging given the limited availability of supply. Many food banks across America have experienced supply shortages. The Food Bank of Northwest North Carolina (FBNWNC), for example, reported three supply depletions between 2009 – 2012, two of which occurred in 2011 (Campbell 2011; Garms 2012). Some of its metropolitan areas, including Winston-Salem and Greensboro/High Point were among the most food insecure in the country in the calendar years where shortages were reported (FRAC 2011, 2012 and 2013).

Food items provided by government agencies are not expected to fully satisfy local hunger needs, and the primary responsibility of cooperating food banks is their own local communities. Food banks must generate sufficient supply from their local communities to promote on-going service. This is extremely difficult because the amount of food provided by local donors is both uncertain and in many cases, not obtained unless an on-site collection is performed. Furthermore, the use of forecasting methods in food banks operations is limited. Supply forecasting could help to mitigate the effects of supply shortages by quantifying the amounts of different in-kind food types available through different donor locations. If not addressed, supply shortages are inevitable.

1.2.2. Routing Schedule Complexity

A second challenge for the CFSFSC is presented by the complex routing challenges associated with providing equitable food assistance to individuals through a food bank service area without compromising client safety and neglecting inventory replenishment needs. Since food banks are responsible for much of the food delivery to distant charitable agencies, there is a tremendous workload assigned to them. For example, the Good Shepard food bank of Maine states cites "a lack of transportation as a common and significant barrier for its food pantries," (GSFB 2013). This, coupled with the need to replenish supply through local donation sources, presents a very complicated vehicle scheduling problem with many alternatives that should be assessed. The challenge is further complicated by the need to keep commodities that have not been evaluated (i.e. collected foods) at the warehouse (i.e. food bank) separated from items that have been verified as safe (i.e. distributed foods). Lastly, limitations with respect to operation time, vehicle capabilities and fleet size, and concrete food delivery and collection requirements force routing decisions to occur over a multi-day planning horizon.

1.3. Contribution of Work

The goal of this research is to explore opportunities to improve food bank operations through forecasting and scheduling practices. The first contribution of this research is the use of knowledge discovery methods and supervised machine learning to characterize the nature of food availability at supermarkets. In particular, multiple layer perceptron neural network models are developed to predict the amount of in-kind food types available for collection at different supermarkets. Historic food collection records, community-specific employment data, and financial wellness indicators are used to characterize food donations. A quasi-greedy heuristic is used to select a multi-layer perceptron model to represent the relationship between the donation-specific information and collection amounts.

The scope of this research is limited to inventoried food items. Forecasts are limited to food retailers providing large quantities of in-kind donations (i.e. supermarkets).

The specific research questions addressed are as follows.

- Are the MLP-NN estimates for in-kind food collections more accurate than other approximation methods that are more commonly utilized when information sharing is and is not permitted?
- Do the estimates translate to an observed improvement in transportation costs?

Few researchers address in-kind donations forecasting for food banks. Among the areas discuss in literature include the assessment of food quality and food bank workforce in Canada (Teron and Tarasuk 1999, Tarasuk and Eakin 2005), community assessments that validate the need for food banks (Vinopal and Cooper 2011, Winter 2009, Mosley and Tiehen 2004), and the scheduling of vehicles and food allocation to charitable agencies and/or collection (Bartholdi et al. 1983, Lien et al. 2008, Gunes et al. 2010). Researchers have also identified methods of approximating the average food supply received from donors over a specified time interval (Davis et al. 2013a, 2013b) and schedule the collection of the average supply to meet aggregate demand (Phillips et al. 2011). Nonetheless, the shortage of related contributions, suggests that there is little literature which introduces methods for estimating the amount of food that is available at grocery stores at the time of an on-site food collection.

The second contribution introduces a new model formulation that addresses the vehicle routing problems experienced by food banks when collecting and distributing inventoried food. In particular a periodic vehicle routing problem with backhauls (PVRPB) is presented. The essential features of this problem consists of constructing routes over a fixed time horizon that encompass food collections, food deliveries, constraints on vehicle capacity, food spoilage, and operator workday, as well as collection and delivery frequency. Model formulations and computational complexity are discussed. In addition, a genetic algorithm-based approached is presented to find good solutions for large problems. Such a problem has not been discussed in the literature. Gunes et al. (2010) discuss different formulations for the one-commodity generalized pickup and delivery problem that can be used to manage the collection of prepared food items and delivery to charitable agencies. This model is appropriate for food items that have very short shelf lives which prevent them from being stored as inventory. Solak et al. (2012) introduce the stop-and-drop problem to assign agencies to food delivery sites. This approach, however, does not take food collection into account, nor does it address routing food collections or deliveries over a multi-day planning horizon.

1.4. Key Findings

This research makes a case for incorporating artificial intelligence approaches into two important aspects of food bank operations: forecasting and scheduling. The research suggests that multi-layer perceptron neural network (MLP-NN) models are more effective than traditional forecasting methods at accounting for supply uncertainty in the CFSFSC. This claim is supported by demonstrating how their improved forecasts result in better estimates for actual transportation costs. Furthermore, a quasi-greedy algorithm is introduced which executes a MLP-NN models selection process which incorporates the impacts of the model structure, its initial weights, and data partitioning strategies. The research also suggests that food banks can implement the PVRPB as a universal routing strategy for the collection and delivery of inventoried food items. When fewer customers are in the network, these schedules can be determined using a model formulation. Schedules which for larger networks, however, should use advanced search procedures. This research presents a genetic algorithm-based metaheuristic capable of finding a feasible set of routes in a very reasonable computation time.

The proposed metaheuristic makes a contribution to vehicle routing literature in that it can be applied to a variety of routing problem generalizations.

1.5. Organization of Dissertation

This research is organized as follows. Chapter 2 presents a combined data mining/supervised machine learning approach to estimating the amounts of different in-kind food types that are available for collection. The approach involves the selection of multi-layer perceptron neural networks to estimate the amounts of different in-kind food types available for collection at supermarket branches. The usefulness of the neural network models is compared to traditional forecasting methods both in terms of predictive error and impacts on food collection costs. Chapter 3 presents model formulations for variants of the periodic vehicle routing problem with backhauls. Formulations for the problem variants are introduced. In addition, the computational complexity and resulting transportation costs for different test instances are observed. Chapter 4 presents a hybrid genetic algorithm that finds good solutions for each routing problem in a reasonable computation time. After providing a detailed description of the metaheuristic, a set of experiments are performed to validate its ability to provide consistent, cost effective solutions for each routing problem. Chapter 5 summarizes the findings of this research and identifies opportunities for future research extensions.

CHAPTER 2

Estimating Available Supermarket Commodities for Food Bank Collection in the Absence of Information

2.1. Introduction

Food banks collect, store, and distribute food donated by local businesses (i.e. food producers/manufacturers, food distributors, and supermarkets) and community-serving organizations. These commodities are processed, stored, and eventually dispatched to charitable agencies. The charitable agencies, in turn, distribute the items that they receive to individuals and families experiencing food insecurity. Their warehousing capabilities, interest in providing unbiased service to the agencies, and cooperative approach to counteracting hunger make food banks an attractive non-profit agency to high-volume donors. Supermarkets are one of the high-volume donation sources for food banks. Commodities that are generated from supermarkets include food items that are usable yet for various reasons, unsellable in local markets. Examples of these edible food items include dented canned goods, bruised fruit, and non-perishables approaching manufacturer-recommended sell by dates. The donation of these items is both good-hearted and practical because their disposal would otherwise be managed by the supermarket branch and/or franchise. Food banks welcome these items as tight funding both limits the amount of food that can be purchased in local markets and the amount of money that can be allocated for daily operations.

One of the obstacles to scheduling food bank operations is the uncertainty in available supply. Food banks must make collections at supermarkets with no indication as to whether desired food items are available, and if so, how much. Unlike typical for-profit supply chains, suppliers (i.e. supermarkets) have a different objective than their downstream recipients (i.e. food banks). While interested in aiding the food insecure, these donors are in business to make a profit. This profit is realized by selling food items rather than donating them. Furthermore, supermarkets typically elect not to share information regarding product availability because it is either difficult to forecast (Pechenizkiy 2008) or kept confidential. Without having knowledge of what items are available for collection at different stores, the degree to which food banks can make cost-effective transportation schedules is limited. The problem is further complicated by this being a decision that is made each day.

The goal of this research is to identify an approximation method that is useful when estimating the amounts of different in-kind food types available for collection at a supermarket branch. This extends the work of Brock and Davis (2012) to address instances where there is no information shared between supermarket branches and the regional food banks. As specified in the preceding investigation, quantifying food availability is complicated because 1) collections can occur at different points in time, 2) the amount of surplus food available changes over time, 3) food is perishable and must be collected and distributed quickly to avoid spoilage. Given the dynamics associated with collection frequency and food availability, approximation methods must have the ability to generalize what is received in a specific collection event. Therefore, knowledge discovery and supervised machine learning approaches are used to predict food availability. In particular, multi-layer perceptron neural network (MLP-NN) models are proposed to determine the amounts of food available for a specific collection event. The models incorporate information related to the observable characteristics of a collection event, the financial wellness of communities served, and past operational decisions made by the food bank. The MLP-NN models are compared to more traditional approximation methods. Specifically, we consider multiple linear regressions, the average collection amount received by a regional food bank, and the average collection amount received by a specific warehouse maintained by the regional food bank. The results of a computational study show that MLP-NN models are more effective than traditional forecasting methods at accounting for supply uncertainty. The results also show that the improved forecasts also result in better estimates for transportation costs.

The remainder of this chapter is organized as follows. Section 2.2 provides a review of literature related to the forecasting problem. Sections 2.3 and 2.4 summarize the approximation methods considered in this research. Section 2.5 provides a more detailed description of supermarket and regional food bank practices that affect food availability. Section 2.6 presents a case study implemented using data from the Food Bank of Central and Eastern North Carolina Food Bank (FBCENC). Section 2.7 compares the four approximation methods based on both prediction error and the impacts of using each method to schedule food collections. Section 2.8 summarizes the key findings of this research and identifies opportunities for future research extensions.

2.2. Related Literature

The idea of using data mining in the context of operations management is not new. In fact it has been utilized in a number of applications including engineering design (Feng and Wang et al. 2003; Feng et al. 2006) and production and maintenance scheduling (Luxhoj et al. 1997; Sha and Liu 2005). While there is little work published in the context of in-kind donations forecasting, there is considerable work published that is relevant to demand forecasting. A partial review of recent work addressing demand forecasting is provided. For a comprehensive review of demand forecasting methods, the reader is referred to Zhang et al. (1998).

Meulstee and Pechenizkiy (2008) address the challenges associated with wholesale food suppliers estimating demand for food items sold. The researchers incorporate ensemble learning approaches to predict product sales. The problem is motivated by the need for food suppliers to improve forecasting ability. This problem posed in their investigation is similar to this research in that the forecasted outcome (i.e. product sales) is perceived as affected by some unknown context (i.e. consumer preference, habits, interests, etc.) that could not be effectively monitored through previous forecasting methods. The ensemble model incorporates sales for different product types, past weather conditions, and school holidays.

Gutierreza et al. (2008) evaluates forecasting methods that are appropriate for products with intermittent demand. Such forecasts are used when there may be long periods when items are not demanded followed by periods when demand is elevated. In their investigation, they compare MLP-NN models, simple exponential smoothing and the smoothing approximations of Croston (1972) and Syntetos and Boylan (2005).

Shahrabi et al. (2009) evaluates different forecasts when determining long-term demand for car components. In their investigation, the moving average, exponential smoothing, exponential smoothing with trend, support vector regressions, and MLP-NN models are compared.

Forecasting methods are also utilized to assess demand for limited resources including water (Adamowski 2008, Firat et al. 2009, Pulido-Calvo et al. 2007), energy consumption in buildings (Ekici and Aksoy 2009), energy consumption by communities as a whole (Geem and Roper2009, Murat and Ceylan (2006); Wang and Liang 2009).

The problem presented in this study is one where food banks must be able to estimate how much of each in-kind food type is available for collection on specific collection days. Each of the aforementioned forecasting methods is limited for this problem. Most time series methods focus on the cumulative collection amount received. When attempting to schedule vehicles for food collection it is necessary to focus on the anticipated amount of food received through an isolated collection event. Davis et al. (2013) does give some estimate for what can be received through a collection; however, estimates are limited to the average monthly collection amount. This is an important distinction because food banks may elect to perform zero, one, two or multiple collections at either a given supermarket branch, supermarket franchise, or at supermarkets in general for an unspecified period of time. The contribution of Brock and Davis (2012) is limited to situations where the types of food that supermarket branches have on-hand is known. This is unlikely for many systems, as there is no coordination mechanism that allows food banks to know with certainty which items will be available for collection upon arrival.

2.3. Methodology

This study compares four forecasting methods that may be utilized to estimate the amount of different in-kind food types available for collection. These forecasting methods are 1) the average amount received from a supermarket branch (SM Average), 2) the average amount received by a specific warehouse from a supermarket branch (SMWH Average), 3) the predicted amount received by a regional food bank as determined using multiple linear regressions (MLR), and 4) the predicted amount received by a regional food bank as determined by the selected MLP-NN models. The first two forecasting methods are based on observed collection amounts at a specific location. MLR and MLP-NN models are based on a number of event-related characteristics. Event-related characteristics considered are observable system characteristics, the financial wellness of supermarket customers, and past operational decisions made by the food bank. All four approximation methods are evaluated to determine which is most appropriate to estimate the amount of food available in the next planning period.

While making predictions, the MLR and MLP-NN models are evaluated to determine whether they identify the system as stationary or non-stationary. System stationarity implies that in separate observations, there is no change in the overall impacts of system inputs. In relation to this problem, system stationarity would be implied by approximations methods presented in this research producing similar predictions in different planning periods. When systems are non-stationary, planning methods must compensate for system variability.

Past transactions involving supermarket branches and food bank-managed warehouses are based on historic records maintained by a regional food bank. A regional food bank consists of one or more warehouses which perform food collections. A variety of pre-processing techniques are applied to the data, resulting in a final set of observations that are used to define a relationship between a set of system characteristics and the resulting collection amounts. After pre-processing data, different forecasting methods are used to estimate the amount of each of the different food types targeted by the regional food bank.

2.3.1. Assumptions

The system characteristics outlining food collection events are as follows. Parties directly involved in the system include supermarket locations and regional food bank affiliates. Supermarket branches generate usable, yet unsellable in-kind food items on a nightly basis as a result of a number of customer purchasing decisions. It is assumed that customers can be generalized to the county in which they reside. The amount that is generated nightly is not shared with food bank affiliates nor is the amount accumulated. Food items generated remain on-hand until a collection occurs or the end of the calendar week. At the end of the calendar week, any on-hand food items are disposed. A regional food bank receives unusable food items through on-site collections performed by personnel at one of its affiliate warehouses. Each warehouse maintains its own refrigerated vehicles. Records of all collections are maintained by the regional food bank. Each record indicates the date of food collection, the supermarket branch where food was collected, what food was received, and which warehouse collected the food. It is assumed that warehouses process all collected commodities on the day they are collected, classifying each food as a specific food type. Planning activities for food collections occur prior to the start of the operating week. The planning horizon, including the availability of food in a collection week is provided by Figure 2.1.

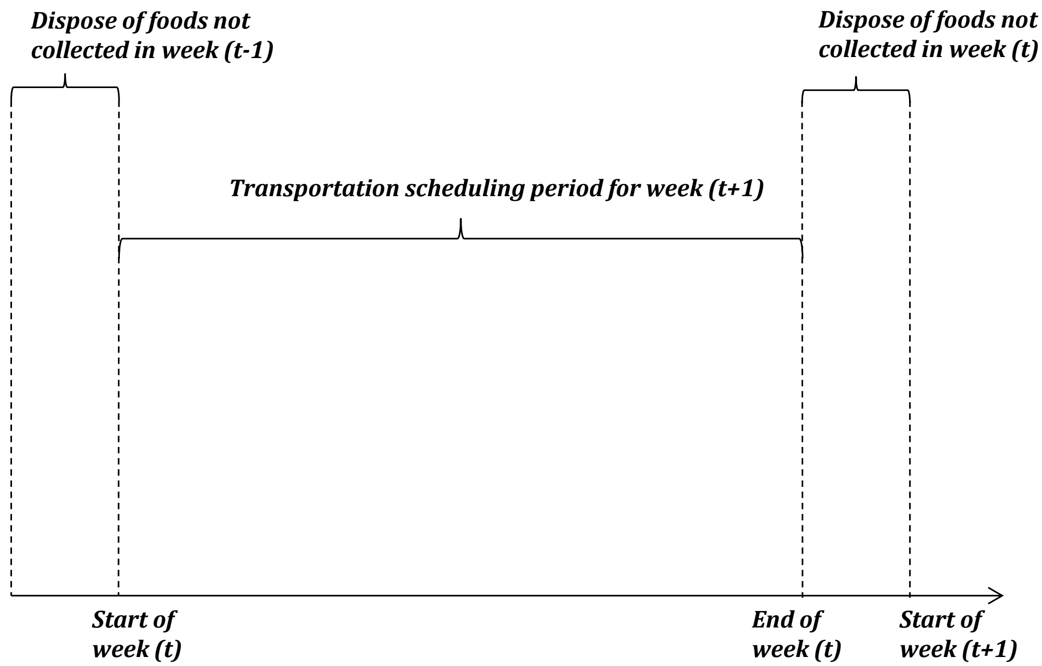


Figure 2.1. Description of Planning Horizon for Food Collections

2.3.2. Variables and Representation

The dependent variable (DV) in this system is the amount of a particular in-kind food type that is available for receipt at a particular supermarket. This amount is represented by the term *Output_Amount.* Independent variables (IV) considered in this study are classified as indicating 1) the observable characteristics of a collection event, 2) the financial wellness of targeted supermarket customers, or 3) past operational decisions.

2.3.2.1. *Observable System Characteristics*. Observable characteristics are those that are most apparent when a collection occurs. They express who performed the food collection, where it was collected, and when it occurred. *Week_of_Year* and *Weekday* indicate the week of the calendar year and day of the calendar week when a collection occurs. The inclusion of these variables allows the model to express impacts that may be the result of seasonality and collecting on specific days of the calendar week. The parties involved in past food collections are given by *WH* and *SM*. *WH* identifies the food bank warehouse that collected donated food items. This variable is important when food banks operate out of more than one storage location, as it allows the model to differentiate between receipts collected at each. *SM* identifies the supermarket branch from which collections are received.

2.3.2.2. *Financial Wellness*. *Real_GDP* provides a measure for the gross domestic product for a state after accounting for price inflations and deflation. The gross domestic product expresses the overall standard of living in a specific state. This value is calculated yearly. *Unemploy_Rate* gives the rate of unemployment in the current calendar month. This measure is evaluated at the county level monthly by the North Carolina Bureau of Labor Statistics. *Cons_Confid* gives a measure for consumer confidence in the economy. This measure is calculated based on statewide consumer spending over the previous calendar month.

2.3.2.3. *Past Operational Decisions*. Numeric measures which identify the recency, frequency, and monetary value of customers are important factors in customer relationship management. These are typically called RFM variables. They have been useful in many contexts including direct marketing (see e.g., McCarty and Hastak (2007)) and business management (see e.g., Li et al. (2008)). Given the success of using these factors in their respective contexts, historic donation records are manipulated to express the recency, frequency, and value of supermarket donors measured from prior donations. This measurement is recorded with respect to both overall receipts by the regional food bank and those collected by the receiving warehouse.

2.3.2.3.1. *Recency.* The recency of past collections is measured through the number of days which have elapsed since the last collection. *CW_Rec_SM* measures the number of elapsed days since a collection at the supermarket has been attempted by any warehouse affiliated with the regional food bank. *CW_REC_SMWH* measures the number of elapsed days since a collection at the supermarket had been attempted by the warehouse performing the current collection. Recency is defined by integers in the range [0, 8], where a value of 8 represents that there has not been a prior collection in the calendar week.

2.3.2.3.2. *Frequency.* *CW_Freq_SM* and *CW_Freq_SMWH* indicate the number of collections at the contributing supermarket that have occurred in the current calendar week which a) have been received by any warehouse of the regional food bank and b) have been received by a specific warehouse, respectively. *Ttl_Freq_SM* and *Ttl_Freq_SMWH* indicate the total number of recorded collections at the contributing supermarket a) received by any warehouse of the regional food bank and b) received by a specific warehouse, respectively.

2.3.2.3.3. *Value.* *Week_Amt_SM* and *Week_Amt_SMWH* indicate the amount of a specific food type that has been generated from the supermarket previously in the current calendar week which a) has been received by any warehouse affiliated with the regional food bank and b) has been received by a specific warehouse, respectively. *Ttl_Amt_SM* and *Ttl_Amt_SMWH* express the total amount of a specific food type that has been generated from the supermarket branch which a) has been received by any warehouse affiliated with the regional food bank and b) has been received by a specific warehouse, respectively. Both *Ttl_Amt_SM* and *Ttl_Amt_SMWH* include all donations received to date prior to the current receipt.

A complete listing of the representation for system variables is given in Table 2.1. The *one-of-m* encoding is used for independent variables that are either nominal variables or have smaller ratio values. This encoding scheme uses a set of *m* dummy variables to represent each possible value. Variables such as *Week_of_Year,* W*eekday,* and *WH* are represented through this scheme. Independent variables that have continuous values are scaled to the range of [0, 1]. The decision was made to scale continuous variables to the range [0, 1] instead of [-1, 1] so that both discrete and continuous variables are maintained within the same interval. The decision was made to simply scale continuous variables with larger ranges (i.e. *Week_Amt_SM*, *Ttl_Amt_SMWH*, etc.) to the range [0, 1] for two reasons. First, a considerable amount of time is required to identify the most effective bins for each category represented by a *one-of-m* encoding scheme. Second, system characteristics are prone to change in subsequent scheduling periods, making the bins used to make one set of projections inappropriate for the next planning period. Crone et al. (2006) suggest that scaling continuous variables to this range promotes improved neural network performance as opposed to no pre-processing. While *Output_Amount* was originally scaled to this range as well, the decision was made not to scale the dependent variable.

Table 2.1

Representation for System Variables

Classification	**Sub-Classification**	**Variable**	**Role**	**Representation**
Observable Characteristics	Collection Date	*Week_of_Year*	IV	[1-of-*m*]
		Weekday	IV	[1-of-*m*]
	Transaction Parties	*WH*	IV	[1-of-*m*]
		SM	IV	[1-of-*m*]
Financial Wellness		*Real_GDP*	IV	[0,1] scaling
		Unemloy_Rate	IV	[0,1] scaling
		Cnsmr_Confid.	IV	[0,1] scaling

Table 2.1 (cont'd.)

Past Operational Decisions	Recency	*CW_Rec_SM*	IV	[1-of-m]
		CW_Rec_SMWH	IV	[1-of-m]
	Frequency	*CW_Freq_SM*	IV	[0,1] scaling
		CW_Freq_SMWH	IV	[0,1] scaling
		Ttl_Freq_SM	IV	[0,1] scaling
		Ttl_Freq_SMWH	IV	[0,1] scaling
	Value	*Week_Amt_SM*	IV	[0,1] scaling
		Week_Amt_SMWH	IV	[0,1] scaling
		Ttl_Amt_SM	IV	[0,1] scaling
		Ttl_Amt_SMWH	IV	[0,1] scaling
Collection Amount		*Output_Amount*	DV	[0,∞)

2.3.3. Traditional Forecasting Methods

The model formulations for each of the traditional forecasting methods are now discussed. The SM Average and SMWH Average are based solely on outcomes from historical collections. The MLR incorporates all of the independent and dependent variables presented in Table 2.1. It should be noted that each of the traditional methods represent an approach that might be implemented in practice. The SM Average and SMWH Average are naïve estimates assumed to remain constant throughout the planning period.

2.3.3.1. *Model #1: SM Average*. The average amount of food generated from a specific supermarket is defined in equation (2.1). Given a specific food type l, $y_o^{(n,l)}$ gives the amount of food available from supermarket branch

n at collection event o. The total number of collection events occurring from branch n is given by S_n.

$$\hat{Y}_{n,l} = \frac{\sum_{o=1}^{S_n} y_o^{(n,l)}}{S_n} \tag{2.1}$$

2.3.3.2. *Model #2: SMWH Average*. The average amount of food generated from a specific supermarket when collected by a food bank warehouse is defined in equation (2.2). Given a specific food type l, $y_o^{(n,l,a)}$ gives the amount of food available from supermarket branch n collected by warehouse a at collection event o. The total number of collection events occurring from branch n is given by $S_{n,a}$.

$$\hat{Y}_{n,l,a} = \frac{\sum_{o=1}^{S_{n,a}} y_o^{(n,l,a)}}{S_{n,a}} \tag{2.2}$$

2.3.3.3. *Model #3: MLR Model*. MLR creates an approximation for the amount of food collected based on the linear combination represented by a set of weighted system inputs. MLR is approximated by equation (2.3) where x_i represents system input i, m reflects the number of system inputs, and w_i represents the coefficients assigned to each input. Model bias is represented by w_o.

$$\hat{Y} = w_0 + \sum_{i=1}^{m} w_i x_i \tag{2.3}$$

The coefficient values are identified by solving for the least squared error among all observations. In the case of this modeling approach, this is approximated using a linear function. MLR models are sufficient for representing systems when multiple co-linearity among system inputs is negligible and the represented system is not subject to noise.

2.4. Multi-Layer Perceptron Neural Network Models

2.4.1. Model Structure

MLP-NNs are a type of feed-forward artificial neural network represented through at least three layers of neurons. The neurons of a MLP-NN are represented through an acyclic directed graph where each neuron is represented by a node. The first layer of nodes represents identified system inputs and the last layer represents observed system outcomes. The nodes in layers located between the first and last layer represent intermediate signal transmissions. These nodes and layers are hereafter referred to as hidden nodes and hidden layers, respectively. The arcs that connect nodes of a preceding layer to those in the next layer reflect the portion of the cumulative signal emitted from a firing neuron that is directed toward a receiving neuron. Arcs also represent a bias that is attributed to each layer on neurons. A visual representation of a MLP-NN with L layers is given in Figure 2.2. An in-depth explanation of the figure follows.

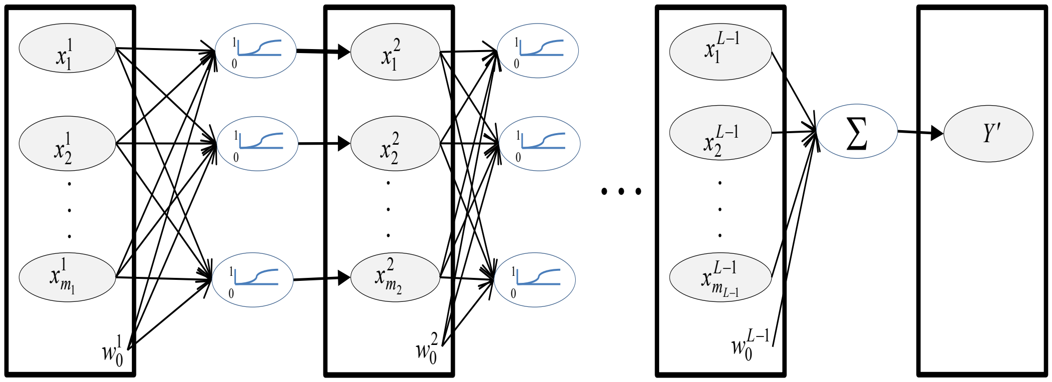


Figure 2.2. A Visual Representation of an L-Layered MLP-NN

All nodes in the network have a value assigned to them which represents a signal. First layer nodes reflect the signal resulting from observed system characteristics. The signal from these nodes is transmitted to nodes in subsequent layers until it reaches the output layer. The signal observed by a transmitting neuron is x_i^k where i indicates a specific node in layer k when that node is not located in the output layer. Nodes that are located in the output layer are represented by $\hat{Y}$. Each weighted arc is represented by w_{ij}^k where i indicates the transmitting node of layer k and j indicates the receiving node of the next layer. Layer bias is represented by w_{0j}^k. The cumulative signal ψ_j^{k+1} is defined in equation (2.4).

$$\psi_j^{k+1} = w_{0j}^k + \sum_{i=1}^{m_{k-1}} w_{ij}^k x_i^k \tag{2.4}$$

The response of receiving neurons is represented using a monotonically increasing transfer function. According to Demuth and Beale (1993), approximation models typically utilize an S-shaped transfer function to represent the cumulative signal received at hidden layers and linear transform activation functions to represent the cumulative signal received in the last layer. This study uses the logistic sigmoid function to represent this response. The resulting neuron and its firing strength, denoted by x_j^{k+1} are given by (2.5).

$$x_j^{k+1} = \frac{1}{1 + e^{-\psi_j^{k+1}}} \tag{2.5}$$

Since the last layer is represented using a linear transform function, the outcomes portrayed by nodes is defined according to equation (2.6).

$$\hat{Y} = w_{0j}^{L-1} + \sum_{i=1}^{M_{L-1}} w_{ij}^{L-1} x_i^{L-1} \tag{2.6}$$

MLP-NN models, therefore, have non-linearity distributed throughout their structures.

2.4.2. Quasi-Greedy Algorithm for MLP-NN Model Selection

MLP-NN models are impacted by a number of random events including a) the initial arc weights and b) model structure, and c) observations assigned to different data partitions. These models are also a function of the selected back-propagation method and neural network configuration. Model selection methods should, therefore, take these factors into consideration. Many researchers have attempted to identify rules that can be used to approximate the number of hidden neurons that should be used for different model constructions. However, no overall approximation for this range is widely accepted. Hence, a trial-and error approach is typically used (Zhang et al. 1998).

This model selection process can be very time consuming, as it is dependent upon the number of user-specified conditions for (a) early-stopping criteria, (b) maximum training time, and (c) maximum number of training epochs. The model selection process is further complicated by the need for researchers to introduce test conditions and manually record their results. This is especially true when performing repeated runs for each configuration to account for the impact of random arc initialization conditions and data partitioning. In the case of this study, multiple food types are explored, thus further complicating the organization of the model selection process.

These challenges are overcome by creating a quasi-greedy algorithm to automate the testing of different training parameters, comparing constructed MLP-NN, and selecting a non-dominated MLP-NN. Parameters introduced by the algorithm include 1) different data partitions for training/test sets, 2) the number of hidden layers, and 3) the number of hidden nodes in each layer. It also permits users to select the number of times that each combination of parameters should be repeated. The algorithm is classified as quasi-greedy because although it has the ability to terminate when a more complex model structure results in an inferior solution, the algorithm can also explore even more complex model structures. The algorithm is initialized by evaluating models constructed using the set of input and output. At initialization the MLP-NN structure evaluated has one hidden layer containing one hidden node. Model complexity is introduced by adding neurons to the hidden layer. Complexity is also increased by adding one additional hidden layer at a time. This process is repeated iteratively until a non-dominated MLP-NN with the lowest generalization error is found. A visual representation of this algorithm is illustrated in Figure 2.3.

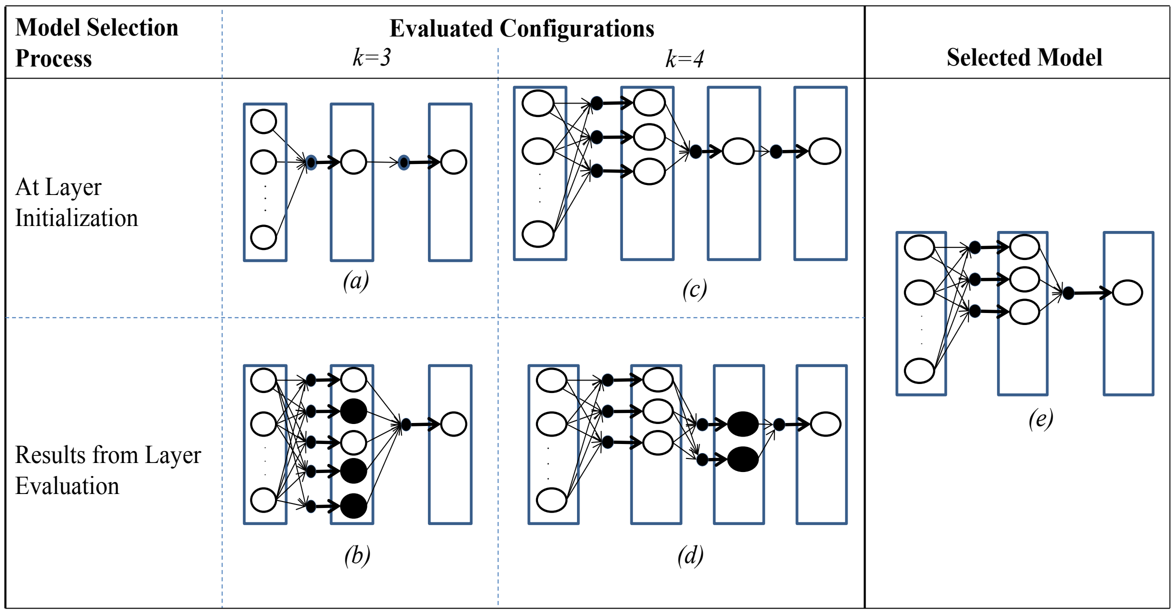


Figure 2.3. Diagram of the Quasi-Greedy Algorithm Model Selection Process

In this example, user-specified conditions state that model structures with no more than two additional neurons in the actively-evaluated hidden layer can be explored without a change in the non-dominated MLP-NN. Models constructed where the additional neuron did not produce improvements in predictive error are shaded black. The search process is initialized with $k = 3$. The algorithm initially explores different models that are structured with one hidden layer containing one hidden neuron (2.3a). After evaluating structures with five neurons, the non-dominated MLP-NN for structures with k = 3 layers is determined to be a model with three hidden nodes in its hidden layer (2.3b). The algorithm continues by evaluating model structures with k = 4 layers. The first model evaluated has the same structure as the non-dominated MLP-NN with an additional hidden layer containing one neuron (2.3c). Model configurations with as many as two neurons are explored before terminating (2.3d), both of which are inferior to the non-dominated MLP-NN where k = 3. Since the non-dominated MLP-NN for k = 3 is superior to the non-dominated MLP-NN where k = 4, the quasi-greedy heuristic selects the non-dominated MLP-NN model where k = 3 to make forecasts for system outcomes (2.3e). The pseudocode provided in Appendix A.1 summarizes the algorithm.

2.5. Computational Study

2.5.1. Data

A computational study is conducted using data provided by the Food bank of Central and Eastern North Carolina (FBCENC). FBCENC operates out of 6 warehouses. The historical data used in this study reflect food collections at supermarket branches between July 1, 2006 and April 30, 2011. The data consists of a total of 17,555 records. Four of the 6 food bank warehouses are represented in these collection records. The collection amounts range from 1 to 19,585 pounds. Each collection record indicates the date of collection, supermarket branch, commodity type, receiving warehouse, and zip code of the contributing supermarket. There are a total of 70 zip codes, 124 supermarkets, and 15 different in-kind food types represented in the data.

2.5.2. Data Preprocessing

The pre-processing, model training, and experimentation processes implemented in this study are described in Figure 2.4. Transactions reflecting receipts from supermarket branches are extracted and matched with financial indicators based on the date of collection and county in which the donating supermarket resides. Records pertaining to food types received in less than 5% of all collections are then removed. Additionally, the collection amounts are limited to being no more than the lower 95^{th} percentile of what had been reported for the food type in historic records. Next, past operational decisions are interpreted based on the dates when one of the remaining in-kind food types is received. Having incorporated all variables and addressed all outliers, data are partitioned to represent the known history and a future planning horizon.

Observations from the known history are used to predict SM Averages and SMWH Averages. These models are validated by comparing the predicted collection amounts over the known history to the actual collections in the future planning horizon. Prior to making predictions using MLR or MLP-NN models, variables are converted as described in Section 2.4.1. MLR models are created without partitioning the known history. MLP-NN models are selected using the quasi-greedy algorithm.

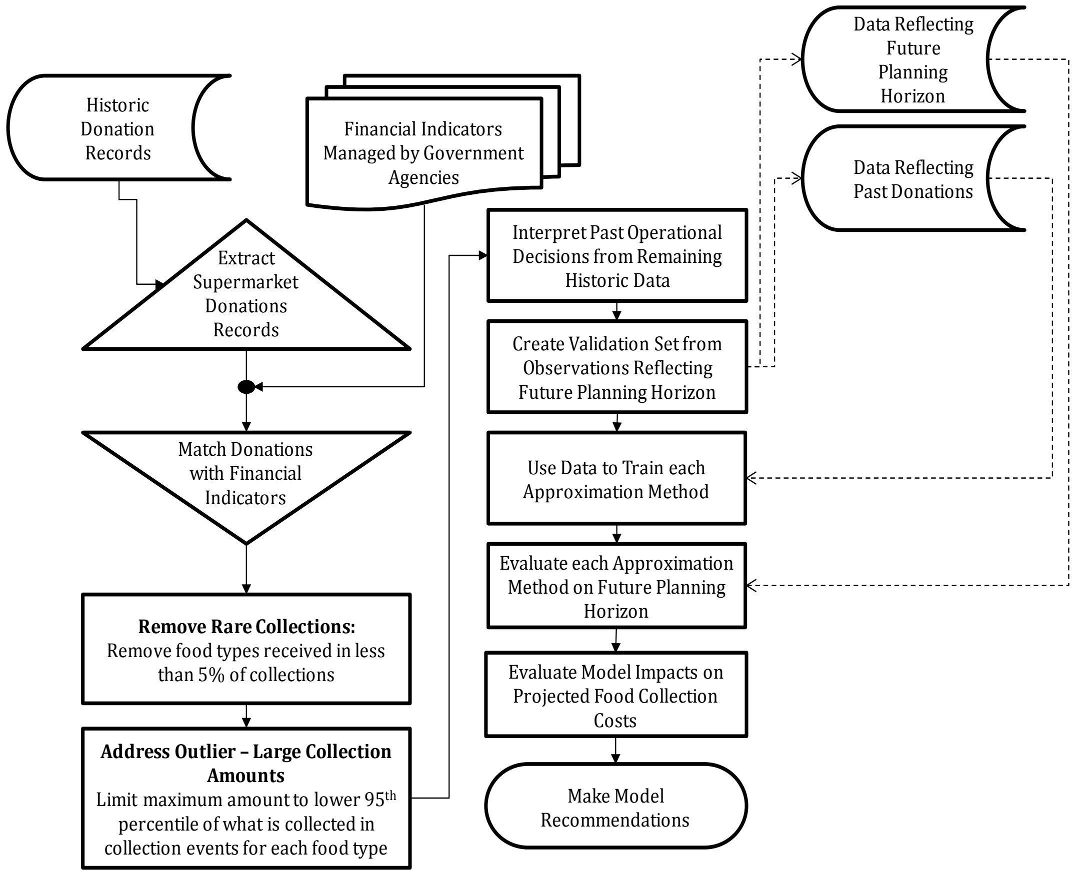


Figure 2.4. Data Pre-processing, Model Training, and Experimentation Processes Implemented for the Computational Study

2.5.3. Experiments

Two experiments are performed to evaluate the accuracy of the proposed forecasting approaches. In the first experiment, the predictive errors of the four forecasting models are compared using a data set not included in model training (i.e. known history). The second experiment evaluates the impact supply forecasting can have on operations decisions. Transportation costs are evaluated when vehicles are scheduled for food collection using predicted collection amounts verses estimates based on perfect information. This experiment is motivated by transportation savings that can be realized by reducing uncertainty in supply availability. If the food bank had perfect information (i.e. food availability at supermarket branches known with certainty), planners could conceivably develop schedules that lower transportation costs and maximize vehicle utilization more readily than when they there is greater supply uncertainty.

2.5.3.1. *Experiment #1: Forecasting Efficacy.* Each of the four forecasting methods is constructed using observations that occur prior to April 25, 2011. For the MLP-NN, the complete list of training conditions is given in Table 2.2. MLP-NN training involves an iterative process in which the arc weights are assigned a set of measurements. The training process continues until an MLP-NN is identified as providing its best approximation for the relationship between inputs and outcomes. The best approximation is based on the minimization of the generalization error as measured by the mean squared error (MSE). The calculation for the MSE is provided in Equations (2.7), where n represents the number of observations.

$$MSE = \frac{\sum_{o=1}^{n}\left(Y_o - \hat{Y}_o\right)^2}{n} \tag{2.7}$$

The Scaled Conjugate Gradient back propagation algorithm is used for model training due to its relatively fast convergence when used for large data sets (Moller 1993).

This study prevents over-fitting by using an early-stopping criteria which terminates the aforementioned training process when deemed appropriate. The incorporation of this condition requires the model to make projections for unobserved system observations after each training epoch. This condition is satisfied when the generalization error associated with projections for the unobserved system events is perceived to have reached a global minimum. The incorporation of an early-stopping condition requires that all available system observations are partitioned into two or more data sets. One of these data sets is used to train models. The second data set (i.e. test set) is used to represent previously unseen system outcomes. A popular data partitioning approach is to assign observations to training and test sets randomly. Typical partitions include 50/50, 60/40, and 67/33 allocations to training and test sets, respectively. In some instances, a third data set (i.e. validation set) is used to provide a second set of previously unseen system outcomes. The minimum generalization error across the validation set is useful in understanding system stability. When the epoch where the minimum generalization error occurs for test and validation sets is the same, the system is assumed to be stationary. When the epochs are different, the system is assumed to be non-stationary. The third data set is especially useful in instances when the system is expected to change with respect to time. In this study, the validation data set consists of observations in the period April 25 – 30, 2011.

Table 2.2

Training Parameters for the Quasi-Greedy Algorithm

Training Parameters	
Number of Runs Per Data Partition	5
Data Partitions	{50/50, 60/40, 67/33}
Maximum Training Time per Run	10 minutes
Early Stopping Criteria	150 consecutive epochs without improvement
Maximum Training Epochs	1000
Activation Function	
Forward to Hidden Node	Logistic Sigmoid
Forward to Output Node	Weighted Linear Sigmoid
BP Algorithm	Scaled Conjugate Gradient

While useful for training, the MSE does not give an accurate assessment of overall predictive error. Therefore, the mean absolute error (MAE) and coefficients of determination (R^2) are used to evaluate forecasts efficacy over future collection periods. The MAE, as shown in Equation (2.8), gives an estimate for the expected difference between predicted and actual system outcomes.

$$MAE = \frac{\sum_{o=1}^{n} \left| Y_o - \widehat{Y_o} \right|}{n} \tag{2.8}$$

R^2 measures the proportion of the variation in the actual collection amount that can be attributed to the observed characteristics defining a collection event. This measurement is bound by $0 \leq R^2 \leq 1$. Models that are more effective at accounting for variation in food availability are assigned an R^2 which is closer to 1. Models that are less effective at accounting for variation in food availability are assigned an R^2 which is closer to 0.

2.5.3.2. *Experiment #2: Impacts of Forecasts on Transportation Decisions.* Food collections for the data set are grouped by the date of their occurrence and the collecting warehouse. After grouping data, aggregate collection amounts are estimated for each food type using each of the forecasting methods. Lastly, the saving heuristic of Clarke and Wright (1964) is used to determine appropriate routing solutions for a set of capacity constrained vehicles. For this experiment, the rental cost per vehicle use is $1000. The high rental cost is used to promote the reduction of vehicles whenever possible. This allows the resulting assignments to only be impacted by the aggregate collection amount, tour duration, and limitations of load capacitated vehicles. The fuel cost associated with a vehicle travel is $.40 per mile. The refrigeration cost is $4 per hour. The collection time at each supermarket is negligible. The tow capacity for each vehicle is set at 10,000 lbs.

Both experiments are executed using customized MATLAB code. The first experiment uses code based on the Neural Network Toolbox. The second experiment is developed without the use of an additional toolbox. Both are run on a computer with a processing speed of 2.99 GHz, and 3.00 GB of RAM.

2.6. Results

After completing all preprocessing, four food types remain: grains, frozen meats, frozen mixed foods, and produce. The average, standard deviation, and coefficient of variation for each food type are given in Table 2.3. After preprocessing, 10,464 records remain. There are 10,336 records that reflect known history. The remaining 128 observations reflect the future planning period. the maximum collection amounts of grains, frozen meats, frozen mixed foods, and produce are 2500, 360, 2200, and 2500 pounds, respectively.

Table 2.3

Summary Statistics for Data Set after Preprocessing (By Food Type)

	In-Kind Food Type			
Parameter	*Grains*	*Frozen Meats*	*Frozen Mixed Foods*	*Produce*
Maximum Collection Amount	2500	360	2200	2500
Average Collection Amount	414.17	117.94	358.70	321.41
Standard Deviation	987.27	149.26	935.03	973.00
Coefficient of Variation	2.38	1.27	2.61	3.03

2.6.1. Performance of Forecasting Methods

The characteristics of the selected MLP-NN models are summarized in Table 2.4. The selected model configuration is read from left to right. The model selected for grains, for example, has 288 nodes in the input layer, 4 nodes in the first hidden layer, 2 nodes in the second hidden layer, and 1 node in the output layer. The 60/40 partition yielded the best MLP-NN models for each food type. Readers, therefore, should not conclude that this partition is the most effective in all situations. Three of the four selected models are constructed using only a single hidden layer. This supports the idea that neural networks with a single hidden layer and a sufficient number of neurons can represent any function (Gallant and White 1998; Hornik et al. 1989; Hornik 1991; Lippmann 1987). While the model selected for grains does not reject this idea, it does suggest that model configurations that have more than one hidden layer should be considered in the model selection process.

Table 2.4

Model Characteristics for Selected MLP-NN Models

Performance Measure	**In-Kind Food Type**			
	Grains	*Frozen Meats*	*Frozen Mixed Foods*	*Produce*
Selected Model	[288-4-2-1]	[288-2-1]	[288-7-1]	[288-8-1]
Best Epoch - Test Set	177	389	230	246
Best Epoch - Validation Set	151	30	254	321
Total Epochs	327	539	380	396
Test Set Partition %	0.4	0.4	0.4	0.4
Training Set Partition %	0.6	0.6	0.6	0.6

The total number of epochs required to train each model is less than the maximum permitted. Similarly, the termination conditions for training were reached before the maximum number of permitted training epochs. This indicates that the models were terminated as a result of obtaining what is perceived as the minimum MSE for test set observations. The table also shows the best epoch for the test set is different from that of the validation set. This suggests the system is non-stationary.

Table 2.6 shows the overall predictive error for each approximation method. The most difficult to forecast food type is frozen meats. This is also the food type with the lowest collection amount and the smallest coefficient of variation. This could suggest that since the amount of food generated is so small, a less weighted performance measure (i.e. MAE) may have produced better results. Another explanation for this is that initial months of food collection which were performed for frozen meats reflect different food collection practices than the rest of the occurrences. Frozen meats are the only food type of those remaining after pre-processing that is collected in 2006 and 2007. Nonetheless, each of the forecasting methods is better than estimating the collection amount using an overall average that is indiscriminate of the supermarket branches.

Table 2.5

Model Approximation Performance (By Food Type)

Model	**Data Set**	**Measure**	**In-Kind Food Type** *Grains*	*Frozen Meats*	*Frozen Mixed Foods*	*Produce*
MLP-NN	Known History (Training)	R^2	0.7815	0.6136	0.8543	0.8378
		MAE	165.59	46.38	101.40	6.69
	Known History (Test)	R^2	0.7896	0.5711	0.8720	0.8301
		MAE	168.50	47.28	103.88	6.64
	Next Period (Validation)	R^2	0.6936	0.5380	0.7375	0.7948
		MAE	331.84	47.52	253.80	12.92
MLR	Known History	R^2	0.5067	0.2654	0.6605	0.6574
		MAE	230.53	52.69	155.66	8.62
	Next Period	R^2	0.6907	0.4280	0.7441	0.6979
		MAE	366.72	55.63	315.62	14.65
Averages	SM – Next Period	R^2	0.6285	0.3509	0.7043	0.4623
		MAE	366.87	56.15	264.24	559.84
	SMWH – Next Period	R^2	0.6430	0.3572	0.7095	0.4787
		MAE	353.09	55.45	260.20	555.44

Based on the values for R^2, most of the variability in the collection amounts of each food type can be attributed to the set of independent variables when using the MLP-NN models. The exception is frozen meat, which is marginally above 0.5. In terms of MAE, MLP-NN models are superior to each of the other forecasting methods across each data set. This supports the idea that the relationship between observable system characteristics and the amount of each food type collected is best approximated through non-linear functions. The MAE appears to be at its lowest for training set projections. While very close to those of the training set, the approximations for the MAE for test set observations are slightly higher. The greatest MAE is observed for the validation set. This is expected of forecasting models, as it indicates that the model projections are most accurate when assessing known history and less accurate when making predictions for observations not used in training as well as taken from a different planning horizon. The similarities between the performances of selected models when forecasting grains and produce suggest that the system characteristics and predicted outcomes are very similar in both data partitions. In contrast, there is a noticeable change in the MAE observed for past and future observations. This supports the idea that the system is non-stationary.

MLR appears to produce very poor projections for the known history. Only grains, frozen mixed foods, and produce have coefficients of determination greater than 0.5. This suggests that less than half of the variability in the amount of frozen meats collected through a donation can be attributed to system inputs when the

MLR model is used. The R^2 for frozen mixed foods and produce, although higher than other food types, is less than 0.7. Each of these models is inferior to the MLP-NN model projections corresponding to the same food type in terms of both R^2 and MAE. A very interesting finding is that MLR models appear to make much better projections for the next period than for the known history. This improvement suggests that certain observations from the next planning period better suited for the model than those used to train the model. This is concerning because one would expect model accuracy to either remain the same or decrease when used to make estimates for an unobserved planning period. Given the unexpected behavior of MLR models, one can confidently observe that the systems represented by the MLR models are non-stationary. Since such a drastic improvement in R^2 is observed, one can also observe that MLR models are inappropriate for forecasting food availability. Given the results for the MLP-NN models, the likely reason for MLR models being outperformed is their inability to account for interactions between system characteristics when accounting for variability in the collection amount. This is a limitation of linear causal models.

SM and SMWH averages both have coefficients of determinations for frozen meats and produce that are less than 0.5. The MAE obtained using the SM average to estimate future receipts of grains and frozen meats yields similar result to using MLR. The SM average produces better results than MLR when used to estimate frozen mixed foods. The performance of both the SM and SMWH averages are noticeably low for produce. This is believed to be because the averages do not consider some of the parameters for which causal models like MLR and MLP-NN can adjust. This suggests that both the SM average and SMWH averages can greatly inflate food bank estimates. The MAE obtained from using both averaging methods for grains, frozen meats, and frozen mixed foods are either comparable or better than those obtained using MLR. Nonetheless, the R^2 value for each averaging method is inferior to its MLR counterpart. This is believed to be due to the variability in the observed collection amount which may be related to factors not considered through these averaging methods (refer to Table 2.3).

2.6.2. Impacts on Transportation Costs

When scheduling food collections using the aggregate collection amounts predicted by each forecasting method, only estimates made for the Greenville and Wilmington branches result in inflated transportation costs. For the Greenville branch, when provided with perfect information, only 10 vehicles are required and transportation costs are $18,073.87. When using MLP-NN or MLR projections as the basis for scheduling, 11 vehicles are required. The associated transportation cost is $19,416.28. Three additional vehicles are required when schedules are based on estimates obtained using SM or SMWH averages. The transportation cost estimates using the SM average are $22,493.35. The costs using the SMWH average are $22,386.46. Figure 2.5 shows both the number of customers visited on each scheduled delivery day (see line graph) as well as the cost increases that would be experienced using one of the approximation methods rather than having perfect information (see bar graph). The figure also identifies the number of supermarkets visited on each day. While all of the forecasting models results in inflated transportation costs, collection estimates determined from MLP-NN and MLR result in lower costs than a more naïve approach.

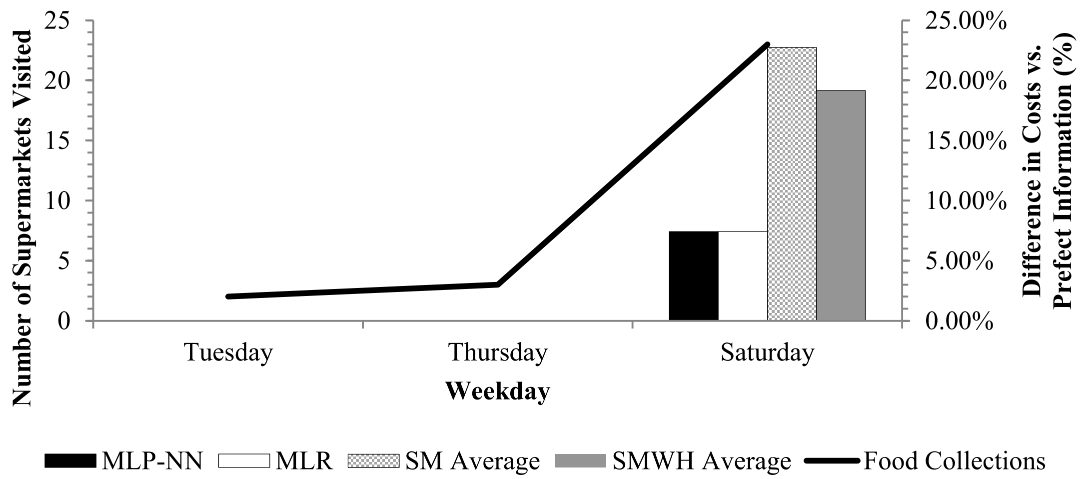


Figure 2.5. Food Collections and Percentage Change in Estimated Transportation Costs for Greenville Branch

For the Wilmington branch, the SM and SMWH averages result in the scheduling of 19 vehicles for food collection on Saturday. This is believed to be the result of both an increased number of supermarkets visited on Saturday and inflated demand estimates for SM and SMWH averages. Only 10 vehicles are required on Saturday when using the selected MLP-NN or MLR models. Figure 2.6 gives the total number of supermarkets visited and the difference in transportation cost estimates when using perfect information verses each forecasting method. The estimates for the MLP-NN and MLR models do not appear because they match the costs obtained under the perfect information case.

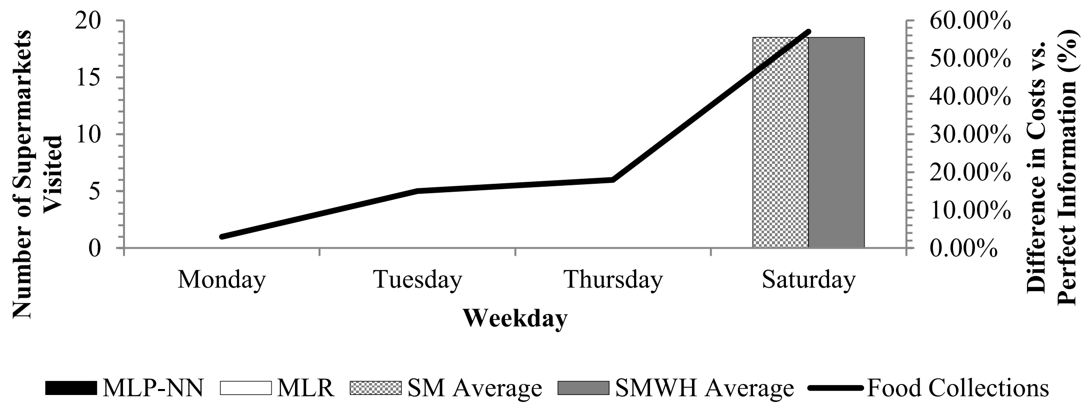


Figure 2.6. Food Collections and Percentage Change in Estimated Transportation Costs for Wilmington Branch

Based solely on the transportation costs incurred for each system, the results show that MLP-NN models or the MLR models produce results closer to the perfect information case. These forecasting methods are likely more attractive than SM and SMWH averages because they are better at taking into account system variability.

2.7. Managerial Impacts

No forecasting method is perfect. The results for this computational study show that all forecasting methods overestimated total collections for the future planning period. However, the amount for which the supply is overestimated is less for the MLP-NN models. Furthermore, the operational impact from using MLP-NN is no more than 10 percent in terms of additional transportation costs.

While not considered in this investigation, operational impacts would also result from underestimating supply. Among these impacts are lost food replenishment opportunities. Underestimating supply could result in one or more vehicles reaching their tow capacity before completing their tour. When vehicle tours are developed off-line, which is the case for the second experiment, the vehicles return to the depot prior to collecting food from the remaining supermarket branches. The receipt of these additional supplies could have been helpful in some instances, especially when inventory is very low. In addition, when inventory replenishment is less critical, underestimated supply suggests that operating funds could have been allocated more effectively. Nonetheless, the findings of this research suggest that MLP-NN models are more appropriate for forecasting in-kind donations than other forecasting methods considered in this study because it is superior at minimizing the overestimation and underestimation of supply.

Despite it providing better results than other approximation methods, each of the selected MLP-NN models interpret the system representing food availability as non-stationary. This non-stationarity suggests that models may not accurately account for system variability as a result of some unobserved trend. It is recommended that planners limit the planning horizon to one calendar week when using the MLP-NN models to forecast food availability. Including the most recent system data allows the models to include data that may be pertinent in capturing some degree of system variability.

2.8. Conclusion

This research explores different forecasting methods that can be used to overcome the supply uncertainty experienced by regional food banks and food recovery organizations when attempting to estimate available supplies at supermarket locations. This study evaluates the impacts of using different forecasting methods to approximate the amounts of different in-kind food types collected in isolated collection events when information is not shared by supermarket branches. Forecasting methods considered are the average amount received by the regional food bank from a supermarket branch, the average amount received by a specific food bank warehouse from a supermarket branch, the predicted amount when all attributes of the collection event are evaluated using multiple linear regressions, and the predicted amount received when all attributes of the collection event are evaluated using selected multi-layer perceptron neural network models. The multi-layer perceptron neural network models are selected using a quasi-greedy heuristic. In addition, the models are constructed using historic collection records along with indicators of financial wellness for counties in which the donations are generated. Results from our investigation suggest that the selected multi-layer perceptron neural network models are superior to each of the other forecasting methods both in terms of prediction accuracy and impacts on transportation costs. The results also suggest that with respect to forecasting, the two methods utilizing supermarket averages outperform the standard multiple linear regression model when trends may not be observed through linear models. These averages, however, can greatly inflate the projected collection amounts when the collection amount is not random and can lead to incorrect estimates for food collection requirements.

The findings of this study should promote further research into in-kind donation estimation. The system representation developed for this study is based on available collection records, discussions with food bank and supermarket personnel, and public information maintained by government agencies. Similarly, while not inclusive of all forecasting methods, this study evaluates both traditional and artificial intelligence approaches that are relevant to the research problem.

While this research does not make the assertion that the prescribed MLP-NN selections produce the most accurate predictions, it does make a strong case for using more advanced forecasting methods when predicting the amounts of food type available for receipt in the next planning period. Future research will continue to study the data set and incorporate other supervised learning methods which are more effective at accounting for system variability. Research extensions should also develop fiscally-responsible inventory-based vehicle routing strategies resulting from forecasts.

CHAPTER 3

Formulations for the Periodic Vehicle Routing Problem with Backhauls

3.1. Introduction

Vehicle routing problem is one of the most research-studied in operations research. Since the introduction of the capacitated vehicle routing problem by Dantzig and Ramseur (1959), the original problem has been adapted to unique challenges experienced in different industries. These adaptations often result in the introduction of new problem variants. Among these variants are the vehicle routing problem with time windows (VRPTW), the vehicle routing problem with backhauls (VRPB), and the periodic vehicle routing problem (PVRP). The VRPTW builds upon the foundation of the capacitated VRP by adding the requirement that customer deliveries must be satisfied within pre-determined time intervals. The VRPB extends the capacitated VRP such that vehicles are used to satisfy two sets of customers, one requiring service through the delivery of commodities from a depot (i.e. linehauls) and the other requiring service through the on-site collection of commodities for deposit at the depot (i.e. backhauls). An important feature of this problem is that the two commodity types cannot be on a truck at the same time. The PVRP relaxes the assumption that all customers are served in a single day. Instead, the vehicles are scheduled to make a collection (delivery) at customer locations on one or more days over a finite planning horizon. Given the overall importance of minimizing transportation costs across different industries, a considerable amount of literature is published that is related to at least one of these routing problem generalizations. These extensions are often based on two or more existing generalizations. Examples of these papers include the vehicle routing problem with backhauls and time windows (see e.g., Zhong and Cole (2005)) and the periodic vehicle routing problem with time windows (see e.g., Nguyen et al.(2014); Michallet et al. (2014)).

While there is considerable literature published related to variants of the VRPB and VRPTW, as well as growing interest in the PVRP, the periodic vehicle routing problem with backhauls (PVRPB) appears to one that has been overlooked. This generalization has practical applications in numerous industries, particularly in distribution networks where both the suppliers and customers for a warehouse are located in the same geographic region. Examples of possible applications include (1) food recovery and distribution operations managed by supermarket warehouses and local charities such as food banks, (2) distribution and collection routes coordinated by manufacturers to promote the safe receipt and disposal of pharmaceutical drugs, and (3) mail carrier pickup and delivery services.

This dissertation chapter provides the first formal introduction of the PVRPB. In doing so, three objectives are met in terms of understanding the problem variant. The first objective is to provide a problem description and model formulations for three problem variants. The second objective is to understand the impacts of each problem variant in terms of the objective function value and the computational complexity of different instances. Vehicle routing problems are based on the traveling salesman problem, a NP-complete problem where there is one vehicle with infinite capacity. Vehicle routing problems are much more difficult to solve as they present multiple vehicle assignment options and add load constraints to available vehicles. As such, obtaining even a feasible solution to larger problems could be problematic (see e.g., Mingozzi et al. (1999)). The PVRPB presents an even more complex problem which, depending on problem complexity, might not be solved to optimality. The third objective is the introduction of tour limitation constraints which when added to the model formulations, permit commercial solvers to identify good solutions for many of these problems.

While making these contributions, two research questions are addressed. The first research question evaluates the ability of commercial solvers to find solutions for these routing problems. Using the obtained solutions as a baseline, a second research question determines the effectiveness of adding tour limitation constraints to each model formulation.

The remainder of the dissertation chapter is organized as follows. Section 3.2 provides a brief review of literature specific to the PVRP and VRPB in terms of their computational complexity and related problem variants. Section 3.3 presents a formal definition for the PVRPB, including a model formulation. Section 3.4 discusses how the model formulation can be expanded into two variants of the PVRPB, in particular, the periodic vehicle routing problem with backhauls and time windows (i.e. PVRPBTW) and the heterogeneous fleet periodic vehicle routing problem with backhauls (i.e. HPVRPB). Section 3.5 introduces a set of constraints that when added to the formulation allows feasible solutions for the PVRPB, PVRPBTW, or HPVRPB to be found for larger routing problems when provided with a sufficient number of vehicles. Section 3.6 presents a set of experiments based on the aforementioned research objectives. Section 3.7 discusses the results of the experiments, and Section 3.8 provides managerial insights that are obtained from the experiments. Section 3.9 summarizes the research findings and identifies opportunities for future research extensions.

3.2. Literature Review

A review of literature is provided to understand routing problems that are similar to the PVRPB. The review was initialized by searching for pre-existing literature that addresses problems that might be classified as a PVRPB. An extensive review of this generalization is completed using online engineering databases including Compendex, Web of Science, and Google Scholar. Key words used in the search include "period vehicle routing problem" AND "backhauls", "PVRP" AND "backhauls", "period distribution routing problem" AND "collection", and "multi-day routing" AND "backhauls" AND "linehauls". These queries were performed as late as January 5, 2014 to ensure the inclusion of the most recent literature. Manuscripts written in languages other than English are excluded. The abstracts for manuscripts obtained through the search are reviewed to identify the routing problems that match the characteristic of the PVRPB. Those which appear to be closely related to the problem were read in their entirety.

Davis et al. (2014) provide the only known contribution that in some way addresses this problem variant. The researchers utilize a variation of the PVRPB to schedule the collection and distribution of charitable foods. The problem minimizes the total vehicle travel distance for food bank vehicles when delivering commodities to remote charitable agencies and collect needed food items from high-volume donors (i.e. supermarkets, food manufacturers, etc.). A two-phased model-based heuristic is implemented to (a) assign the charitable agencies to a food delivery point and (b) develop routes that allow food bank vehicles to both deliver allocated commodities to the food delivery points and make collections at the high-volume donors over the course of a 5-day planning horizon. The approach limits vehicles to making only one delivery per tour.

The PVRPB is anticipated to be a hybrid model based on the PVRP and VRPB. Accordingly, a concise, yet comprehensive review of key literature for both generalizations is provided. Readers interested in a review of the PVRP or VRPB are referred to Francis et al. (2008) and Goetschalckx and Jacobs-Blecha (1989).

3.2.1. PVRP

The PVRP is a routing problem which consists of a set of customers requiring transportation services one or more times over a multi-day planning horizon. Every customer transportation service results in every customer being served by a delivery (or collection). All services are performed by vehicles which have both pre-defined maximum tow capacities and tour durations. Depot distribution and collection capabilities are assumed to be infinite. The CVRP is an instance of the PVRP where there is only one day in the planning horizon and all customers are visited once. This causes the PVRP present a more complex extension of the CVRP.

Previous literature expresses a general consensus that the PVRP is a combination of two classical problems: an assignment problem and a routing problem (Baptista et al. 2002). The assignment problem is used to allocate customers to a preliminary set of arrival day(s) and/or a vehicle route. The routing problem follows by searching for the most efficient sequence in which customers assigned can be served by capacitated vehicles. The solution methods that are explored include exact methods (EM), classified as classic heuristics (CH), and metaheuristics (MH). The customer assignments for each problem include an assignment problem (AP) as well as a geometrically-based assignment problem (GAP). The GAP is a special case of the assignment problem that uses geometric approximations for travel distance to make its selections. The routing problem classification states whether the authors determine route configuration by solving separate traveling salesman problems (TSP) or vehicle routing problems (VRP). A classification of each of these routing approaches is given in Table 3.1. The table highlights EM and CH approaches. MH approaches are discussed in Chapter 4.

Table 3.1

Problem Decomposition Used to Solve the PVRP

	Solution Method		**Customer Assignment**		**Routing Problem**	
Author(s)	EM	CH	AP	GAP	TSP	VRP
Beltrami and Bodin (1974)		●	●			●
Russell and Igo (1979)		●	●			●
Christofides and Beasley (1984)		●	●		●	
Tan and Beasley (1984)		●		●	●	
Russell and Gribbin (1991)		●		●		●
Gaudioso and Paletta (1991)		●	●			●
Baptista et al. (2002)						
Francis et al. (2006b)	●		●			
Mourgaya and Vanderbeck (2007)	●			●		

3.2.1.1. *CH Contributions.* Many early publications addressing the PVRP incorporated classical heuristic methods. Beltrami and Bodin (1974) use optimization methods to assign customers to delivery day combinations. The initial assignment is followed by solving vehicle routing problems for each day of a planning horizon. In their investigation, schedules which limited customers to being served 3 or 6 days of the planning horizons are considered. Russell and Igo (1979) explore more flexible delivery frequencies. Christofides and Beasley (1984) present the first model formulation for the PVRP; however, they solve the problem using a three phased heuristic. The first phase of the heuristic assigns customers to clusters which indicate a specific vehicle used on a given day of the planning horizon. These clusters are based on the least cost heuristic of Eilon and Christofides (1971). Solutions are improved by solving separate periodic traveling salesman problems (PTSP) problems for each day. Each tour of the PTSP is solved separately using 2-opt intra-route improvement heuristics introduced by Lin and Kernighan (1973). Different customer delivery day combinations are explored by re-incorporating the least cost heuristic followed by re-solving the PTSP. Tan and Beasley (1984) and Russell and Gribbin (1991) present heuristics based on the random seed generation procedure of Fisher and Jaikumar (1981). This random seed generator inserts customer into vehicle routes based on the change in cost of a round trip from the depot through each seed point. Gaudioso and Paletta (1991) present a unique application of the PVRP which is designed to balance workload among vehicles. Initial routes are constructed by assigning each customer to a multi-day delivery schedule, and one of the available vehicles for each day using the delivery amount per day. Assignments are completed based on a first-fit decreasing bin packing procedure. After an initial assignment, customers are reassigned to vehicles based on the application of the 2-opt heuristic of Lin and Kernighan (1973) for both inter-route and intra-route exchanges occurring on the same day, and a smoothing algorithm designed to balance workload on peak delivery days. After each customer reassignment, tours are reconfigured according to a first-fit decreasing bin packing procedure.

3.2.1.2. *EM Contributions*. Few researchers solve the PVRP using exact formulation methods. Francis et al. (2006b) present an exact solution method for solving the problem based on the Lagrangian relaxation of an integer programming formulation of the problem. The authors reduce the dimensionality of the problem by limiting the service schedules to permissible day combinations where a given customer is served to a set of disjoint day combinations and a single schedule which includes all days in the planning horizon. Through the Lagrangian relaxation, the problem is decomposed into a capacitated assignment problem and a separate TSP for each day. Further improvements to the integer solution are made using a branch-and-bound algorithm. This approach was applied to problem instances with up to 50 nodes, each of which were solved to within 2% of the optimal solution. Mourgaya and Vanderbeck (2007) use column generation to solve a special case of the PVRP where the objective function promotes a balanced vehicle workload and improved vehicle regionalization. The problem is formulated as a generalized assignment problem which groups customers into geometrically-dispersed clusters. All customers assigned to the same customer are served by the same vehicle route. The formulation uses approximations to determine improvements to vehicle travel time based on the total travel time incurred by a vehicle visiting the imaginary central point for each cluster before returning to the depot. After providing the formulation, its limitations are discussed in terms of estimating system costs and problem relaxation. These limitations are overcome through a Dantzig-Wolf reformulation which assigns each tactical scenario to customers and customers to clusters. The reformulated problem is solved using column generation to minimize the cost of serving the specified clusters. Despite reformulation, the problem remains difficult to solve with reported optimality gaps of 14 – 30%.

3.2.1.3. *MH Contributions*. Metaheuristic search procedures comprise the most prevalent method for solving the PVRP. Those that have been effective for the PVRP include tabu search (see e.g. Rusdiansyah and Tsao (2005)), variable neighborhood search (see e.g., Hemmelmayr et al. (2009b)), scatter search (see e.g. Alegre et al. (2007)), and genetic algorithms (see e.g., Vidal et al. (2012)). Initial solutions are based on a preliminary assignment of customers to vehicle routes and visit schedules. Most metaheuristics surveyed for the PVRP use construction algorithms similar in form to Beltrami and Bodin (1974) and Gaudioso and Paletta (1991). Exceptions to this form of construction include Vidal et al. (2012) and Nguyen et al.(2014) who both utilize a two-vector representation to solve the PVRP, periodic vehicle routing problem with time windows (PVRPTW), and multi-depot vehicle routing problem (MDVRP) through a hybrid genetic algorithms. The initial construction and search process associated with each, however, is based on augmenting the service schedule assigned for each customer, followed by updating the affected vehicle routes. A more in-depth discussion of the metaheuristics approaches is provided in Chapter 4.

3.2.2. VRPB

The VRPB presents a routing problem where two sets of customers are served through the capacitated vehicle fleet. One set of customers is satisfied by the delivery of a set of commodities from a depot (i.e. linehauls) and the other set of customers is satisfied by the collection of commodities which are deposited into the depot (i.e. backhauls). Vehicles are limited by both their tow capacity and pre-defined maximum tour durations. Those which deliver linehaul items contain all commodities that are requested by the customers they serve. As such, a key assumption of this problem is that the depot has infinite distribution and collection capabilities. The CVRP is an instance of the VRPB where there are only linehaul or backhaul customers, making the VRPB a more complex extension of the CVRP. Yano et al. (1987) present an exact algorithm which places special constraints on the number of total customers that can be satisfied on a vehicle tour. Toth and Vigo (1997) present an exact algorithm to solve the VRPB with both symmetric and asymmetric travel distances. Each of these algorithms was only able to identify the optimal solution for problems with 100 or fewer customers. Furthermore, convergence required more time than heuristic methods. Gelianas et al. (1995) present an exact algorithm to solve instances where there are customer-specific time windows. Less restrictive heuristic algorithms for the VRPB are also presented in literature. The publications include variant with time windows (Duhamel et al. 1997), heterogeneous fleet (Tutuncu 2010), and multiple depots (Wang and Li 2009, Wang et al. 2009).

3.2.3. Unaddressed Research Area

Since the PVRPB has not been studied previously, a direct application of one of the pre-existing heuristics designed to solve PVRP or VRPB is premature. In the absence of formal heuristic methods, the use of an exact method may be sufficient. Advances in computer processing capabilities, coupled with improved problem relaxation, cutting plane, and branch and bound methods allow optimal solutions to be obtained for growing number of difficult problems (see e.g., Toth and Vigo 2002). Many of these methods are incorporated into commercial software applications. The ability of modeling software to solve this understudied problem has yet to be evaluated. Additionally, the characteristics of problem extensions where customers are served within pre-determined time intervals or where collections and deliveries are performed using a heterogeneous vehicle fleet should be studied. Practices that reduce the complexity of the routing problem to find good solutions in a reasonable time given its complexity may be identified.

3.3. Model Formulation

The PVRPB is based on the following assumptions. A fleet of capacitated vehicles travel across a network in order to satisfy two sets of customers. The first set consists of customers requiring delivery service whereas the second consists of customers requiring collection service. Each customer who requires both delivery and collection service is treated as two separate customers. Vehicle tours originate at a single depot. This depot is the origin for all dispatched commodities and the destination for collected commodities. The first group of customers is satisfied by receiving commodities dispatched from the depot through linehauls. The second group of customers is satisfied by having commodities collected for backhaul to the depot. The amount of food that is dispatched or collected at each location is assumed to be known with certainty. Hence, the service time for customer deliveries (collections) are assumed to be known with certainty. The delivery (collection) requirements of customers are satisfied by vehicles over a specified number of days while strictly adhering to tow capacity and tour duration restrictions. In addition, while permitted to serve both linehaul and backhaul customers on the same tour, vehicles cannot simultaneously contain commodities from the two sets of customers. Finally, the depot is assumed to have infinite capacity.

A mathematical formulation for the PVRPB is now introduced. The network associated with this formulation is represented through the graph $G = (\boldsymbol{N}, \boldsymbol{A})$, where $\boldsymbol{N}$ corresponds of the set of customer locations with $i \in N$corresponding to customer locations and $i = \{0\}$ indicating the depot. Customers served through linehauls are members of the set $\boldsymbol{N_L} \subset \boldsymbol{N}$ whereas those serviced by backhauls are members of the set $\boldsymbol{N_B} \subset \boldsymbol{N}$, where $\boldsymbol{N_L} \cap \boldsymbol{N_B} = \emptyset$ and $\boldsymbol{N_L} \cup \boldsymbol{N_B} = \boldsymbol{N}$. Thus, if there are no linehaul or backhaul customers, the problem reduces to a PVRP. The set $\boldsymbol{A}$ represents arcs along which vehicle travel, represented by the ordered pair (i, j). Sets $\boldsymbol{V}$ and $\boldsymbol{D}$ define the sets of vehicles and days included in the planning horizon, respectively. Individual vehicles and days within the planning horizon are represented by $v \in \boldsymbol{V}$ and $p \in \boldsymbol{D}$, respectively. When traveling between two nodes, a distance of d_{ij} and a travel time of t_{ij} are incurred. Each time that a vehicle arrives at a customer, a service time of b_i is incurred. The amount of the commodity received (linehauls) or distributed (backhauls) through this arrival is given by q_i. The total number of times a customer must be visited is given by R_i. Customers are limited to being served according to a set of permissible schedules $\boldsymbol{S}$*.* For a specific schedule, a_{sp} indicates service on a given day p by a value of 1 and no service by a value of 0. The capacity of each vehicle is given by Q.

Decision variables for this model are the following: Variables y_{is} identify the schedule selected for delivery and collection customers. Variables x_{ijvp} indicate whether vehicle v travels along arc (i, j)on day p. Variables u_{ip}indicate the time of departure from customer i on day p.

The PVRPB is formulated as a mixed integer linear programming (MILP) problem. The objective function for the PVRPB is based on costs incurred from vehicle usage, fuel consumption for travel, and utilizing vehicle refrigeration capabilities. The parameters associated with daily vehicle usage, fuel cost per mile, and refrigeration cost per hour are given by f, g, and h, respectively. The resulting formulation is as follows.

$$\begin{aligned} Minimize\ Z = f \sum_{p \in D} \sum_{v \in V} \sum_{j \in N} x_{0jvp} \\ + g \sum_{p \in D} \sum_{v \in V} \sum_{j \in N \cup 0} \sum_{i \in N \cup 0, i \neq j} d_{ij} x_{ijvp} \\ + j \sum_{p \in D} \sum_{v \in V} \sum_{j \in N \cup 0} \sum_{i \in N \cup 0, i \neq j} (t_{ij} + b_i) x_{ijvp} \end{aligned} \quad (3.1)$$

Subject to:

$$\sum_{p \in D} \left(\sum_{s \in S} a_{sp} y_{is} \right) = R_i \qquad \forall i \in \boldsymbol{N}, \quad (3.2)$$

$$\sum_{k \in N_L \cup 0} \sum_{v \in V} x_{kivp} - \sum_{s \in S} a_{sp} y_{is} = 0 \qquad \forall i \in \boldsymbol{N}, p \in \boldsymbol{D}, \quad (3.3)$$

$$\sum_{k \in N_L \cup 0} x_{kivp} - \sum_{j \in N \cup 0, j \neq k} x_{ijvp} = 0 \qquad \forall i \in \boldsymbol{N_L}, v \in \boldsymbol{V}, p \in \boldsymbol{D}, \quad (3.4)$$

$$\sum_{k \in N \cup 0} x_{kivp} - \sum_{j \in N_B \cup 0, j \neq k} x_{ijvp} = 0 \qquad \forall i \in \boldsymbol{N_B}, v \in \boldsymbol{V}, p \in \boldsymbol{D}, \quad (3.5)$$

$$\sum_{j \in N_L} \left(q_j \sum_{i \in N_L \cup 0} x_{ijvp} \right) \leq Q \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \quad (3.6)$$

$$\sum_{j \in N_B} \left(q_j \sum_{i \in N_B \cup 0} x_{ijvp} \right) \leq Q \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \qquad (3.7)$$

$$u_{ip} + (b_j + t_{ij}) x_{ijvp} - (1 - x_{ijvp}) T \leq u_{jp} \qquad \forall i, j \in \boldsymbol{N}, i \neq j, v \in \boldsymbol{V}, \; p \in \boldsymbol{D}, \qquad (3.8)$$

$$u_{ip} + t_{i0} \sum_{v \in V} x_{i0vp} \leq T \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \qquad (3.9)$$

$$b_j + \sum_{s \in S} (a_{sp} y_{js}) + t_{0j} \sum_{v \in V} x_{0jvp} \leq u_{jp} \qquad \forall j \in \boldsymbol{N}, p \in \boldsymbol{D}, \qquad (3.10)$$

$$\sum_{s \in S} y_{is} = 1 \qquad \forall i \in \boldsymbol{N}, \qquad (3.11)$$

$$\sum_{s \in S} x_{0jvp} \leq 1 \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \qquad (3.12)$$

$$u_{ip} \geq 0 \qquad \forall i \in \boldsymbol{N}, p \in \boldsymbol{D}, \qquad (3.13)$$

$$x_{ijvp} = \{0,1\} \qquad \forall i, j \in \boldsymbol{N} \cup 0, v \in \boldsymbol{V}, \; p \in \boldsymbol{D}, \qquad (3.14)$$

$$y_{is} = \{0,1\} \qquad \forall i \in \boldsymbol{N}, s \in \boldsymbol{S}. \qquad (3.15)$$

The objective function (3.1) minimizes the vehicle costs over the planning horizon. This cost is a function of the number of vehicles rented, the cost of fuel, and hourly usage costs for refrigerated trucks. Note that the first or third component of the objective function can be removed when vehicle usage or refrigeration is necessary. Constraints (3.2) ensure that the number of collections or deliveries specified for a customer is satisfied. Constraints (3.3) ensure that all collections and deliveries are satisfied by the arrival of a vehicle. Constraints (3.4) and (3.5) ensure the conservation of flow for customers served through deliveries and collections, respectively. Constraints (3.6) and (3.7) ensure that vehicle capacity is not exceeded while completing linehauls and backhauls, respectively. Constraints (3.8) maintain proper vehicle sequencing between successive customers on the same tour. As such is the case, they also eliminate sub-tours. Constraints (3.9) ensure that vehicle tours do not exceed their maximum duration. Constraints (3.10) identify the departure times from the first customers served through vehicle tours. Constraints (3.11) ensure that only one schedule is selected per customer served. Constraints (3.12) ensure that each vehicle is assigned no more than one tour per day. Constraints (3.13), (3.14), and (3.15) ensure that decision variables maintain non-negativity and binary values. In total, there are up to$(|\boldsymbol{N}| + 1)^2 \times |\boldsymbol{V}| \times |\boldsymbol{D}| + |\boldsymbol{N}| \times |\boldsymbol{S}|$ binary decision variables and $|\boldsymbol{N}| \times |\boldsymbol{D}|$ non-negative decision variables.

3.4. Formulations for Variants for the PVRPB

Many systems have additional characteristics that greatly impact routing decisions. For example, many customers can only be served within pre-determined time intervals. It is also common for vehicles to have different capabilities and/or incur different fixed and variable costs when used. Accordingly, MILP formulations for the periodic vehicle routing problem with backhauls and time windows (i.e. PVRPBTW) and the heterogeneous fleet periodic vehicle routing problem with backhauls (i.e. HPVRPB) are introduced for these additional system characteristics. The next two subsections discuss constraints added to the PVRPB formulations to solve each problem variants.

3.4.1. Time Windows

When time windows are considered, additional constraints are added to represent interval of time in which customers can be served. This interval is represented by an earliest time (denoted by e_i) and a latest time (denoted by l_i). This investigation considers instances where the time windows are soft, characterized by permitting vehicles to arrive at the customer early and wait until the earliest start time to begin service.

$$u_{ip} \geq (e_i + b_i) \sum_{s \in S} a_{sp} y_{is} \qquad \forall i \in \boldsymbol{N}, p \in \boldsymbol{D}, \quad (3.16)$$

$$u_{ip} \leq l_i \qquad \forall i \in \boldsymbol{N}, p \in \boldsymbol{D}. \quad (3.17)$$

Constraints (3.16) ensure that customers are not served prior to the earliest service start time. Constraints (3.17) ensure that customers are served prior to the latest permitted departure time. The constraints can also be observed to be appropriate for instances where a customer is not served on a specific day. It should be noted that the formulation for the PVRPBTW is only a lower-bound for the food bank routing problem because the formulation does not account for refrigeration costs incurred when vehicles remain idle while waiting until service is permitted to start.

3.4.2. Heterogeneous Fleet

When considering a heterogeneous fleet, equations for the PVRPB formulation are modified to reflect the different characteristics of each vehicle. In this research, the physical characteristics of vehicles such as their tow capacity, fixed costs, fuel efficiency, and refrigeration costs per hour are adapted as such,

$Q \rightarrow Q_v$, $f \rightarrow f_v$, $g \rightarrow g_v$, and $h \rightarrow h_v$. Accordingly, equations (3.1), (3.6), and (3.7) are adapted as seen in equations (3.18), (3.19), and (3.20) respectively.

$$minimize\ Z = \sum_{p \in D} \sum_{v \in V} \left(f_v \sum_{j \in N} x_{0jvp} \right) + \sum_{p \in D} \sum_{v \in V} \left(g_v \sum_{j \in N \cup 0} \sum_{i \in N \cup 0, i \neq j} d_{ij} x_{ijvp} \right) + \sum_{p \in D} \sum_{v \in V} \left(h_v \sum_{j \in N \cup 0} \sum_{i \in N \cup 0} t_{ij} x_{ijvp} + b_i x_{ijvp} \right) \quad (3.18)$$

$$\sum_{j \in N_L} \left(q_j \sum_{i \in N_L \cup 0} x_{ijvp} \right) \leq Q_v \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \quad (3.19)$$

$$\sum_{j \in N_B} \left(q_j \sum_{i \in N_B \cup 0} x_{ijvp} \right) \leq Q_v \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \quad (3.20)$$

3.5. Tour Limitation Constraints

The PVRPB and its variants are NP-complete because they represent problems that are at least as complex as the PVRP. It complexity is affected by the size of the underlying assignment and routing problems, both of which are also NP-complete (Karp 2010). The CVRP has underlying TSP and generalized assignment problem characteristics (Fisher and Jaikumar 1981); therefore, it is the more complex of the two NP-complete problems from which the multi-period routing problems are based. The TSP is the more difficult of the two underlying problems to solve. The complexity of this problem is a function of the number of customers visited on a vehicle tour.

A set of constraints which reduce the number of customers included in vehicle tours are now discussed. Constraints (3.21) and (3.22) are added to each of the model formulations to reduce the maximum number of linehauls permitted per vehicle tour (denoted by *MaxLH*) as well as the maximum number of backhauls permitted per vehicle tour (denoted by MaxBH). The PVRPB, PVRPBTW, and HPVRPB provided previously can be viewed as solving instances where *MaxLH* $=|\boldsymbol{N_L}|$and *MaxBH* $=|\boldsymbol{N_B}|$.

$$\sum_{\boldsymbol{k}\in\boldsymbol{N_L}\cup\boldsymbol{0}}\sum_{\boldsymbol{i}\in\boldsymbol{N_L}} x_{kivp} \leq MaxLH, \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}, \qquad (3.21)$$

$$\sum_{\boldsymbol{k}\in\boldsymbol{N_B}\cup\boldsymbol{0}}\sum_{\boldsymbol{i}\in\boldsymbol{N_B}} x_{kivp} \leq MaxBH, \qquad \forall v \in \boldsymbol{V}, p \in \boldsymbol{D}. \qquad (3.22)$$

The additional constraints are beneficial in obtaining a good solution using less computation time; nevertheless, these extended formulations for the PVRPB, PVRPBTW, and HPVRPB produce solutions that may be inferior to the objective function.

3.6. Experimentation

Two experiments are performed to study the PVRPB and extensions discussed. The first experiment compares the solution quality and computation times obtained using the PVRPB, PVRPBTW, and HPVRPB to solve different test scenarios. The second experiment studies the impacts of incorporating the tour limitation constraints to each of the PVRPB variants in terms of solution quality and computation time.

3.6.1. Case Study: Charitable Food Distribution and Collection Challenges of Food Banks

All test scenarios are based on a perceived opportunity for food banks to realize lower transportation costs, improve food access for remote charitable agencies, and promote safe food replenishment. Many remote agencies do not have refrigerated vehicles, preventing them from safely transporting perishable food items across the vast distance between their place of operation and the food bank. As a means of promoting safe food transfer, food banks serve one or more charitable agencies through shuttle services. Food bank vehicles are also essential for food replenishment. Food banks are the central warehouse through which many of the in-kind food contributions donated by for-profit companies (i.e. grocery stores, food manufacturers, etc.) are repurposed for charitable intent. These items are received by the food bank through on-site food collections performed by its own vehicles. Items that are collected include usable, yet unsellable food items such as items in dented cans, perishable foods approaching manufacturer-recommended sell-by dates, and test products which perform poorly in the market. After completing collection runs, vehicles return all commodities to the food bank, where they are inspected and stored for future distribution.

For the prescribed set of scenarios, charitable agencies are served through deliveries to shuttle locations. Each linehaul customer (i.e. a set of agencies served through a shuttle location) receives a single shipment of a prescribed amount of food measured in pounds. Each linehaul requires 1 hour. Local food donors represent the set of backhaul customers. Each donor contributes 300 pounds of food per collection. Only one collection is permitted at each backhaul customer per day and exactly three are made over the course of a five-day planning horizon. The service time for each collection is 30 minutes. Each vehicle tour is limited to 10 hours. The scenarios present instances where 5 to 42 customers are served. The characteristics of customers including the food delivery (collection) amounts for each are provided in Appendix B.1 and B.2. The distances and travel times between two locations are provided in Appendices B.3 and B.4. The designation of customers as being served through linehauls or backhauls is given in Table 3.2.

Table 3.2

Characteristics for each Test Scenario

			Model Parameters	
Scenario	$\mathbf{N_L}$	$\mathbf{N_B}$	**|V|**	**|D|**
1	C1-3	C:4-5	7	5
2	C:1-4	C:5-6	7	5
3	C:1-5	C:6-7	7	5
4	C:1-6	C:7-8	7	5
5	C:1-7	C:8-9	7	5
6	C:1-8	C:9-10	7	5
7	C:1-10	C:11-15	7	5
8	C:1-10	C:11-20	7	5

Table 3.2 (cont'd.)

9	C:1-14	C:15-25	7	5
10	C:1-14	C:15-30	7	5
11	C:1-14	C:15-35	7	5
12	C:1-14	C:15-42	7	5

The characteristics of each vehicle are provided in Table 3.3. The PVRPB and PVRPBTW are solved using the homogeneous vehicle fleet, whereas the HPVRPB is solved using the heterogeneous fleet type.

Table 3.3

Vehicle Characteristics

Fleet Type	Vehicle	Capacity (lbs.)	Fixed Cost ($/Use)	Fuel Efficiency ($/mile)	Refrigeration Cost ($/hour)
1	1 – 7	20000	150.00	0.40	1.25
2	1	20000	150.00	0.40	1.25
	2	18550	125.00	0.40	1.50
	3	15000	100.00	0.35	1.75
	4	15000	80.00	0.30	2.00
	5	12500	60.00	0.25	2.25
	6	5000	50.00	0.20	2.50

Table 3.3 (cont'd.)

2	7	1000	0.00	0.15	3.00

Homogeneous Fleet indicated by Fleet Type = 1
Heterogeneous Fleet indicated by Fleet Type = 2

3.6.2. Experimental Design

Each of the experiments uses one or more of the following performance measures: observed transportation costs, optimality gap, and the required computation time. The transportation costs are recorded to understand how the each problem variant impacts company operational costs. The optimality gap provides insight into how close the best observed solution is to the LP Relaxation (denoted by LP^*). The optimality gap is calculated using the formula,

$$Optimality\ Gap = \frac{Best\ Observed\ Solution - \ LP^*}{Best\ Observed\ Solution} \qquad (3.20)$$

Thus, the optimality gap is a non-negative proportion between [0, 1] with a value of 0 indicating that the optimal solution for the problem is obtained. When the solution obtained by a commercial solver matches the best possible solution, the optimality gap is zero. The computation time indicates the required runtime to obtain a solution. This computation time reaches a pre-determined upper-bound unless the optimal solution is obtained. These experiments are conducted with each of the first 6 scenarios permitting a computation time of 15,000 seconds and test scenarios 7 – 12 permitting a computation time of only 5,000 seconds. The difference in computation times is due to the computer memory requirements for larger problems. The details of each experiment are discussed below.

3.6.2.1. *Experiment #1: A Comparison of Different Problem Generalizations*. The PVRPB, PVRPBTW, and HPVRPB are tested on all twelve test set scenarios. The output for each scenario, including the total transportation costs, optimality gap, and computation time are both recorded.

3.6.2.2. *Experiment #2: Impact of Tour Limitation Constraints on Different Problem Generalizations*. The tour limitation constraints are applied to the PVRPB, PVRPBTW, and HPVRPB and evaluated for all twelve test scenarios. In each scenario, *MaxLH* =1 and *MaxBH* = 5. Results from these runs are compared to those obtained without these limitation constraints (see experiment #2) to observe the tradeoffs between solution quality and computational efficiency. Solution quality is measured by comparing the optimality gaps obtained when using the tour limitation constraints to those of the original formulation. This measurement, denoted as *Optimality Gap (TLC)*, is given through the formula,

$$Optimality\ Gap\ (TLC) = \frac{Best\ Observed\ (TLC) - \ LP^*}{Best\ Observed(TLC)} \tag{3.21}$$

The computational efficiency is evaluated by comparing the difference between the computation times of both the original and modified formulation which include the tour limitation constraints.

3.6.3. Equipment

All experiments are performed using the CPLEX solver for mixed integer programming problems through the GAMS interface. All experiments are run on a Pentium Dual Core 2.33 GHz Processor with 2.99 GB of RAM.

3.7. Results

3.7.1. *Experiment #1: Computational Complexity of PVRPB Extensions*

3.7.1.1. *PVRPB Results.* Table 3.4 lists the transportation costs and runtime for the PVRPB. The LP relaxation and solutions for the base model formulation without tour limitation constraints) are given by LP^* and BMF.

Table 3.4

Results Obtained when Attempting to Solve the BMF of the PVRPB

	Transportation Costs			**Runtime (Seconds)**	
Scenario	LP*	BMF	Optimality Gap	BMF	Maximum Allowed
1	566.44	566.44	0.00	0.269	15000
2	570.48	570.48	0.00	6.921	15000
3	670.54	670.54	0.00	8.929	15000
4	709.06	709.06	0.00	874.01	15000

Table 3.4 (cont'd.)

5	682.93	682.93	0.00	2025.63	15000
6	687.57	833.73	.1753	15000	15000
7	410.48	1123.67	.6347	5000	5000
8	300.67	1601.55	.8123	5000	5000
9	405.82	1771.63	.7709	5000	5000
10	nsf	nsf	n/a	5000	5000
11	nsf	nsf	n/a	5000	5000
12	nsf	nsf	n/a	5000	5000

nsf= No solution found

With the exception of Scenario 5, the total transportation cost increases with each scenario. It is believed that this scenario does not follow the same trend as others because it is one where a backhaul customer (i.e. customer 3) is further from the remaining customers. The more frequent travel to this remote location is believed to have made the problem uniquely different from both prior scenarios and Scenario 6. The computation time increases as with every scenario. This is expected as the number of variables is proportionate to the number of locations. The CPLEX compiler could only find the optimal solution for Scenarios 1 - 5. Problems with a greater number of customers could not be solved to optimality. In addition, no solution was obtained for Scenarios 10 – 12. It is important to note that our designation of not obtaining a solution is different from there being no feasible solution.

3.7.1.2. *PVRPBTW Results*. Table 3.5 summarizes the result for the PVRPB with time windows. As expressed for the PVRPB, the objective function value does not always increase with the number of customer visits, as is evident by Scenarios 5 and 6 having lower transportation costs than Scenarios 4 and 5. Furthermore, the CPLEX solver obtained the optimal solutions for Scenarios 1 – 4 and 6. No solution was found for Scenarios 10 – 12.

Table 3.5

Results Obtained when Attempting to Solve the BMF of the PVRPBTW

	Transportation Cost			**Runtime (Seconds)**	
Scenario	LP*	BMF	Optimality Gap	BMF	Maximum Allowed
1	729.29	729.29	0.00	0.269	15000
2	883.82	883.82	0.00	6.921	15000
3	994.28	994.28	0.00	8.929	15000
4	1030.06	1030.06	0.00	874.01	15000
5	842.89	988.04	.1469	2025.63	15000
6	837.03	837.03	0.00	15000	15000
7	470.03	1469.39	.6801	5000	5000
8	444.66	1630.06	.7272	5000	5000

Table 3.5 (cont'd.)

9	562.63	2247.53	.7497	5000	5000
10	nsf	nsf	n/a	5000	5000
11	nsf	nsf	n/a	5000	5000
12	nsf	nsf	n/a	5000	5000

nsf= No solution found, n/a = Not applicable

3.7.1.3. *HPVRPB Results*. Table 3.6 summarizes the result for the heterogeneous fleet PVRPB. Given the characteristics of available vehicles, the HPVRPB produced low-cost solutions. This is expected because the problem variant presents some routing options that are considerably less expensive than the homogeneous fleet-based variants. In addition, attempts to solve the HPVRPB to optimality were more successful than the PVRPB and PVRPBTW. Considering vehicle characteristics allows the costs associated with vehicle usage to be compared and allows certain routing decisions to be prioritized over others. This is likely the explanation for improvements in solution quality observed for this variant. The first five test scenarios are solved to optimality and a solution for Scenario 6 is found that is within 10% of the best possible solution. Second, the solver was able to obtain a feasible solution for Scenario 10. Neither of the other two problem variants is able to obtain a solution for this test scenario.

Table 3.6

Results Obtained when Attempting to Solve the BMF of the HPVRPB

Scenario	Transportation Costs			Runtime (Seconds)	
	LP*	BMF	Optimality Gap	BMF	Maximum Allowed
1	270.27	270.27	0.000	0.269	15000
2	274.56	274.56	0.000	6.921	15000
3	399.26	399.26	0.000	8.929	15000
4	420.04	420.04	0.000	874.01	15000
5	481.57	481.57	0.000	2025.63	15000
6	572.19	572.19	0.000	15000	15000
7	367.47	748.88	.5093	5000	5000
8	341.70	792.06	.5686	5000	5000
9	364.74	988.73	.6311	5000	5000
10	405.29	1275.76	.6823	5000	5000
11	nsf	nsf	n/a	5000	5000
12	nsf	nsf	n/a	5000	5000

nsf= No solution found, n/a = Not applicable

Figure 3.1 compares the transportation costs obtained for the PVRPB, PVRPBTW, and HPVRPB. Scenarios for which no solution is found are not included.

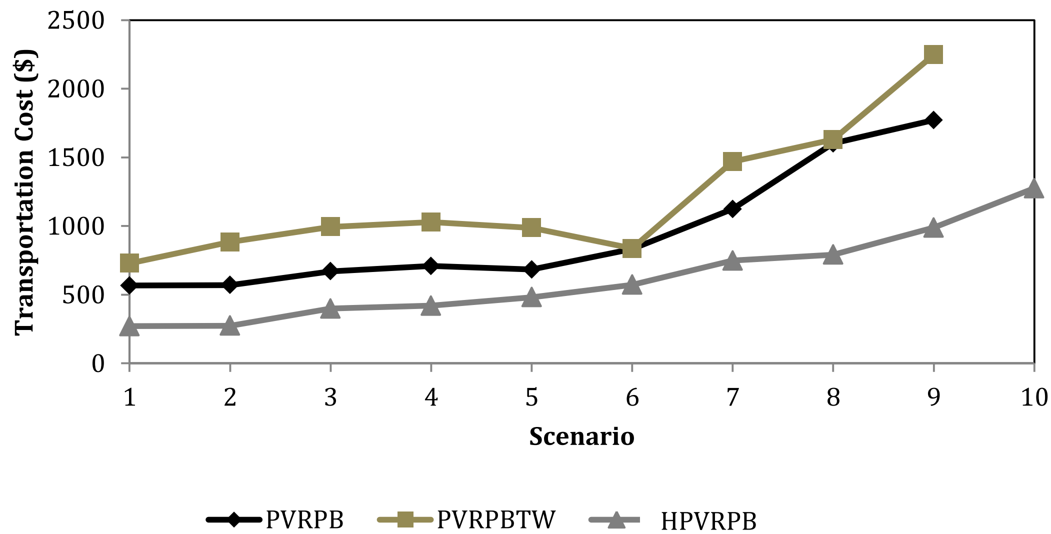


Figure 3.1. Transportation Costs Obtained for PVRPB, PVRPBTW, and HPVRPB in Test Scenarios

The HPVRPB has the lowest transportation costs across all scenarios. As stated previously, this is because certain vehicles are less expensive to use than others. Vehicles 2 – 7 are less expensive per use and more fuel efficient than any of the vehicles used for the PVRPB and PVRPBTW. The differences in transportation costs between the different problems indicate that the lower-cost vehicles were largely preferred over the higher-cost vehicle with greater tow capacity. Conversely, the transportation costs of the PVRPBTW tend to be higher than those of the PVRPB. This suggests that considering time windows can make the routing problem more expensive. Figure 3.2 expresses the solution quality of the three problem variants in terms of the optimality gap. Each of the three problem variants performs well for the first four scenarios. Only the HPVRPB obtains feasible solutions for Scenarios 5 and 6. The HPVRPB has the lowest optimality gap for all remaining scenarios. No conclusion can be made as to whether the PVRPB out performs the PVRPBTW in terms of optimality gap.

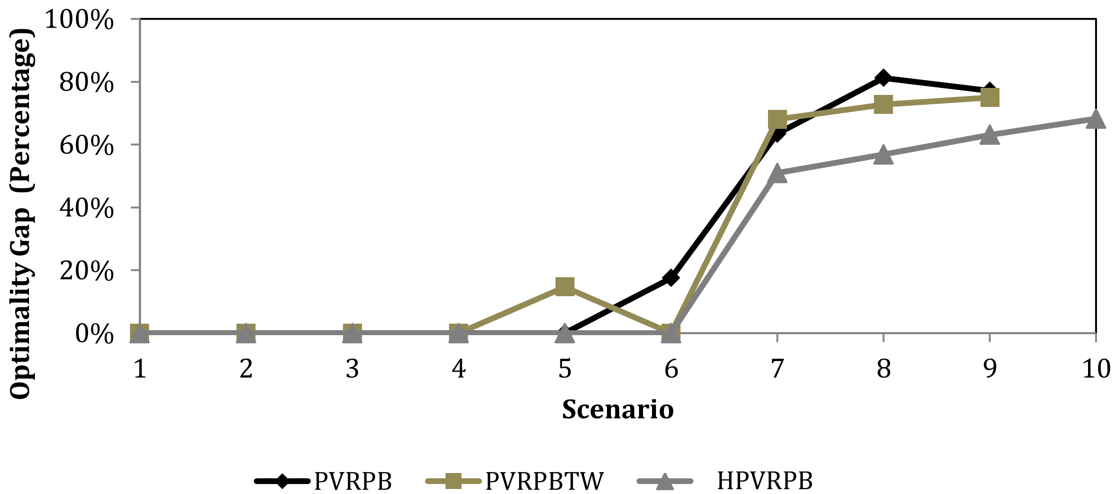


Figure 3.2. Optimality Gap Percentages for the BMF when Solving Test Scenarios for the PVRPB, PVRPBTW, and HPVRPB

3.7.2. Experiment #2: Tour Limitation Constraints

The objective function value, optimality gap percentages, and percentage of available runtime incurred for the PVRPB, PVRPBTW, and HPVRPB are given in Tables 3.7 – 3.9.

Table 3.7

Solutions Obtained when solving the BMF and TLC Formulations of the PVRPB

Scenario	**Transportation Costs**			**Optimality Gap**		**Runtime (in seconds)**	
	LP*	BMF	TLC	BMF	TLC	BMF	TLC
1	566.4	566.4	567.7	0.000	.0022	1.23	0.14
2	570.5	570.5	725.7	0.000	.2139	5.18	0.19
3	670.5	670.5	982.0	0.000	.3172	11.16	0.22
4	709.1	709.1	1,177.2	0.000	.3977	65.49	0.44
5	682.9	682.9	1,308.9	0.000	.4782	508.53	0.36
6	687.6	833.7	1,473.5	0.000	.5334	15,000	0.27
7	410.5	1,123.7	1,903.6	.5093	.7844	5,000	2.23
8	300.7	1,601.6	1,923.1	.5686	.8437	5,000	5,000
9	405.8	1,771.6	2,617.9	.6311	.8450	5,000	18.23
10	nsf	nsf	2,722.9	n/a	n/a	5,000	5,000
11	nsf	nsf	3,181.5	n/a	n/a	5,000	5,000
12	nsf	nsf	4,493.3	n/a	n/a	5,000	5,000

nsf= No solution found, n/a = Not applicable

Table 3.8

Solutions Obtained when solving the BMF and TLC Formulations of the PVRPBTW

Scenario	**Transportation Costs**			**Optimality Gap**		**Runtime (in Seconds)**	
	LP*	BMF	TLC	BMF	TLC	BMF	TLC
1	729.3	729.3	889.8	0.000	.1803	1.498	0.22
2	883.8	883.8	1,191.7	0.000	.2584	7.925	3.74
3	994.3	994.3	1,309.6	0.000	.2408	9.157	0.11
4	1030.1	1030.1	1,359.7	0.000	.2424	144.48	0.20
5	842.9	988.0	1,309.2	.1469	.3562	15,000	0.31
6	837.0	837.0	1,460.1	0.000	.4267	847.85	0.52
7	470.0	1,469.4	1,915.8	.6801	.7547	5,000	5,000
8	444.7	1,630.1	1,947.2	.7272	.7716	5,000	5,000
9	562.6	2,247.5	2,767.0	.7497	.7967	5,000	5,000
10	nsf	nsf	nfs	n/a	n/a	5,000	5,000
11	nsf	nsf	nfs	n/a	n/a	5,000	5,000
12	nsf	nsf	nfs	n/a	n./a	5,000	5000

nsf= No solution obtained, nfs = No feasible solution, n/a = Not applicable

Table 3.9

Solutions Obtained when solving the BMF and TLC Formulations of the HPVRPB

Scenario	Transportation Costs			Optimality Gap		Runtime (seconds)	
	LP*	BMF	TLC	BMF	TLC	BMF	TLC
1	270.3	270.3	270.3	0.00	0.00	0.269	0.062
2	274.6	274.6	326.6	0.00	0.1594	6.921	0.15
3	399.3	399.3	451.5	0.00	0.1157	8.929	0.15
4	420.0	420.0	524.9	0.00	0.1997	874.01	0.24
5	481.6	481.6	586.4	0.00	0.1788	2,025.6	0.50
6	572.2	572.2	677.0	0.00	0.1549	15,000	0.36
7	367.5	748.9	944.6	50.93	0.6110	5,000	6.43
8	341.7	792.1	971.9	56.86	0.6484	5,000	2,606.4
9	364.7	988.7	1,271.7	63.11	0.7132	5,000	185.00
10	405.3	1,275.8	1,361.6	68.23	0.7024	5,000	5,000
11	nsf	nsf	1,463.5	n/a	n/a	5,000	5,000
12	nsf	nsf	2,654.1	n/a	n/a	5,000	5,000

nsf= No solution found, n/a = not applicable

Solutions were obtained in each of the 12 scenarios when the tour limitation constraints were applied. The difference in optimality gap percentages between the BMF and TLC formulations also appear to decrease as system complexity increases. In terms of the computation time, only scenarios 8, 10 – 12 when solving the PVRPB, 7 – 12 when solving the PVRPBTW, and 10 – 12 when solving the HPVRPB were not solved within the available computation time (See Table 3.7).

Another important finding is that the TLC has an infeasible solution for Scenario 12. This suggests that this formulation may not find a feasible solution for systems where there are few vehicles or where trucks are expected to have high utilization. This is useful in practical application because slight modifications to customer requirements might result in there being insufficient resources (i.e. vehicle capacity, tour duration, number of vehicles, etc.) to satisfy all customers in the network.

3.8. Managerial Insights

The results of these experiments give many implications which are helpful when developing transportation schedules for networks that have characteristics of a PVRPB variant. Two aspects of this research must be discussed so that the implications are not taken out of context. First, since the experiments are conducted using the commercial software, the results expressed in this study are specifically limited to exact solution methods. Hence, differences in computing capabilities can lead to alternate solution outcomes. Despite these differences, it is expected that the comparisons made in this study are universal, regardless of the commercial solver used or computer processing capabilities. Second, the results provided are problem-specific. There can be realistic systems which have different customer requirements, vehicle capabilities, and time constraints. These results provide guidelines that while conservative, can be beneficial for realistic systems.

The experiments suggest that the PVRPB can indeed be formulated using mixed integer programming and solved to optimality for certain systems. The problem is NP complete, making each problem variant difficult to solve to optimality as the problem size grows. This complexity causes many large problems not to be solvable. In light of this limitation, one may elect to include the tour limitation constraints in the formulation. Doing so allows good solutions to be found for many instances in substantially less time. Secondly, when multiple customers express a willingness to be served at the same delivery points (see e.g. Davis et al. (2014)), many of the modeling difficulties may be avoided.

It is important to note, however, that the limitations placed on linehaul and backhaul customers could promote poor solutions if not assigned in light of system characteristics. In the scenarios presented in this research, the service times are at least 5 percent of maximum tour duration. The prescribed *MaxLH* and *MaxBH* are to some degree based on system knowledge. These assigned values are not necessarily the best upper bounds for the tour limitation constraints.

3.9. Conclusions

In closing, this research provides the first detailed study of the periodic vehicle routing problem with backhauls where (a) vehicles are not constrained and (b) there can be more than one linehaul customer per tour. This study introduces a MILP formulations for each routing problem that is NP-complete. In addition, a set of constraints which promote the identification of feasible solutions using less computation time is introduced. Two experiments are conducted to help one understand the complexity of the routing problems and identify practices that are useful when using commercial software to obtain routing schedules for realistic systems.

The model formulations introduced in this research are useful when the number of decision variables is small. Through this investigation, model formulations obtained feasible solutions for systems with 42 customers using tour limitation constraints. Nonetheless, because the tour limitation constraints assigned in this research are problem specific, they do not necessarily reflect the best solutions for the problem. Furthermore, if the maximum number of linehauls and backhauls permitted per tour is not based on a proper understanding of the network, infeasible solutions or no solution can result. Future research should provide planners with insight into what values should be assigned to these constraints. Future research will introduce more sophisticated methods for solving these routing problems.

CHAPTER 4

A Hybrid Genetic Algorithm for Solving the Periodic Vehicle Routing Problem with Backhauls

4.1. Introduction

The computational complexity associated with applying the PVRPB towards realistic, industry problems presents formidable challenges. The network may include over 100 charitable agencies and high-volume donation sites (i.e. supermarkets). Even when delivery site consolidation approaches are utilized (see Solak et al. (2012), Davis et al. (2014)), the routing problem remains very complex, resulting in over-estimated transportation costs and in some cases, undetermined vehicle routing decisions.

This research introduces an HGA designed to find good solutions for the PVRPB variants introduced in the previous chapter, particularly for systems where there are many food banks and charitable agencies served. Since the HGA solves multi-day routing problems with multiple customer types and in some cases, multiple types of vehicles, this HGA is called MULTI-HGA-ROUTE (MHR). The metaheuristic is designed based on the strategies of maintaining a diverse population of solutions and maintaining information learned in the most recent training epochs. By coupling these two components, the algorithm identifies a set of cost-effective routes for a number of days simultaneously. The metaheuristic has three unique characteristics. First, MHR is the only known metaheuristic search procedure that solves a variant of the PVRPB. The HGA is designed to ensure that linehaul precede backhauls on vehicle routes. Second, MHR is based on a permutation-based representation that allows the solution to be efficiently and effectively augmented as well as easily interpreted. The representation is beneficial for the search process, as it allows characteristics learned through the search procedure to be protected a specified number of training epochs. The results of a computational study show MHR finds good solutions for the PVRPB and PVRPBTW that are comparable and in some cases better than feasible solutions obtained using commercial solvers in a reasonable computation time. The results also show that MHR provides feasible solutions for the HPVRPB.

The remainder of the paper is organized as follows. Section 4.2 provides a review of related literature. Section 4.3 describes the characteristics of MHR. Section 4.4 describes the computational study conducted to evaluate the performance of MHR. Section 4.5 discusses calibration steps implemented for the algorithm. Section 4.6 summarizes the results of the experiments. Section 4.7 provides managerial insights obtained from the experiments that are relevant to food bank operations. Section 4.8 concludes the chapter by summarizing the findings and identifying opportunities for future research extensions.

4.2. Literature Review

The existing literature for the PVRPB is limited. Only Davis et al. (2014) consider a variation of this routing problem. This approach consists of a two-phase heuristic method which determines a delivery and collection schedule for constrained food banks vehicles that allow them to (a) deliver processed food items to remote charitable agencies and (b) collect donated food items from high-volume donors, such as supermarkets. The first phase assigns each charitable agency (i.e. linehaul customer) to a food delivery point (FDP) through an assignment model. Customers are assigned to FDPs based on vehicle capacity and their ability to access the alternative FDPs without compromising food safety. Having considered vehicle capacity in the first phase, the second phase solves a multi-period vehicle routing problem where (a) FDPs and food collection sites are visited multiple times over the planning horizon and (b) each route is permitted at most one linehaul. Thus, this component may be considered a periodic vehicle routing problem with one linehaul (i.e. PVRPB-1L). Both phases of the approach utilize commercial solvers to obtain feasible solutions.

Given the limited number of papers related to the routing problem, there are no known metaheuristic approaches that solve the PVRPB. A review of literature that highlights metaheuristic methods to solving the periodic vehicle routing problem (PVRP) and the vehicle routing problem with backhauls (VRPB) follows. Through this review, important insights which are useful in solving the PVRPB are identified. It is important to note that this review is not intended to be comprehensive. Comprehensive reviews for the PVRP and VRPB are provided by Francis et al. (2008) and Thangiah et al. (1996).

4.2.1. Metaheuristic Search Procedures for the PVRP

With the exception of problem variants that consider the impacts of different service choices (see e.g., Francis et al. (2006a), (2006b)), a specified number of visits is determined for each customer and maintained in every solution considered. Given one or more initial solutions, alternative solutions are considered where customers are served on alternate days, through alternate vehicle routes, and in a different order on the same vehicle routes. A unique feature of metaheuristics is that they allow the exploration of solutions after a local optimal solution is obtained (Laporte 2007).

The metaheuristic approaches to solving the PVRP can be categorized as either local search, population-based search methods, or hybrid search methods. Table 4.1 lists contributions that have introduced different metaheuristic search methods for the PVRP. Among these methods are memoryless heuristics (Unsp.), Tabu search (Tabu), genetic algorithms (GA), ant colony optimization (ACO), variable neighborhood search (VNS), scatter search (SS), and hybrid genetic algorithms. Local search methods include Unsp., Tabu, and VNS. Population-based methods include GA, SS, and ACO. HGA presents a hybrid approach that incorporates both local search and population-based methods.

Table 4.1

Metaheuristics Methods to Solving the PVRP

	Metaheuristic Method						
Author(s)	Unsp.	Tabu	VNS	GA	ACO	SS	HGA
Chao et al. (1995)	●						
Cordeau et al. (1997)		●					
Drummond et al. (2001)				●			
Matos and Oliveira (2004)					●		
Alegre et al. (2007)						●	
Pourghaderi et al. (2008)	●						
Hemmelmayr et al. (2009b)			●				
Vidal et al. (2012)							●
Nguyen et al. (2014)							●

The vast majority of heuristics implemented are single-point, single neighborhood local search methods. Chao et al. (1995) introduced one of the first heuristic methods for the PVRP that is structured to converge on a local optimal solution. This search method generates an initial solution by assigning customers to visit schedules while simultaneously minimizing the maximum load assigned to a vehicle. In the subsequent improvement process, better solutions are found by moving customers to other tours by changing their visitation schedule. Customers are identified based on the greatest offenders to feasibility. A modified Clarke and

Wright algorithm (Golden et al. 1977) is used to schedule new route assignments for the affected days. The improvement process also utilizes exchange heuristics that allow customers to be reassigned to other possible routes that do not change their delivery schedule.

Cordeau et al. (1997) present a tabu search heuristic for solving the PVRP and other more complex routing problems, including a multi-depot vehicle routing problem (MDVRP) and periodic traveling salesman problem. Unlike its predecessor, this metaheuristic incorporates some degree of exploration, thereby permitting it to escape local optimal solutions. The relative success of this algorithm led to its application for a number of problem variants and industry applications such as those with multiple depots (Hadjiconstantinou and Baldacci (1998)), intermediate capacity replenishment (Angelelli and Speranza (2002), Alonso et al. (2008)), and customer-specific time windows (see e.g., Cordeau et al. (2004)). Variable neighborhood search methods for solving the PVRP are introduced by Hemmelmayr et al. (2009a) and Pirkwieser and Raidl (2009).

Contributions utilizing global-based search procedures to solve the PVRP are limited. Drummond et al. (2001) utilize GA to solve the PVRP problem using a two-vector representation that indicates vehicle schedule and cumulative collection amount as a result of sequencing. Much of the processing capability of this approach is based on its use of a parallel computing infrastructure to manage smaller subpopulations on different processors. Each subpopulation manages pre-determined customer schedules.

Population-based search methods are very effective covering different aspects of the solution space. They are also limited in terms of their ability to compare observed points to neighboring solutions. This is supported in related literature. Zäphel and Bögl (2008) for example, found that genetic algorithm-inspired search are inefficient when compared to tabu search variations. Alegre et al. (2007) propose a scatter search procedure for the PVRP designed to solve problems with a large number of periods. It is the first single processor based global search method for the PVRP that outperformed pre-existing local search methods.

4.2.2. Hybrid Genetic Algorithms for Routing Problems

Using hybrid genetic algorithms (HGA) to solve combinatorial optimization problems can often outperform GAs. HGA incorporates neighborhood-based local search as a reproduction operator. This allows some of the learning that was destroyed through crossover and mutation operators to be recovered (Wang and Wu 2004). If diversity is not properly managed, the local search components of HGAs can diminish population diversity after a number of training epochs (Merz and Katayama 2004).

The effectiveness of HGAs is supported by its success towards other problem variants including the CVRP (Braysy et al. 2004) and the multi-depot vehicle routing problem with backhauls (MDVRPB). Wang et al. (2009) and Chunyu et al. (2009) also use an HGA to solve a variant of the MDVRPB with heterogeneous vehicle types.

There are several recent publications that use HGAs for the PVRP. Vidal et al. (2012) present an HGA that can be used to solve both the PVRP and the MDPVRP. Solutions are expressed using a three-vector representation. The vectors define (a) the visit schedules for each customer, (b) the depot or origin for vehicles serving each customer, and (c) a sequential ordering of customers visited on each day without the depot serving as a delimiter. The algorithm incorporates selection, crossover, education, and replacement operators. Selection identifies parents according to a uniform distribution. The crossover operator is applied to the third chromosome by utilizes a splitting algorithm discussed by Prins et al. (2004). This algorithm assigns all sequences pertaining to a specific day (and depot) from each parent and probabilistically assigns route segments for each of the remaining days. The education operator repairs infeasible solutions by solving VRP problems for each day. Reproduction uses an elitist strategy to penalize solutions that have the same customer visit schedule and customer assignment to depots. In future work, Vidal et al. (2014) demonstrate that the algorithm finds the optimal solution for test instances for 29 of the most common routing problem variants without loss of generality.

Nguyen et al. (2014) solve the PVRPTW using a two-vector representation. The first vector indicates the visit schedules for each customer; the second vector identifies the concatenated sequence in which customers are served. Their algorithm uses selection, crossover, mutation, education, and replacement operators. A roulette wheel selection operator is utilized to create the mating pool.

The crossover operator creates two offspring by choosing new visit patterns for customers followed by assigning customers to new routes. Violations to customer visit requirements are corrected based on which options results in the maximum cost increase. A combination of the unified tabu search of Cordeau et al. (2004) and random variable neighborhood of Pirkweiser and Raidl (2009) are used to perform customer visit schedule improvements. Each pattern improvement is followed by route improvement procedures.

4.2.3. Limitations of Existing Metaheuristics for Research Problem

The review identifies numerous characteristics that are relevant to this research. First, it verifies that there is a need to create a metaheuristic approach that can be used to solve the PVRPB. In addition, this review suggests that HGAs are the current state-of-the-art in terms of advanced search procedures for routing problems. New methods should incorporate them when possible. A third, indirectly related finding is that metaheuristics created to solve the PVRP routing problems may be appropriate for solving a variety of routing problem generalizations without loss of generality. This is important to this research because food banks can experience a number of unique system requirements when scheduling vehicles for food delivery and collection. A more universal heuristic search method would empower personnel to make cost-effective transportation schedules regardless of vehicle characteristics or customer requirements.

4.3. MHR Characteristics

The process flow for MHR is provided in Figure 4.1. All solid lines represent sequential transitions between operators. Dotted lines represent the storing of information about individuals in the population, mating pool, and offspring as well as their consideration in HGA operators.

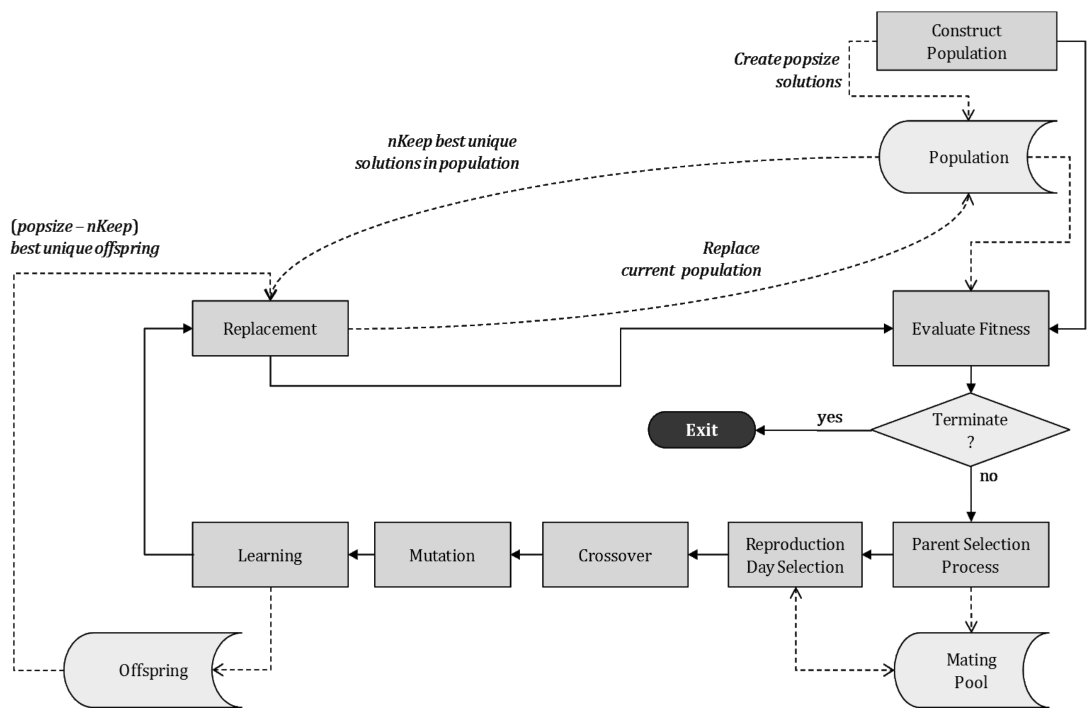


Figure 4.1. Process Flow for MHR

MHR is executed as follows. A population of initial solutions is constructed. Each individual (i.e. solution) represents a combination of vehicle tours which collectively satisfy customer demand without violating tour requirements. Tour requirements pertaining to this model are that (a) the depot is the first location involved of a vehicle route, (b) linehaul customers precede backhaul customers when both are served on the same route, (c) vehicle routes do not exceed maximum tow capacity or maximum tour duration, and (d) vehicles can only serve customers between pre-determined time intervals. Each individual is evaluated to determine its overall fitness with respect to both the objective function and compliance to system constraints. This measurement is used as the basis for a selection process which picks individuals to be the *parents.* Parents are placed in a mating pool where they are paired to create two *offspring.* Offspring characteristics are based on three reproduction operators: crossover, mutation, and learning. The reproduction operators are preceded by referring to a tabu list to identify which characteristics are targeted for change. The list identifies scheduling days that have been targeted by prior generations of each parent. Based on the list, a scheduling day is targeted and reproduction operators are implemented. Crossover creates the initial offspring by exchanging characteristics of the parents on the targeted reproduction day followed by correcting any violations resulting from a) a change in the number of times a customer is visited or b) the relative order in which the depot, linehaul customers, and backhaul customers are scheduled on a route after crossover. Mutation involves a recombination of customers assigned to a single route in a manner that does not violate tour characteristics. Learning executes local search procedures to improve the vehicle routes on the targeted day. After all offspring are created, a replacement operator picks individuals for the population of a new generation. This iterative process continues until a termination condition is reached. At termination, the individual with the best fitness reflects the algorithm approximation for the optimal solution.

4.3.1. Solution Space

Let $\boldsymbol{R}(Y)$represent a set of vehicle tours making up solution which spans a multiple day planning horizon. Each route $r \in \boldsymbol{R}(Y)$ starts at the depot, visits a sequence of $k_1$linehaul customers $\varrho_1^r, \varrho_2^r, \dots \varrho_{k_1}^r$followed by a sequence of $k_2$backhaul customers $\delta_1^r, \delta_2^r, \dots \delta_{k_2}^r$. It should be noted that k_1 and k_2 are route-specific. However, for simplicity, the subscript r is not included. ϱ_i^r denotes the *i*th customer visited in route r and corresponds to a specific linehaul customer in the set $\boldsymbol{N_L}$. Similarly, δ_i^r denotes the *i*th customer visited in route r and corresponds to a specific backhaul customer in the set $\boldsymbol{N_B}$. After visiting all customers, the vehicle returns to the same depot. The depot is represented by ϱ_0^rat the start of any tour where $k_1 \geq 1$ and δ_0^r at the beginning tours where $k_1 = 0$. The return to the depot is represented by $\delta_{k_2+1}^r$where $k_2 \geq 1$ and $\varrho_{k_1+1}^r$ when $k_2 = 0$. Vehicle routes are characterized by cumulative linehaul tow amounts (4.1), cumulative backhaul tow amounts (4.2), a total driving time (4.3), and a total duration (4.4). The notation is defined in Table 4.2.

Table 4.2

Notation for Solution Space

Symbol	Definition
q_i	Amount collected from or delivered to location i
c_{ij}	Total driving time between locations i and location j
τ_i	Service time at location i
T	Maximum tour duration
Q_V	Maximum vehicle tow capacity
$Q_L(r)$	Total delivered to linehaul customers for a given route r
$Q_B(r)$	Total collected from backhaul customers for a given route r
$C(r)$	Total distance for route r
$\mathrm{T}(r)$	Total tour duration for route r

$$Q_L(r) = \begin{cases} \sum_{j=1}^{k_1} q_{\varrho_j^r}, & \text{if } k_1 \geq 1, \quad (4.1a) \\ 0, & \text{otherwise.} \quad (4.1b) \end{cases}$$

$$Q_{\boldsymbol{B}}(r) = \begin{cases} \sum_{j=1}^{k_2} q_{\delta_j^r}, & \text{if } k_2 \geq 1 \quad (4.2a) \\ 0, & \text{otherwise.} \quad (4.2b) \end{cases}$$

$$C(r) = \begin{cases} \sum_{j=0}^{k_1} c_{\varrho_j^r, \varrho_{j+1}^r}, & \text{if } k_1 \geq 1 \text{ and } k_2 = 0 \quad (4.3a) \\ \sum_{j=0}^{k_2} c_{\delta_j^r, \delta_{j+1}^r}, & \text{if } k_1 = 0 \text{ and } k_2 \geq 1 \quad (4.3b) \\ \sum_{j=0}^{k_1-1} c_{\varrho_j^r, \varrho_{j+1}^r} + c_{\varrho_{k_1}^r, \delta_1^r} + \sum_{j=1}^{k_2} c_{\delta_j^r, \delta_{j+1}^r}, & \text{if } k_1 \geq 1 \text{ and } k_2 \geq 1 \quad (4.3c) \end{cases}$$

$$\mathrm{T}(r) = \begin{cases} c(r) + \sum_{j=1}^{k_1} \tau_{\varrho_j^r}, & \text{if } k_1 \geq 1 \text{ and } k_2 = 0 \quad (4.4a) \\ c(r) + \sum_{j=1}^{k_2} \tau_{\delta_j^r}, & \text{if } k_1 = 0 \text{ and } k_2 \geq 1 \quad (4.4b) \\ c(r) + \sum_{j=1}^{k_1} \tau_{\varrho_j^r} + \sum_{j=1}^{k_2} \tau_{\delta_j^r}, & \text{if } k_1 \geq 1 \text{ and } k_2 \geq 1 \quad (4.4c) \end{cases}$$

Equations (4.2a), (4.3a), and (4.4a) pertain to instances where only linehaul customers served through tour *r.* Equations (4.2b), (4.3b), and (4.4b) pertain to instances where there are only backhaul customers served through tour *r*. When both linehaul and backhaul customers are served through route *r,* equations (4.2c), (4.3c), and (4.4c) apply.

4.3.2. Representation

Each solution $\boldsymbol{Y}$ is composed of selected and unselected scheduling options. The options indicate (a) whether or not a specific vehicle is used on a specific day of the planning horizon, (b) whether or not a customer is served on a specific day of the planning horizon, (c) which vehicle is assigned to customers served on a specific day of the planning horizon, and (d) the sequence in which customers assigned to a specific vehicle on a specific day of the planning horizon are served. A solution $\boldsymbol{Y}$ is represented by a chromosome comprised of three equally-sized vectors. The length of each chromosome is given by Equation (4.5) where $\boldsymbol{D}$ is the set of days in the planning horizon. $\boldsymbol{N}$ is the set of customers, and $\boldsymbol{V}$ is the set of all vehicles.

$$sl = |\boldsymbol{D}| \times (|\boldsymbol{N}| + |\boldsymbol{V}|) \qquad (4.5)$$

The relative position of each allele is given by $i = \{1,2,\dots,sl\}$. The first vector, $\boldsymbol{X_1}$ is a permutation-based representation of all scheduling options, each of which is represented allele $\alpha_{1,i}$. Each allele in the second vector $\boldsymbol{X_2}$ is represented by $\alpha_{2,i}$. The second vector, uses a limited number representation to identify allele $\alpha_{1,i}$ as a selected customer or vehicle option ($\alpha_{2,i}$= 1), unselected vehicle departure ($\alpha_{2,i}$ =2), or unselected customer visit ($\alpha_{2,i}$ =3). The third chromosome, $\boldsymbol{X_3}$, uses a limited, natural number representation where each allele $\alpha_{3,i}$ identifies vehicle assignments prescribed for $\alpha_{1,i}$. While a value is prescribed for in each allele, the value of $\alpha_{3,i}$is negligible when $\alpha_{2,i} \neq 1$. Further detail into the representation follows.

The location indicated by $\alpha_{1,i}$ is denoted by $n(\alpha_{1,i})$. The associated location is determined using equation (4.6).

$$n(\alpha_{1,i}) = \begin{cases} 0, & \text{if } \alpha_{1,i} > |\boldsymbol{N}| \times |\boldsymbol{D}| & \text{(4.6a)} \\ mod(\alpha_{1,i}, |\boldsymbol{N}|), & \text{if } \alpha_{1,i} \leq |\boldsymbol{N}| \times |\boldsymbol{D}| \text{and } mod(\alpha_{1,i}, |\boldsymbol{N}|) > 0 & \text{(4.6b)} \\ |\boldsymbol{N}|, & \text{otherwise} & \text{(4.6c)} \end{cases}$$

Vehicle departures are indicated by (4.6a) whereas customer visits are given by equations (4.6b) and (4.6c), respectively. The day associated with $\alpha_{1,i}$ is defined by $p\alpha_{1,i}$. The associated departure day is given by equation (4.6). Condition (4.7a) expresses the day associated with a customer visits option, and condition (4.7b) expresses the day associated with the usage of a specific vehicle.

$$p(\alpha_{1,i}) = \begin{cases} \lceil \alpha_{1,i}/|\boldsymbol{N}| \rceil, & \text{if } \alpha_{1,i} \leq |\boldsymbol{N}| \times |\boldsymbol{D}| & \text{(4.7a)} \\ \lceil (\alpha_{1,i} - |\boldsymbol{N}| \times |\boldsymbol{D}|)/|\boldsymbol{V}| \rceil & \text{if } \alpha_{1,i} > |\boldsymbol{N}| \times |\boldsymbol{D}| & \text{(4.7b)} \end{cases}$$

When $n(\alpha_{1,i}) = 0$, the vehicle that performs the move is given by equation (4.8). When $n(\alpha_{1,i}) \in \boldsymbol{N}$, customers are initially assigned to vehicles according to the route construction algorithm (see Section 4.3.4.). These initial assignments are modified throughout the execution of the metaheuristic.

$$\alpha_{3,i} = \begin{cases} mod(\alpha_{1,i}, |\boldsymbol{V}|), & \text{if } n(\alpha_{1,i}) = 0,\ mod(\alpha_{1,i}, |\boldsymbol{V}|) > 0 \quad \text{(4.8a)} \\ |\boldsymbol{V}|, & \text{if } n(\alpha_{1,i}) = 0,\ mod(\alpha_{1,i}, |\boldsymbol{V}|) = 0 \quad \text{(4.8b)} \end{cases}$$

A gene corresponds to the three alleles $(\alpha_{1,i}, \alpha_{2,i}, \alpha_{3,i})$ in the same relative position. Genes that represent selected customer visit options are indicated by $n(\alpha_{1,i}) \in \boldsymbol{N}$, and $\alpha_{2,i} = 1$. Genes that represent unselected routing options are indicated when $\alpha_{2,i} \neq 1$. Figure 4.2 provides two examples of individuals that could be created for a routing problem with $|\boldsymbol{N}| = 4, |\boldsymbol{D}| = 3$, and $|\boldsymbol{V}| = 2$ with $\boldsymbol{N_L} = \{1,2\}$ and $\boldsymbol{N_B} = \{3,4\}$. Given 4 customers, 2 days, and 3 vehicles, $\alpha_{1,i}$ takes on integer values in the range [1,18]. The values $\alpha_{1,i} \in \{1,2, \dots ,12\}$ indicate customer visits on a specific day where $i + 4t$ represents a visit to customer i on day $t + 1$ and t is in the range $[0, |\boldsymbol{D}| - 1]$. Customer visits and the departure of a vehicle from the depot are defined according to equation (4.6). The values $\alpha_{1,i} \in \{13,14, \dots ,18\}$ indicate vehicle departures from the depot. The specific vehicle leaving the depot is defined according to equation (4.8). The specific day associated with the vehicle departure and the day in which customers are visited are defined by equation (4.7).

Customers 1, 2, and 3 require two visits and customer 4 requires three visits. Customer demand is fully satisfied over the planning horizon. Let r_1 and r_2 to express vehicle routes assigned to the first and second vehicle. Figure (2a) defines a solution where r_1 = [0-1-3-0] and r_2 = [0-2-4-0] on day 1; r_1 = [0-4-0] and r_2 = [ ] on day 2; and r_1 = [0-1-2-3-4-0] and r_2 = [ ] on day 3. The representation for the individual in Figure (2b) defines a solution where r_1 = [0-1-0] and r_2 = [0-2-3-4-0] on day 1; r_1 = [0-4-2-0] and r_2 = [ ] on day 2; and r_1 = [0-1-0] and r_2 = [0-3-4-0] on day 3. Note that vehicle assignments are based on $\alpha_{3,i}$, not precedence.

	i:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
	X_1	**13**	**1**	**14**	**2**	**3**	**4**	**15**	**8**	**5**	**6**	**7**	**16**	**17**	**9**	**10**	**11**	**12**	**18**
$\mathbf{P_1}$	X_2	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**3**	**3**	**3**	**2**	**1**	**1**	**1**	**1**	**1**	**2**
	X_3	**1**	**1**	**2**	**2**	**1**	**2**	**1**	**1**	**1**	**1**	**1**	**2**	**1**	**1**	**1**	**1**	**1**	**2**

(a)

	X_1	**13**	**1**	**14**	**2**	**3**	**4**	**15**	**6**	**8**	**5**	**7**	**16**	**17**	**9**	**10**	**11**	**12**	**18**
$\mathbf{P_2}$	X_2	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**3**	**3**	**2**	**1**	**1**	**3**	**1**	**1**	**2**
	X_3	**1**	**1**	**2**	**2**	**2**	**2**	**1**	**1**	**1**	**1**	**1**	**2**	**1**	**1**	**1**	**1**	**1**	**2**

(b)

Figure 4.2. The Representations for the (a) First and (b) Second Individual

4.3.3. Evaluation of Individuals

Route r is subject to fixed costs f resulting from vehicle use, and variable costs h and g as a result of fuel consumption and utilizing vehicle refrigeration capabilities. The resulting transportation cost attributed to route r is given by equation (4.7).

$$Z(r) = f + g \times C(r) + h \times \mathrm{T}(r) \tag{4.7}$$

Let ω_1, ω_2, and ω_3 represent the penalty charges for exceeding vehicle capacity, tour duration, and customer-specified time windows. The departure time for linehaul customer location ϱ_j in route r is defined as $\lambda_{\varrho_j^r}$. Similarly, the departure time for backhaul customer location δ_j in route r is defined as$\lambda_{\delta_j^r}$. The latest departure time for a specified linehaul (backhaul) customer is given by $l_{\varrho_j^r}\left(l_{\delta_j^r}\right)$. The penalized cost of route r is defined as the transportation cost plus a weighted penalized sum of the total number of units exceeding prescribed system requirements as given in equation (4.9).

$$\begin{aligned}\phi(r) = Z(r) + \omega_1 max\{0, \tau(r) - T\} + \omega_2 max\{0, Q_B(r) - Q_V\} \\ +\omega_2 max\{0, Q_L(r) - Q_V\} \\ +\omega_3\left[\sum_{j=1}^{k_1} max\left\{0, \lambda_{\varrho_j^r} - l_{\varrho_j}\right\}\right. \\ \left. + \sum_{j=1}^{k_2} max\left\{0, \lambda_{\delta_j^r} - l_{\delta_j}\right\}\right]\end{aligned} \tag{4.9}$$

An individual consists of multiple routes during the planning horizon and is evaluated using a total penalized cost (Equation 4.10).

$$\Phi(\boldsymbol{Y}) = \sum_{r \in R(\boldsymbol{Y})} \phi(r) \tag{4.10}$$

Penalty cost coefficients $(\omega_1, \omega_2, \omega_3)$ are updated each epoch according to weighted penalty charges as expressed by Nguyen et al. (2014). Let $\bar{O}$, $\bar{M}$,and $\bar{W}$ represent average number of violations to vehicle capacity, tour duration, and time window constraints observed for the population. The number of violations is based on the standard metric used to reflect distance and time. Ω represents an extremely large penalty. This penalty should be large enough to make every solution for which there is a violation inferior to those where there is not a violation. The updated penalty coefficients are given by the equations (4.11), (4.12), and (4.13).

$$\omega_1 = \begin{cases} \Omega\left(\dfrac{\bar{O}}{\bar{O} + \bar{M} + \bar{W}}\right), & if\ \bar{O} + \bar{M} + \bar{W} > 0, \quad (4.11a) \\ 0, & \text{otherwise.} \quad (4.11b) \end{cases}$$

$$\omega_2 = \begin{cases} \Omega\left(\dfrac{\bar{M}}{\bar{O} + \bar{M} + \bar{W}}\right), & if\ \bar{O} + \bar{M} + \bar{W} > 0, \quad (4.12a) \\ 0, & \text{otherwise.} \quad (4.12b) \end{cases}$$

$$\omega_3 = \begin{cases} \Omega\left(\dfrac{\bar{W}}{\bar{O} + \bar{M} + \bar{W}}\right), & if\ \bar{O} + \bar{M} + \bar{W} > 0, \quad (4.13a) \\ 0, & \text{otherwise.} \quad (4.13b) \end{cases}$$

The coefficient values apply whenever there is at least one violation, as expressed by (4.11a), (4.12a), and (4.13a). When there are no violations in the population, the coefficient values are dropped.

4.3.4. Initial Population

The population is created using a new route construction algorithm which iteratively assigns zero or more linehaul and backhaul customers to each vehicle. The algorithm is initialized by generating three random vectors identifying a sequence of possible vehicle departures from the depot, linehaul visit, and backhaul visit options. A counter is also initialized to track all customers visits assigned. After initialization, the algorithm iteratively evaluates each vehicle departure option to determine if unsatisfied customers can be served and whether to assign linehaul and/or backhaul customers. When the evaluated vehicle is the last one available for a specific day, every customer who (a) has not been assigned for that day and (b) whose total number of required visits has not been satisfied is assigned. If the vehicle is not the last one available for that day, it is assigned a tour containing all linehauls, all backhauls, or linehauls followed by backhauls. The maximum number of linehauls and backhauls assigned to each of the constructed routes is randomly selected. Linehaul and backhaul customers are assigned to vehicles in the order that they are arranged during initialization. All linehaul and backhaul visit options which are not assigned to a vehicle are prescribed as unselected customer visits. This algorithm is repeated iteratively based on the size of the population. The pseudocode for this process is given in Appendix A.2.

4.3.5. Mating Selection Reproduction Day Targeting

4.3.5.1. *Mating Pool Selection*. The selection operator is designed to choose individuals within the population with high-fitness for mating purposes. A mating pool is identified through the fitness proportionate selection operator. This operator performs a biased selection from the current population. For the purposes of minimizing the similarity of solutions in the mating pool, this research explores a variation of the strategy presented by Vidal et al. (2012). Their approach penalizes the weighted fitness cost of individuals in proportion to the number of alternate solutions within the population for which customers are served through the same underlying visit schedules. Accordingly, the biased weighted penalty cost is given by Equation 4.14.

$$\Phi_{\mathrm{B}}(\mathbf{Y}) = \frac{\Phi(\mathbf{Y})}{\eta} \tag{4.14}$$

The number of individuals in a population which have a specific customer visit schedule is represented by η. The selection consists of *popsize* picks with replacement.

4.3.5.2. *Reproduction Day Targeting Operator.* Reproduction is targeted toward genes prescribed for a selected day $p \in \boldsymbol{D}$. MHR uses a tabu list to promote equal treatment of all genes. The list records the crossover option used for the past Ψ generations. The list size is based on equation (4.15).

$$\Psi = \begin{cases} \lfloor |\boldsymbol{D}|/2 \rfloor, & \text{if } |\boldsymbol{D}| \text{ is an odd number} \quad (4.15\text{a}) \\ \lfloor |\boldsymbol{D}|/2 \rfloor - 1, & \text{if } |\boldsymbol{D}| \text{ is an even number} \quad (4.15\text{b}) \end{cases}$$

An example of the reproduction day targeting operator is given in Table 4.3. Given a system with a 5-day planning horizon, the targeted reproduction day for two parents, P_1 and P_2, is selected as follows. First, the union of the tabu lists of the two parents is identified. In this problem, this combined list is the set {3,4,5}. Next, the set of days in the planning horizon which are not in the combined list are identified. These non-tabu days serve as the candidate reproduction day targets. The reproduction day target is selected according to a uniform distribution. This target is temporarily stored as the “selection” which directs crossover and mutation operators. At the completion of all reproduction operators, the target enters the tabu list. Days in the tabu list are treated on a first-in/first-out basis.

Table 4.3

Example of Tabu List Selection

Parent	Tabu List	Possible Target	Selected Day	Updated Tabu List (Next Epoch)
P_1	{5,4}	{1,2}	2	{4,2}
P_2	{3,5}			{5,2}

4.3.6. Crossover Operator

The crossover operator creates two offspring θ_1 and θ_2 which are based on the features of two parents P_1 and P_2 selected from the mating pool. Each offspring matches the genes of one of the parents for scheduling options occurring on the targeted reproduction day. Scheduling options not on the targeted day are based on the other parent. Using the targeted reproduction day to define the crossover region, alleles are exchanged between the two parents. This process is followed by a correction procedure that maintains the prescribed customer visit frequency. More specifically, when the exchange results in too many visits for a specific customer, an option is removed. If an exchange results in too few visits for a specific customer, an option is added. This process is summarized in the pseudocode in Appendix A.3.

Figure 4.3 gives a visual example of how the crossover operator works. Referring to the description provided for the representation (see Section 4.2.3), and assuming that the tabu list = {1}, and the targeted reproduction day is {2}, two offspring $\theta_1$and $\theta_2$are created from parents P_1 and P_2. The figure expresses that after exchanging genes on the targeted reproduction day, the total number of times that customer 2 is visited in $\theta_1$increases to 3. This is corrected by removing the customer visit on day 3. After crossover, the number of times customer 2 is visited in $\theta_2$decreases from 2 to 1. This is corrected by adding a visit on day 3. It is worth mentioning that since there was an unused vehicle for this scheduling day, the added customer is placed on a new vehicle route. This is an important feature of the algorithm because it prevents pre-mature vehicle removal from solutions. (See Section 4.2.3).

$\mathbf{P_1}$	X_1	13	1	14	2	3	4	15	8	5	6	7	16	17	9	10	11	12	18
	X_2	1	1	1	1	1	1	1	1	3	3	3	2	1	1	1	1	1	2
	X_3	1	1	2	2	1	2	1	1	1	1	1	2	1	1	1	1	1	2
$\mathbf{P_2}$	X_1	13	1	14	2	3	4	15	6	8	5	7	16	17	9	10	11	12	18
	X_2	1	1	1	1	1	1	1	1	1	3	3	2	1	1	3	1	1	2
	X_3	1	1	2	2	2	2	1	1	1	1	1	2	1	1	1	1	1	2

(a)

$\boldsymbol{\theta_1}$	X_1	13	1	14	2	3	4	15	8	6	5	7	16	17	9	10	11	12	18
	X_2	1	1	1	1	1	1	1	1	1	3	3	2	1	1	3	1	1	2
	X_3	1	1	2	2	2	2	1	1	1	1	1	2	1	1	1	1	1	2
$\boldsymbol{\theta_2}$	X_1	13	1	14	2	3	4	15	8	5	6	7	16	17	9	18	11	12	10
	X_2	1	1	1	1	1	1	1	1	3	3	3	2	1	1	1	1	1	1
	X_3	1	1	2	2	2	2	1	1	1	1	1	2	1	1	2	1	1	2

(b)

Figure 4.3. Example of (a) Two Parents and (b) Offspring Created by the Crossover Operator

When a customer is added to a vehicle route through the correction process, the route is resorted such that all linehauls precede backhauls. The relative order in which linehaul and backhaul customers are served is maintained.

4.3.7. Mutation Operator

In order not to duplicate a recombination that might be produced by the learning operator, MHR utilizes a modified version of the shift change mutation operator discussed in Wang and Wu (2004). It consists of reordering the sequence in which linehaul and/or backhaul customers assigned to an arbitrarily-selected vehicle route are visited in a cyclical manner, such that the last linehaul (backhaul) customer served on a tour becomes the first linehaul (backhaul), the first linehaul (backhaul) customer becomes the second, and so on. The modified shift operator is executed only on linehaul or backhaul customers served by a randomly-selected vehicle. Figure 4.4 gives a visual example of the impacts of the operator. In this example, there are two vehicles and four customers. Customers 1 and 2 are served through linehauls; customers 3 and 4 are served through backhauls. Vehicle 1 is assigned the route [0-2-3-4] and vehicle 2 is assigned the route [0-1-0]. Vehicle 1 is targeted by the mutation operator. The shift operator modifies route 1 to [0-2-4-3-0].

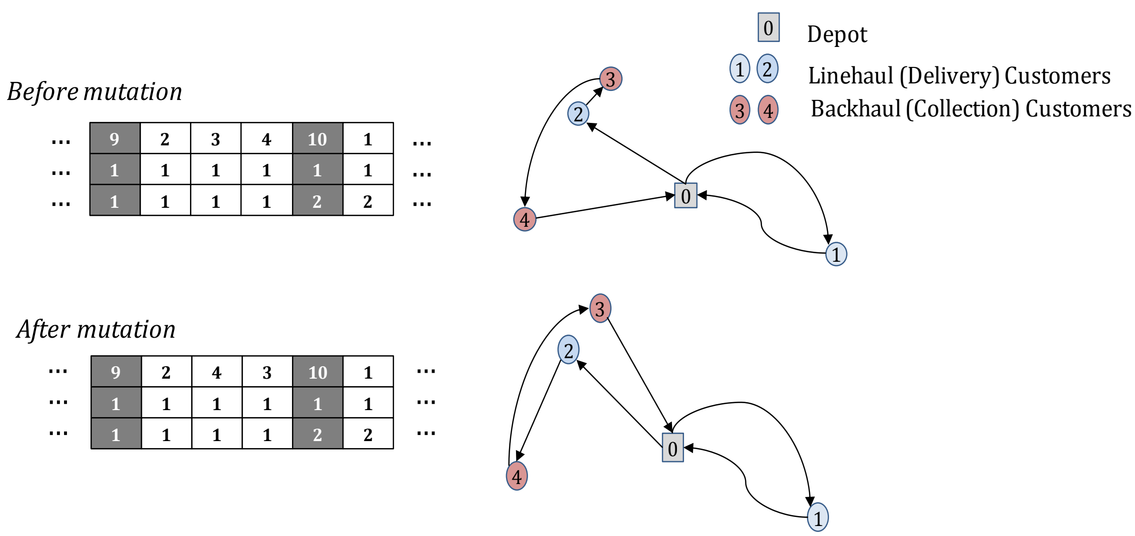
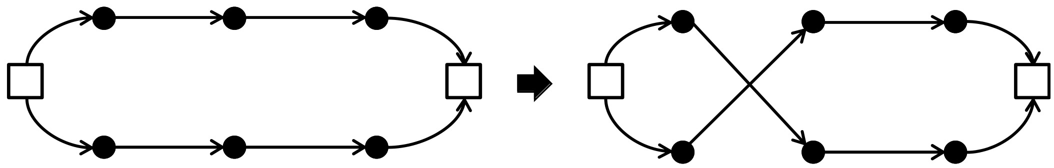


Figure 4.4. Visual Representation of Modified Shift Operator

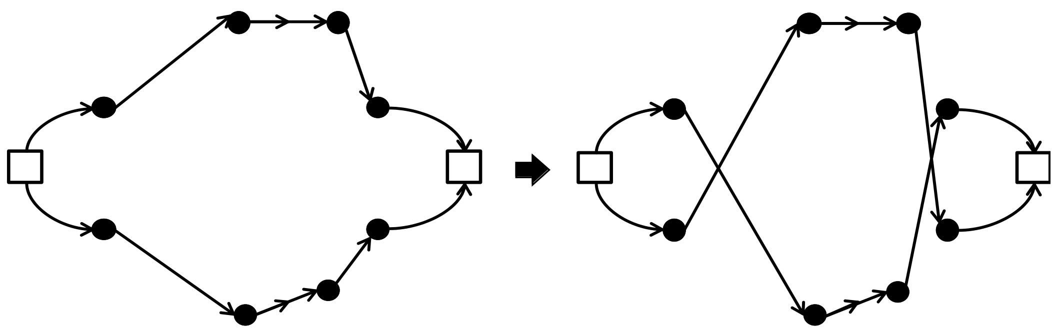
4.3.8. Learning Operator

Learning explores local search procedures which move customers from one route to another (i.e. inter-route operators) as well as those which change the order in which customers on the same route are visited (i.e. intra-route operators). Local search procedures which involve moving customer from one route to another (i.e. inter-route operators) are controlled through random variable neighborhood search (RVNS). RVNS implements each inter-route operator repeatedly until no improvement to the route results. The process is continued for another inter-route heuristic until each has been implemented. A unique feature of RVNS is that the order in which inter-route operators are implemented is randomly selected. Readers interested in an in-depth explanation of RVNS are referred to Hansen and Mladenovic (2001). Each Intra-route operator is repeated in succession until no improvement occurs from either.

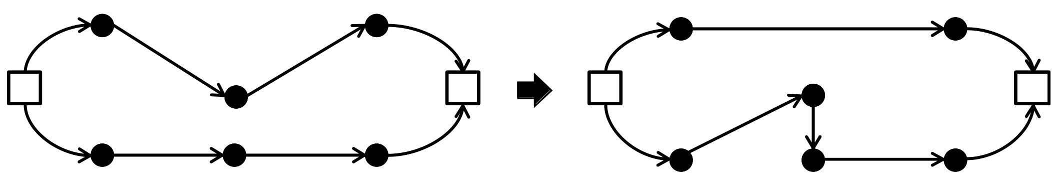
Modifications for three inter-route operators and two intra-route heuristics are created to ensure that they ensure that linehauls precede backhauls on vehicle tours. Modified inter-route operators include the Two-Opt* exchange operator discussed in Potvin and Rousseau (1995), the RELOCATE move operator introduced in Savelsbergh (1992), and the CROSS-exchange operator introduced in Tailliard et al. (1997). Visual representations for these operators are provided in Figure 4.5.



(a)



(b)

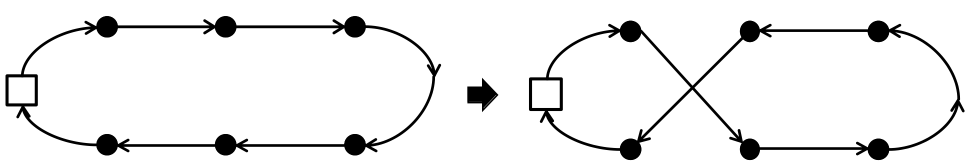


(c)

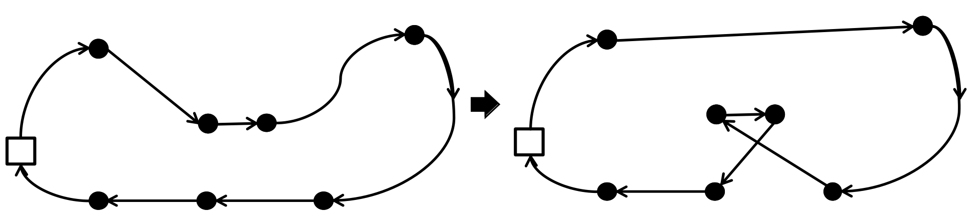
Figure 4.5. Visual Representation for (a) Two-Opt*, (b) CROSS, and (c) RELOCATE Inter-Route Operators (Braysy and Gendreau 2005)

As show in the figure, the Two-Opt* operator exchanges the vehicle travel paths for two routes that occur after randomly-selected points. The CROSS operator exchanges segments of one or more customers between two routes. The RELOCATE operator moves a route segment containing one or more customers and places them on another route. To ensure linehauls precedence on vehicle tours, modified Two-

Opt* and CROSS operators are executed between routes when the segments involved in the exchanges contain only linehaul customers or backhaul customers. The modified RELOCATE operator considers segments that contain linehauls customers, backhauls customers, or both linehauls and backhaul customers. Moved segments are limited to being inserted at points which do not violate linehaul precedence requirements. The pseudocode for the modified Two-Opt*, CROSS, and RELOCATE operators are provided in Appendices A.4.1 – A.4.3. Intra-route heuristics implemented in this study are based on the Two-Opt exchange operator of Lin (1965) and the Or-opt operator of Or (1976). Figure 4.6 gives a visual representation of both intra-route heuristics.



(a)



(b)

Figure 4.6. Visual representation of (a) Two-Opt and (b) Or-Opt Intra-Route Operators (Braysy and Gendreau 2005)

The two-opt operator alters a single tour by inverting the order of a tour between the customers immediately following the two exchanged. The Or-opt operator selects an arbitrary number of sequential customers on a tour and repositions them to occur either before or after customers who previously preceded or followed them. Modifications for the two-opt operator restricts the inversion to occurring between two arbitrarily-selected linehaul or backhaul customers assigned to the tour. Much like the modified RELOCATE operator, the modified Or-opt operator allows a sequence of one or more customers of any type to be moved and inserted at any location that would not cause the comingling of linehaul and backhaul items. The pseudocode for the modified two-opt and Or-opt are provided in Appendices A.4.4 and A.4.5.

4.3.9. Replacement Operator

The replacement operator copies $nkeep$ best unique individuals of the original population and the $(popsize - nkeep)$ best unique offspring. Unique individuals are based on the underlying customer visit schedules. Note that the first pick from both the original population and the offspring result in the individual corresponding to having the best overall penalized costs is selected. In subsequent selections, however, the pick is limited to individuals who do not have the same visits schedules for customers as a prior selection. If there are no unique alternatives, individuals are selected at random.

4.4. Calibration of MHR Settings

MHR is used run in a manner that maximizes solution diversification, intensification, and learning. For that reason, the probability of crossover, $pcross$, probability of mutation, $pmut$, and probability of learning, $plearn$, are each set to 1.0. The values assigned for $popsize$ and $numIter$ are set to 50 and 500, respectively. This decision is made to assign these values based on the results observed in Vidal et al. (2012).

The value for $nkeep$ is calibrated to prevent multiple individuals in the same population from having shared customer visit schedules. This makes it possible for each individual to represent a solution containing low-cost routes without duplication. Accordingly, $nkeep = \{.1, .2, .3, .4, .5\} \times popsize$ are considered. The smallest of these values that maximizes the number of unique customer visit schedules in each epoch is selected. The calibration is based on the observed population when solving the PVRPB for test scenario #1 (see Chapter 3 Section 3.6.)

The population only contained unique solutions in every training epoch when $nkeep$ is set to $0.50 \times popsize$. The population diversity when $nkeep = \{.1, .2, .3, .4\} \times popsize$ is given in Figure 4.5. It is important to note that $nkeep = .4 \times popsize$ also produces good diversity in many instances, as at least 41 individuals with unique customer visit schedules are maintained in a population in each epoch. When $nkeep = 0.1 \times popsize$, as few as 11 and no more than 15 individuals with unique customer visit schedules were maintained in the population in each epoch.

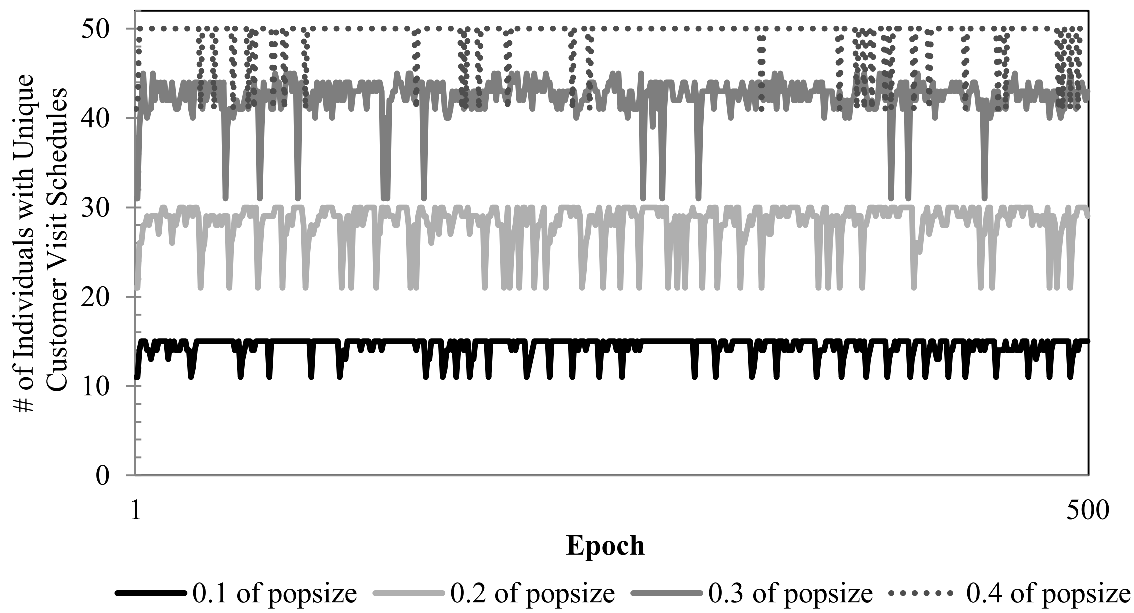


Figure 4.7. Impact of different *nkeep* Values on Diversity

4.5. Experimentation

MHR is evaluated in terms of its effectiveness and efficiency when solving each of the routing problems variants introduced in this research. This effectiveness is based on both the precision of MHR in successive runs and the accuracy of MHR in terms of finding the minimum transportation cost. The efficiency of MHR is evaluated in terms of the required computation time to complete training.

Given the average and standard deviation of transportation costs corresponding to feasible solutions for scenario z, the precision of the algorithm is calculated using the corresponding coefficient of variation CV_z. This measurement is calculated using equation 4.16.

$$CV_z = \frac{\sigma_z}{\mu_z} \tag{4.16}$$

Smaller values for CV indicate that MHR provides consistent outcomes for the routing problem. Larger values for CV indicate that the obtained solution is more the result of chance rather than its ability to learn the solution space.

The accuracy of MHR is based on the difference between its calculated transportation costs and those obtained using the BMF and TLC introduced in the previous chapter. Both model formulations are calculated using exact solution methods executed by commercial software. The global optimal solution is obtained for smaller test instances. The optimality gap between LP relaxation and the best feasible solution obtained by these model formulations increases as the problem size grows. For some very complex problems, a feasible solution is not obtained.

When a feasible solution is obtained for a scenario using the BMF, it is treated as the *best known* solution. When MHR is superior to BMF, it indicates that the metaheuristic is capable of finding solutions for the test scenario more efficiently than the commercial solver. When a solution is obtained using TLC, it represents a second feasible solution to the BMF. This second feasible solution represents the time that can be expected to find a feasible solution for the routing problem that is acceptable in terms of the number of linehaul and backhaul visits. As specified in the previous chapter, TLC evaluate solutions where vehicle tours are limited to serving no more than 1 linehaul customer and 5 backhaul customers. When no feasible solution is obtained for both BMF and TLC, any feasible solution obtained by MHR is an improvement to the *best known* solution. In addition to providing a second feasible solution, the TLC gives a lower bound for the time required to solve the routing problem to optimality using the commercial solver. If MHR requires less time to solve the TLC, it also requires less time to solve the BMF. Alternatively, if MHR requires more time to solve than TLC, it indicates that a feasible solution can be obtained using the commercial solver in less time than MHR.

A key performance measure for evaluating effectiveness is the relative gap between the transportation costs obtained through MHR and the model formulation methods. The relative gap in transportation costs when comparing MHR to BMF is calculated for a scenario z using equation 4.17.

$$vs.BMF_z = \frac{MHR_z - BMF_z}{MHR_z} \tag{4.17}$$

Similarly, the relative gap in transportation costs when comparing MHR to TLC is calculated for a scenario z using equation 4.18.

$$vs.TLC_z = \frac{MHR_z - TLC_z}{MHR_z} \tag{4.18}$$

The resulting accuracy when compared to either model is within the range $(-\infty, 1)$ with negative values indicating that MHR improves upon the best known solution for the test scenario and positive values suggesting that MHR provides inferior solutions for the test scenario. Solutions obtained for MHR that match or improve upon those of BMF are favored. If the relative gap between MHR and BMF is a small positive number, it indicates that the HGA may be preferred in practical use, depending upon the difference in computation time for both solution methods. This is especially true when the relative gap between MHR and TLC is negative. If the relative gap between MHR and TLC is positive, it indicates that MHR is an inferior solution method. When no solution is found for a scenario using BMF or TLC, a value n/a is assigned for the corresponding relative gap.

MHR efficiency is based on the computer runtime. This is the total time between the initialization and termination of a run. In this study, the minimum runtime of MHR is compared to that of TLC. This time is selected to represent the model formulations rather than both BMF and TLC for two reasons. First, TLC represents a smaller problem that is solved faster than BMF; therefore, if the minimum runtime for MHR is comparable or superior to that of TLC, it is also comparable or superior to that of BMF. Second, much like MHR, TLC represents a heuristic solution method for the overall routing problem. The comparison identifies which of the heuristic methods requires the most time to provide suggested solution. It is important to note that since both methods attempt to improve the solution over multiple iterations, the suggested solution is not obtained until the termination of the search.

Each of these experiments is run using the test scenarios introduced in Chapter 3 (see Section 3.6). MHR is run three times for each test scenario to solving a specific PVRPB variant. The BMF and TLC obtained from each test scenario are used to evaluate model accuracy. MHR is run on an Intel Core i3 CPU 2.4 GHz with 4 GB of RAM.

4.6. Results

4.6.1. MHR Precision for Problem Variants

The precision measures used to evaluate the consistency of MHR transportation cost estimates for each test scenario are provided in Tables 4.4 – 4.6.

Table 4.4

Precision Measures for Feasible MHR Solutions to the PVRPB

Scenario	Average	Std. Dev	CV
1	568.68	1.70	0.00
2	574.66	0.49	0.00
3	782.21	81.09	0.10
4	771.79	79.68	0.10
5	844.19	2.18	0.00
6	972.85	105.33	0.11
7	1,252.75	115.38	0.09
8	1,587.16	124.07	0.08
9	2,132.73	167.53	0.08
10	2,322.19	119.84	0.05
11	2,779.61	173.77	0.06
12	3,654.49	-	-

Table 4.5

Transportation Cost Measures for Feasible MHR Solutions to the PVRPBTW

Scenario	Average	Std. Dev	CV
1	733.16	-	-
2	896.28	5.59	0.01
3	1,056.18	93.79	0.09
4	1,054.63	6.72	0.01
5	1,123.59	110.69	0.10
6	922.07	97.96	0.11
7	1,598.67	92.58	0.06
8	2,166.30	222.07	0.10
9	2,582.53	76.94	0.03
10	3,036.19	164.82	0.05
11	3,795.29	97.77	0.03
12	4,870.39	90.38	0.02

Table 4.6

Transportation Cost Measures for Feasible MHR Solutions to the HPVRPB

Scenario	Average	Std. Dev	CV
1	312.88	15.54	0.05
2	352.56	24.40	0.07
3	477.80	61.05	0.13
4	594.57	48.62	0.08
5	822.30	198.05	0.24
6	809.32	38.26	0.05
7	1,071.42	37.51	0.04
8	1,219.87	34.89	0.03
9	1,427.16	169.13	0.12
10	1,489.38	-	-
11	1,705.29	-	-
12	2,396.03	-	-

The average CV across all test scenarios is 0.06 when solving the PVRPB, 0.05 when solving the PVRPBTW, and 0.07 when solving the HPVRPB. The CV is no more than 0.11 for any test scenario where MHR is used to solve the PVRPB or PVRPBTW. MHR has two test scenarios for which its CV is greater than this value when solving the HPVRPB. The largest CV is found for scenario 5. It is worth mentioning that while solving the HPVRPB, only 1 run provides a feasible solution for scenarios 10 – 12.

4.6.2. MHR Accuracy for Problem Variants

A comparison of the results obtained using MHR to model formulation-based methods of solving the PVRPB is given in Table 4.7. MHR matches or improves upon the overall results in 5 of 12 test scenarios (i.e. Scenarios 7 - 8, 10 – 12).

Table 4.7

Comparison of MHR to BMF and TLC when Solving PVRPB

	Transportation Costs			**Relative Gap (proportion)**		**Runtime (in seconds)**	
Scenario	MHR	BMF	TLC	vs. BMF	vs. TLC	MHR	TLC
1	567.70	**566.4**	567.70	.002	**.000**	216.61	0.14
2	574.38	**570.5**	725.72	.007	**-.263**	225.20	0.19
3	688.58	**670.5**	981.99	.026	**-.426**	216.16	0.22
4	725.79	**709.1**	1,177.2	.023	**-.622**	232.32	0.44
5	842.00	**682.9**	1,308.9	.189	**-.554**	231.72	0.36
6	851.24	**833.7**	1,473.5	.021	**-.731**	248.50	0.27
7	**1,122.3**	1,123.7	1,903.6	**-.000**	**-.696**	281.69	2.23
8	**1,501.8**	1,601.6	1,923.1	**-.067**	**-.281**	**326.72**	5,000
9	1,939.4	**1,771.6**	2,617.9	.087	**-.350**	333.34	18.23

Table 4.7 (cont'd.)

10	**2,185.1**	nsf	2,722.9	**n/a**	**-.246**	**291.12**	5,000
11	**2,632.4**	nsf	3,181.5	**n/a**	**-.209**	**301.39**	5,000
12	**3,654.5**	nsf	4,493.3	**n/a**	**-.230**	**309.97**	5,000

nsf = No solution found

BMF provides the best results for smaller test scenarios. This is anticipated for two reasons. First, the commercial solver finds very good solutions for the first six scenarios. It identifies the optimal solution for the first five test scenarios, and the optimality gap between the BMF and the LP relaxation is within 0.18 for scenario 6 (see Chapter 3, section 3.7). Second, while the metaheuristic is anticipated to find improved solutions, it is not expected to necessarily find the optimal solution. Reasonable solutions are found for most scenarios. In fact, there are only two test scenarios where the relative gap between the best known solution and that of MHR is more than 3 percent (i.e. Scenarios 5 and 9). This may be attributed to network characteristics. The optimal solution for scenario 5 is less than that of scenario 4. This is interesting because scenario 4 presents a problem with 8 total customers requiring a total of 11 visits while scenario 5 presents a problem with 9 total linehaul and backhaul customers, requiring a total of 13 visits. Scenario 9 presents a unique instance where the best possible routes for TLC were obtained within the allowed computation time. This is different from Scenario 8 because the best possible solution for TLC is not obtained in the allowed computation time. The transportation costs for MHR are less than or equal to those obtained using TLC in each of the 12 scenarios. For larger instances (i.e. Scenarios 10 – 12), the search procedure implemented by MHR is completed faster than TLC. The runtime for MHR is also superior to TLC in test scenario 8. This is likely the result of the TLC having to consider more alternate routes as the problem increases.

A comparison of the results obtained using MHR, BMF and TLC to solve the PVRPBTW is given in Table 4.8. Solutions for test scenarios 1 – 4, and 6 are solved to optimality using BMF. For test scenario 5, the optimality gap between the BMF and the LP Relaxation is within 0.15. It is important to note that because BMF and TLC do not consider refrigeration costs incurred when vehicles wait to be serviced (See Chapter 3 Section 3.4.1) they underestimate transportation costs. Since MHR considers these costs, it should be expected that the transportation costs for the metaheuristic would be inferior to optimal solutions obtained from BMF. Furthermore, the relative gaps between MHR and both model formulation-based method are expected to be upper bound values.

Although model formulations did not account for wait time, MHR was able to obtain solutions that are less than .035 of the relative gap for the first six test scenarios. Of the two remaining scenarios, only scenario 8 has a relative gap greater than 0.10. It is interesting to note that the transportation cost for this test scenario using MHR is substantially greater than that obtained by both BMF and TLC. This is unusual because MHR solutions are convincingly superior to those of TLC for all other test scenarios. This could suggest that (a) the test scenario presents a more complex instance for the routing problem that is more easily interpreted by exact methods and/or (b) the exclusion of refrigeration costs during vehicle wait time results in a highly-inflated relative gap. The same could be said of scenario 9, although the relative gap between MHR and TLC favors using the metaheuristic. It is also worth mentioning that these are test scenarios for which there are 20 and 25 linehaul and backhaul customers, requiring a total of 40 and 47 total visits. Given the number of customers served, it is conceivable that after accounting for customer wait time, there may be one or more additional vehicles required. MHR provides the only known solutions for test scenarios 10 – 12 despite permitting 5,000 minutes of runtime for BMF and TLC. Similar to the results for the PVRPB, MHR solves larger test scenarios for the PVRPBTW faster than TLC. The results show that the metaheuristic solves each test scenarios 7 – 12 faster than the exact formulation method.

Table 4.8

Comparison of MHR to BMF and TLC when Solving PVRPBTW

	Transportation Costs			**Relative Gap (proportion)**		**Runtime (in Seconds)**	
Scenario	MHR	BMF	TLC	vs. Base	vs. TLC	MHR	TLC
1	733.16	**729.29**	889.75	0.005	**-0.214**	224.89	0.22
2	889.82	**883.82**	1,191.7	0.007	**-0.339**	202.27	3.74

Table 4.8 (cont'd.)

3	1,002	**994.3**	1,309.6	0.008	**-0.307**	214.10	0.11
4	1,049.3	**1,030.1**	1,359.7	0.018	**-0.296**	213.89	0.20
5	998.6	**988**	1,309.2	0.011	**-0.311**	211.83	0.31
6	864.3	**837**	1,460.1	0.032	**-0.689**	232.30	0.52
7	1,494.6	**1,469.4**	1,915.8	0.017	**-0.282**	**271.9**	5,000
8	2,037.5	**1,630.1**	1,947.2	0.200	0.044	**298.2**	5,000
9	2,493.7	**2,247.5**	2,766.1	0.099	**-0.109**	**324.5**	5,000
10	**2,919.6**	nsf	nfs	**n/a**	**n/a**	**332.4**	5,000
11	**3,700.3**	nsf	nfs	**n/a**	**n/a**	**315.9**	5,000
12	**4,806.5**	nsf	nfs	**n/a**	**n/a**	**326.5**	5000

nsf = No solution found
nfs = No feasible solution

Table 4.9 shows a comparison of MHR to the model formulation-based methods when solving the HPVRPB. The results show that MHR does not perform as well on the HPVRPB as other routing problems. Despite only test scenarios 1 – 5 being solved to optimality using BMF, MHR could not match the transportation costs associated with either of the 9 test scenarios where the model formulation method obtains a feasible solution. In fact, the relative gap between MHR and BMF is less than 10 percent in only 2 scenarios. This is also the only problem variant where the relative gap between MHR and TLC is positive in most scenarios.

Table 4.9

Comparison of MHR to BMF and TLC when Solving HPVRPB

	Transportation Costs			Relative Gap (proportion)		Runtime (seconds)	
Scenario	MHR	BMF	TLC	vs. BMF	vs. TLC	MHR	TLC
1	295.5	**270.3**	270.3	0.085	0.085	185.4	0.062
2	324.6	**274.6**	326.6	0.154	**-0.006**	189.4	0.15
3	411.5	**399.3**	451.5	0.030	**-0.097**	185.1	0.15
4	540.6	**420**	524.9	0.223	0.029	203.8	0.24
5	603.9	**481.6**	586.4	0.203	0.029	181.9	0.50
6	777.9	**572.2**	677	0.264	0.130	185	0.36
7	1,042.2	**748.9**	944.6	0.281	0.094	239	6.43
8	1,195.2	**792.1**	971.9	0.337	0.187	**273.7**	2,606
9	1,307.6	**988.7**	1,271.7	0.244	0.027	**267.5**	185.00
10	1,489.4	1275.8	**1,361.6**	0.143	0.086	**286.1**	5,000
11	1,705.3	nsf	**1,463.5**	**n/a**	0.142	**300.5**	5,000
12	**2,396**	nsf	2,654.1	**n/a**	**-0.108**	**311.1**	5,000

nsf = No solution found

A likely reason for the poor performance of MHR for the problem variant is that unlike the others, the characteristics of assigned vehicles impact solution quality.

This is supported by the set of vehicle routes associated with its best solution for the HPVRPB assigning customers to vehicles with the largest capacity rather than those with the lowest fixed and variable costs (See Appendix C). This is also supported by observing the results for MHR in test scenario 12. The vehicle route schedules show that multiple vehicles are needed to serve customers.

4.7. Managerial Insights

As a general rule, MHR provides good solutions for the PVRPB and PVRPBTW. MHR should be considered for any problem that cannot be solved to optimality using BMF, including larger routing problems for which a feasible solution cannot be obtained using BMF. Food banks can serve hundreds of charitable agencies and receive food from many food donors, making the use of MHR an attractive scheduling method for many realistic systems.

When scheduling routes based on the PVRPBTW, further analysis of routes and transportation costs for BMF should be evaluated since the formulation provides only a lower bound for transportation costs. Users should evaluate routes to determine if there are any additional costs as a result of vehicle wait time at customer locations. When the total costs after considering these costs is less than that obtained using MHR, routes should be determined using BMF. Otherwise, MHR should be used to schedule transportation. The overall effectiveness of MHR for realistic routing problems is such that the solutions obtained using MHR of generally less than those obtained using TLC.

Unlike the other problem variants, MHR is not necessarily recommended when solving the HPVRPB. It has not been shown effective at discerning which vehicles result in lower transportation costs and may not be an effective routing method without further enhancements which enable MHR to account for differences in vehicle characteristics. Despite its limitations, MHR overcomes the computational complexity of larger problems, making it more appropriate than BMF and TLC.

4.8. Conclusions

In closing, a new HGA affectionately referred to as MULTI-HGA-ROUTE is presented to solve multi-period vehicle routing problems with both linehaul and backhaul customers. As specified for other HGAs, this new metaheuristic combines the population-based search characteristics of genetic algorithms with local search procedures to promote a comprehensive search of the solution space. In addition, MULTI-HGA-ROUTE incorporates tabu search to target a routing day for the diversification, intensification, and learning processes implemented through its reproduction operators. The algorithm also incorporates strategies that encourage a diverse population for each epoch of the search process. MULTI-HGA-ROUTE is tested on pre-existing test scenarios for the PVRPB, PVRPBTW, and HPVRPB. Results show that the algorithm produces good solutions for the PVRPB and PVRPBTW without loss of generality using very little computation time. The results show that MULTI-HGA-ROUTE solves the HPVRPB; however, it is only appropriate in certain situations. Some of the network conditions where the HGA is most appropriate are proposed.

The objective of this research is limited to finding good solutions method for three variants of the PVRPB. Little calibration is done before testing MULTI-HGA-ROUTE. Future research will focus more on calibration and evaluate the metaheuristic for larger problems. Future research will also identify methods that make MULTI-HGA-ROUTE more appropriate for the HPVRPB.

CHAPTER 5

Recommendations and Future Extensions

This dissertation addresses the challenges that food banks experience when overcoming (a) the uncertainty of food availability and (b) the complexity of performing multi-day vehicle routes which complete food collection and distribution requirements without violating food safety considerations. This dissertation addresses each of the challenges through metaheuristic search procedures.

The combined of data mining and supervised machine learning approach presented in this research is recommended in order to provide better estimates for food availability at supermarket branches. The incorporation of a quasi-greedy algorithm to select a non-dominated MLP-NN model for each food type is also recommended. This recommendation is supported by a set of experiments which demonstrate that MLP-NN models provide more accurate predictions for future donation amounts for a regional food bank than traditional forecasting methods. Furthermore, when used to schedule transportation for food collection, the total cost of receiving the aggregate collection amounts predicted by the MLP-NN models were less than or equal to those of all other models. In addition, this doctoral research recommends that transportation practices that are related to one of the PVRPB variants be implemented to manage the collection and delivery of inventoried food items. It is also recommended that a metaheuristic search procedure be implemented to solver larger routing problems. Lastly, it is recommended that MHR be considered to determine vehicle routes for larger networks.

As data warehousing capabilities continues to grow, additional predictors may be identified which give better insight into food availability at supermarket locations. Accordingly, future work will identify methods that allow food banks to approximate the availability of donations at other donor types in the CFSFSC. In addition, future work will use historic data to identify desired supermarket donors as well as integrate inventory management strategies into vehicle routing decisions.

References

Adamowski, J. F. (2008). Peak Daily Water Demand Forecast Modeling Using Artificial Neural Networks. *Journal of Water Resources Planning and Management, March/April*, 119 - 128.

Adya, M., & Collopy, F. (1998). How effective are neural networks at forecasting and prediction? A review and evaluation. *Journal of Forecasting, 17*, 481-495.

Alegre, J., Laguna, M., & Pacheco, J. (2007). Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts. *European Journal of Operational Research, 179*(3), 736-746.

Alonso, F., Alvarez, M., & Beasley, J. (2008). A tabu search algorithm for the periodic vehicle routing problem with multiple vehicle trips and accessibility restrictions. *Journal of the Operational Research Society, 59*(Copyright 2008, The Institution of Engineering and Technology), 963-976.

Angelelli, E., & Speranza, M. G. (2002). The periodic vehicle routing problem with intermediate facilities. *European Journal of Operational Research, 137*(2), 233-247.

Baptista, S., Oliveira, R. C., & Zuquete, E. (2002). A period vehicle routing case study. *European Journal of Operational Research, 139*(Compendex), 220-229.

Bartholdi III, J., Platzman, L., Collins, R., & Warden III, W. (1983). A minimal technology routing system for meals on wheels. *Interfaces*, 1-8.

Beltrami, E., & Bodin, L. (1974). Networks and vehicle routing for municipal waste collection. *Networks, 4*(1), 65-94.

Braysy, O., Dullaert, W., & Gendreau, M. (2004). Evolutionary algorithms for the Vehicle Routing Problem with Time Windows. *Journal of Heuristics, 10*(6), 587-611.

Braysy, O., & Gendreau, M. (2005). Vehicle routing problem with time windows, Part I: Route construction and local search algorithms. *Transportation Science, 39*(1), 104-118.

Brock, L. G., & Davis, L. B. (2012). *An Approach to Approximating Contributions Received From Supermarkets by Food Banks.* Paper presented at the Proceeding to the Industrial and Systems Engineering Research Conference, Orlando, FL.

Cabili, C., Eslami, E., & Briefel, R. (2013). White Paper on the Emergency Food Assistance Program (TEFAP) (USDA, Trans.). In USDA (Ed.). Washington, DC.

Campbell, B. (2011). Second Harvest Food Bank Runs Out of Food. *Fox 8 News*, from http://www.myfox8.com/news/wghp-second-harvest-food-bank-warehouse-out-of-food-20110815,0,3607866.story

Chao, I., Golden, B., & Wasil, E. (1995). An improved heuristic for the period vehicle routing problem. *Networks, 26*(1), 25-44.

Christofides, N., & Beasley, J. (1984). The period routing problem. *Networks, 14*(2), 237–256.

Chunyu, R., Zhendong, S., & Xiaobo, W. (2009). *Study on single and mixed fleet strategy for multi-depot vehicle routing problem with backhauls.* Paper presented at the 2009 International Conference on Computational

Intelligence and Natural Computing, CINC 2009, June 6, 2009 - June 7, 2009, Wuhan, China.

Clarke, G., & Wright, J. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research, 12*(4), 568-581.

Coleman-Jensen, A., Nord, M., Andrews, M., & Carlson, S. (2011). Household Food Security in the United States, 2010 *Economic Research Report No. ERR-125. Washington, DC: USDA Economic Research Service*.

Cordeau, J.-F., Laporte, G., & Mercier, A. (2004). Improved tabu search algorithm for the handling of route duration constraints in vehicle routing problems with time windows. *Journal of the Operational Research Society, 55*(5), 542-546.

Cordeau, J. F., Gendreau, M., & Laporte, G. (1997). A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks, 30*(2), 105-119.

Crone, S. F., Lessmann, S., & Stahlbock, R. (2006). The impact of preprocessing on data mining: An evaluation of classifier sensitivity in direct marketing. *European Journal of Operational Research, 173*(3), 781-800. doi: DOI 10.1016/j.ejor.2005.07.023

Croston, J. (1972). Forecasting and stock control for intermittent demands. *Operational Research Quarterly*, 289-303.

Dantzig, G., & Ramser, J. (1959). The truck dispatching problem. *Management Science, 6*(1), 80-91.

Davis, L. B., E.Sengul, Ivy, J., Brock, L., & Miles, L. (2014). Scheduling Food Bank Collections and Deliveries to Ensure Food Safety and Improve Access. *Accepted for Publication in the International Journal of Public Sector Decision-Making: Socio-Economic Planning Sciences*.

Davis, L. B., Jiang, S. X., Morgan, S. D., & Harris, C. (2013). *Forecasting Donated Food Goods at a Local Food Bank.* Paper presented at the 43rd Annual Southest Decision Sciences Institute, Charleston, SC.

Davis, L. B., Jiang, S. X., & Terry, J. (2013). *Empirical Modeling of In-Kind Donations for a Non-Profit Hunger Relief Organization.* Paper presented at the Industry Studies Association Annual Conference, Kansas City, MO.

Demuth, H., & Beale, M. (1993). *Neural network toolbox for use with MATLAB*.

Drummond, L. M. A., Ochi, L. S., & Vianna, D. S. (2001). Asynchronous parallel metaheuristic for the period vehicle routing problem. *Future Generation Computer Systems, 17*(4), 379-386.

Duhamel, C., Potvin, J.-Y., & Rousseau, J.-M. (1997). A tabu search heuristic for the vehicle routing problem with backhauls and time windows. *Transportation science, 31*(1), 49-59.

Eilon, S., & Christofides, N. (1971). The loading problem. *Management Science*, 259-268.

Ekici, B. B., & Aksoy, U. T. (2009). Prediction of building energy consumption by using artificial neural networks. *Advances in Engineering Software, 40*(5), 356-362.

FA. (2011a). Distressed, Surplus or Unsaleable Product Retrieved October 26, 2011, from http://feedingamerica.org/get-involved/corporate-opportunities/become-a-partner/become-a-product-partner/distresses-surplus-or-unsaleable-product.aspx

FA. (2011b). Our Food Bank Network Retrieved October 26, from http://feedingamerica.org/how-we-fight-hunger/our-food-bank-network.aspx

Feng, C.-X., & Wang, X.-F. (2003). Surface roughness predictive modeling: Neural networks versus regression. *IIE Transactions, 35*, 11-27.

Feng, C.-X., Yu, Z.-G., & Kusiak, A. (2006). Selection and validation of predictive regression and neural network models based on designed experiments. *IIE Transactions, 38*, 13-23.

Firat, M., Yurdusev, M. A., & Turan, M. E. (2009). Evaluation of Artificial Neural Network Techniques for Municipal Water Consumption Model. *Water Resource Management, 23*(4), 617 - 632.

Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks, 11*(2), 109-124.

FRAC. (2013). Food Hardship in America 2012: Data for the Nation, States, 100 MSAs, and Every Congressional District.

Francis, P., Smilowitz, K., & Tzur, M. (2006a). Flexibility and complexity in periodic distribution problems. *Naval Research Logistics, 54*(2), 136-150.

Francis, P., Smilowitz, K., & Tzur, M. (2006b). The period vehicle routing problem with service choice. *Transportation science, 40*(4), 439-454.

Francis, P. M., Smilowitz, K. R., & Tzur, M. (2008). *The period vehicle routing problem and its extensions*.

Gallant, A. R., & White, H. (1988). *There exists a neural network that does not make avoidable mistakes.* Paper presented at the IEEE International Conference on Neural Networks, San Diego, CA.

Garms, L. (2012, August 29, 2012). Shortages Put Food Bank in Crisis Mode, *The Winston-Salem Chronicle*. Retrieved from http://www.wschronicle.com/2012/08/shortages-put-food-bank-in-crisis-mode/

Geem, Z. W., & Roper, W. E. (2009). Energy demand estimation of South Korea using artificial neural network. *Energy Policy*, 4049 - 4054.

Gélinas, S., Desrochers, M., Desrosiers, J., & Solomon, M. M. (1995). A new branching strategy for time constrained routing problems with application to backhauling. *Annals of Operations Research, 61*(1), 91-109.

Goetschalckx, M., & Jacobs-Blecha, C. (1989). Vehicle routing problem with backhauls. *European Journal of Operational Research, 42*, 39-51.

Golden, B., Magnanti, T., & Nguyen, H. (1977). Implementing vehicle routing algorithms. *Networks, 7*(2), 113-148.

GSFB. (2013). Food Bank Programs: Cupboard Collective Retrieved May 20, 2014, from http://www.gsfb.org/how-we-help/programs/cupboard-collective/

Gunes, C., van Hoeve, W. J., & Tayur, S. (2010). Vehicle Routing for Food Rescue Programs: A comparison of different approaches. *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, 176-180.

Gutierreza, R. S., Solisb, A. O., & Mukhopadhyay, S. (2008). Lumpy demand forecasting using neural networks. *International Journal of Production Economics, 111*, 409 - 420.

Hadjiconstantinou, E., & Baldacci, R. (1998). A multi-depot period vehicle routing problem arising in the utilities sector. *Journal of the Operational Research Society*, 1239-1248.

Hansen, P., & Mladenović, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research, 130*(3), 449-467.

Hemmelmayr, V. C., Doerner, K. F., & Hartl, R. F. (2009a). A variable neighborhood search heuristic for periodic routing problems. *European Journal of Operational Research, 195*(3), 791-802. doi: 10.1016/j.ejor.2007.08.048

Hemmelmayr, V. C., Doerner, K. F., & Hartl, R. F. (2009b). A variable neighborhood search heuristic for periodic routing problems. *European Journal of Operational Research, 195*, 791-802.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks, 4*(2), 251-257.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*(5), 359-366.

Karp, R. M. (2010). Reducibility among combinatorial problems *50 Years of Integer Programming 1958-2008* (pp. 219-241): Springer.

Laporte, G. (2007). What you should know about the vehicle routing problem. *Naval Research Logistics, 54*(8), 811-819.

Li, S. T., Shue, L. Y., & Lee, S. F. (2008). Business intelligence approach to supporting strategy-making of ISP service management. *Expert Systems with Applications, 35*(3), 739-754. doi: DOI 10.1016/j.eswa.2007.07.049

Lien, R., Iravani, S. M. R., & Smilowitz, K. R. (2008). *Sequential resource allocation for nonprofit operations*. Department of Industrial Engineering and Management Sciences, Northwestern University.

Lin, S. (1965). Computer solutions of the traveling salesman problem. *Bell System Technical Journal, 44*(10), 2245-2269.

Lin, S., & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, 498-516.

Lippmann, R. P. (1987). An introduction to computing with neural nets. *ARIEL, 209*, 115-245.

Luxhoj, J. T., Williams, T. P., & Shyur, H.-J. (1997). Comparison of regression and neural network models for prediction and inspection profiles for aging aircraft. *IIE Transactions, 29*, 91 - 101.

Mabli, J., Cohen, R., Potter, R., & Zhao, Z. (2010). Hunger in America 2010: National Report Prepared for Feeding America. Princeton, NJ: Mathematica Policy Research, Inc.

Matos, A. C., & Oliveira, R. C. (2004). *An experimental study of the ant colony system for the period vehicle routing problem.* Paper presented at the Ant Colony Optimization and Swarm Intelligence. 4th International Workshop, ANTS 2004. Proceedings, 5-8 Sept. 2004, Berlin, Germany.

Maxwell, S., & Frankenberger, T. (1992). Household Food Security Concepts, Indicators, and Measurements. In UNICEF (Ed.). New York, NY: UNICEF.

McCarty, J. A., & Hastak, M. (2007). Segmentation approaches in data-mining: A comparison of RFM, CHAID, and logistic regression. *Journal of Business Research, 60*(6), 656-662.

Meulstee, P., & Pechenizkiy, M. (2008). *Food Sales Prediction: "If Only It Knew What We Know".* Paper presented at the IEEE International Conference on Data Mining Workshops.

Michallet, J., Prins, C., Amodeo, L., Yalaoui, F., & Vitry, G. (2014). Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services. *Computers & Operations Research, 41*, 196-207.

Mingozzi, A., Giorgi, S., & Baldacci, R. (1999). An exact method for the vehicle routing problem with backhauls. *Transportation Science, 33*, 315-329.

Moller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks, 6*(4), 525-533.

Mosley, J., & Tiehen, L. (2004). The food safety net after welfare reform: Use of private and public food assistance in the Kansas City metropolitan area. *Social Service Review, 78*(2), 267-283.

Mourgaya, M., & Vanderbeck, F. (2007). Column generation based heuristic for tactical planning in multi-period vehicle routing. *European Journal of Operational Research, 183*(3), 1028-1041.

Murat, Y. S., & Ceylan, H. (2006). Use of artificial neural networks for transport energy demand modeling. *Energy Policy, 34*, 3165 - 3172.

Nguyen, P. K., Crainic, T. G., & Toulouse, M. (2014). A hybrid generational genetic algorithm for the periodic vehicle routing problem with time windows. *Journal of Heuristics*, 1-34.

Nord, M., Andrews, M., & Carlson, S. (2009). Household Food Security in the United States, 2008 *Economic Research Report* (Vol. 83): United States Department of Agriculture, Economic Research Service.

Or, I. (1976). *Traveling salesman-type combinatorial problems and their relation to the logistics of regional blood banking.* Northwestern University.

Phillips, C., Hoenigman, R., & Higbee, B. (2011). Food Redistribution as Optimization. *arXiv preprint arXiv:1108.5768*.

Pirkwieser, S., & Raidl, G. R. (2009). *Multiple variable neighborhood search enriched with ilp techniques for the periodic vehicle routing problem with time windows.* Paper presented at the 6th International Workshop on Hybrid Metaheuristics, HM 2009, October 16, 2009 - October 17, 2009, Udine, Italy.

Potvin, J. Y., & Rousseau, J. M. (1995). An exchange heuristic for routeing problems with time windows. *Journal of the Operational Research Society*, 1433-1446.

Pourghaderi, A. R., Tavakkoli-Moghaddam, R., Alinaghian, M., & Beheshti-Pour, B. (2008). *A simple and effective heuristic for periodic vehicle routing problem.* Paper presented at the 2008 IEEE International Conference on Industrial Engineering and Engineering Management, Singapore, Singapore.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research, 31*(12), 1985-2002.

Pulido-Calvo, I., Montesinos, P., Rolda´n, J., & Ruiz-Navarro, F. (2007). Linear regressions and neural approaches towater demand forecasting in irrigation districts with telemetry systems. *Biosystems Engineering, 97*, 283 - 293.

Rahimi-Vahed, A., Crainic, T. G., Gendreau, M., & Rei, W. (2013). A path relinking algorithm for a multi-depot periodic vehicle routing problem. *Journal of Heuristics*, 1-28.

Rusdiansyah, A., & Tsao, D. (2005). An integrated model of the periodic delivery problems for vending-machine supply chains. *Journal of Food Engineering, 70*(3), 421-434.

Russell, R., & Igo, W. (1979). An assignment routing problem. *Networks, 9*(1), 1-17.

Russell, R. A., & Gribbin, D. (1991). A Multiphase Approach to the Period Routing Problem. *Networks, 21*(7), 747-765.

Savelsbergh, M. W. P. (1992). The Vehicle Routing Problem with Time Windows: Minimizing Route Duration. *ORSA Journal on Computing, 4*(2), 146-154.

Sha, D. Y., & Liu, C. H. (2005). Using data mining for due date assignment in a dynamic job shop environment. *International Journal of Advanced Manufacturing Technology, 25*(Copyright 2005, IEE), 1164-1174.

Shahrabi, J., Mousavi, S. S., & Heydar, M. (2009). Supply chain demand forecasting: A comparison of machine learning techniques and traditional methods. *Journal of Applied Sciences, 9*(3), 521 - 527.

Solak, S., Scherrer, C., & Ghoniem, A. (2012). The stop-and-drop problem in nonprofit food distribution networks. *Annals of Operations Research*, 1-20.

Syntetos, A. A., & Boylan, J. E. (2005). The accuracy of intermittent demand estimates. *International Journal of Forecasting, 21*(2), 303-314.

Taillard, É. D., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J. Y. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation science, 31*(2), 170-186.

Tan, C. C. R., & Beasley, J. E. (1984). A heuristic algorithm for the period vehicle routing problem. *Omega, 12*(5), 497-504.

Tarasuk, V., & Eakin, J. (2005). Food assistance through “surplus” food: insights from an ethnographic study of food bank work. *Agriculture and Human Values, 22*(2), 177-186.

Teron, A., & Tarasuk, V. (1999). Charitable food assistance: What are food bank users receiving? *REVUE CANADIENNE DE SANTÉ PUBLIQUE, 90*(6).

Thangiah, S. R., Potvin, J. Y., & Sun, T. (1996). Heuristic approaches to vehicle routing with backhauls and time windows. *Computers and Operations Research, 23*, 1043-1057.

Toth, P., & Vigo, D. (1997). An exact algorithm for the vehicle routing problem with backhauls. *Transportation science, 31*(Copyright 1998, IEE), 372-385.

Toth, P., & Vigo, D. (2002). Models, relaxations and exact approaches for the capacitated vehicle routing problem. *Discrete Applied Mathematics, 123*(1), 487-512.

Tutuncu, G. Y. (2010). An interactive GRAMPS algorithm for the heterogeneous fixed fleet vehicle routing problem with and without backhauls. *European Journal of Operational Research, 201*, 593-600.

Vidal, T., Crainic, T. G., Gendreau, M., Lahrichi, N., & Rei, W. (2012). A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research, 60*(3), 611-624.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2014). A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research, 234*(3), 658-673.

Vinopal, K., & Cooper, R. (2011). Food Hardships in America - 2010: Data for the National, States, and 100 MSA, and Every Congressional District (Vol. 33): Food Research and Action Center.

Wang, H.-F., & Wu, K.-Y. (2004). Hybrid genetic algorithm for optimization problems with permutation property. *Computers & Operations Research, 31*(14), 2453-2471.

Wang, J., & Liang, X. (2009, November 7 - 8, 2009). *The Forecast of Energy Demand on Artificial Neural Network.* Paper presented at the International Conference on Computational Intelligence, Shanghai, China.

Wang, X.-B., & Li, Y.-J. (2009). Study on multi-depot and multi-type vehicles vehicle routing problem with backhauls. *Control and Decision, 24*, 1769-1774.

Wang, X., Sun, J., & Ren, C. (2009). *Study on hybrid genetic algorithm for multi-type vehicles vehicle routing problem with backhauls.* Paper presented at the 2009 6th International Conference on Service Systems and Service Management, ICSSSM '09, June 8, 2009 - June 10, 2009, Xiamen, China.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE, 78*(10), 1550-1560.

Winter, J. (2009). *Comprehensive Planning for the Napa County Food System: A Preliminary Study of Problems and Possibilities.* Master of Science, Cornell University.

Yano, C. A., Chan, T. J., Richter, L. K., Cutler, T., Murty, K. G., & McGettigan, D. (1987). Vehicle routing at quality stores. *Interfaces, 17*(2), 52-63.

Zäpfel, G., & Bögl, M. (2008). Multi-period vehicle routing and crew scheduling with outsourcing options. *International Journal of Production Economics, 113*(2), 980-996.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting, 14*, 35 - 62.

Zhong, Y., & Cole, M. H. (2005). A vehicle routing problem with backhauls and time windows: a guided local search solution. *Transportation Research Part E: Logistics and Transportation Review, 41*(2), 131-144.

Appendix A: Pseudocodes for Heuristics

A.1. Pseudocode for the Quasi-Greedy Algorithm

Notation

k: the number of layers

$\vec{m}$: the number of neurons in each layer

$f(k,\vec{m})$: the predictive error of k-layered MLP-NN with $\vec{m}$ neurons after completing all runs with each data partition

Z_k^*: best model for layer k

i: represent the number of consecutive models that are inferior to Z_k^*

T_1: Maximum value for i allowed

T_2: Binary variable reflecting the satisfaction of termination condition for algorithm

Initialization $Z_k^* = \infty, k = 3, m_{k-1} = 1, i = 0, T_2 = 0$

Do

Calculate $Z_L = f(k,\vec{m})$

if $Z_L < Z_k^*$then $Z_k^* = Z_L, m_{k-1} \leftarrow m_{k-1} + 1, i = 0$

else

if $i < T_1$ then $i = i + 1, m_{k-1} \leftarrow m_{k-1} + 1$

else if $Z_k^* < Z_{k-1}^*$, then $i = 0, k = k + 1$, $m_{k-1} = 1$

else $T_2 = 1$

while $(T_2 = 0)$

A.2. Pseudocode for Initial Route Construction

Step 0: Initialization

a) Randomly generate a sequence of vehicle departures per day, $\boldsymbol{ED}$. Associated with each element, $a \in \boldsymbol{ED}$ is a unique vehicle assignment $v(a)$, the depot location the vehicle departs $n(a)$, and a specific departure day $p(a)$. Note that for this problem, there is only one depot; therefore, $n(a) = 0$ for each element a.

b) Randomly generate linehaul customer visits per day $\boldsymbol{LH}$. Associated with each element, $b_1 \in \boldsymbol{LH}$ is a vehicle assignments, $v(b_1)$, unique customer location $n(b_1)$; and the departure day $p(b_1)$.

c) Randomly generate backhauls customer visits per day $\boldsymbol{BH}$. Associated with each element, $b_2 \in \boldsymbol{BH}$ is a vehicle assignments, $v(b_2)$, unique customer location $n(b_1)$; and the departure day $p(b_2)$.

d) Initialize $\boldsymbol{X_1} = [\quad], \boldsymbol{X_2} = [\quad], \boldsymbol{X_3} = [\quad]$. Let i, j, and l correspond to a specific position in vector $\boldsymbol{ED}$, $\boldsymbol{LH}$, and $\boldsymbol{BH}$, respectively. Let $[m]$ represent the position in which a specific vehicle departure or customer visit option is to be assigned in the solution. At initialization, $i = 0$, $j = 0$, $l = 0$, and $m = 0$.

Step 1: Select a vehicle departure option and determine if it will be a selected or unselected vehicle departure option.

a) Set $i = i + 1$ and $m = m + 1$.

b) Select the i-th vehicle departure option, $a_{[i]}$ from $\boldsymbol{ED}$.

c) Determine the ordered set of unsatisfied linehaul customers $\boldsymbol{U_L}$ and unsatisfied backhaul customers $\boldsymbol{U_B}$that can be assigned to vehicle $v(a_{[i]})$ based on satisfying $p(b_{1,[j]}) = p(a_{[i]})\ \forall j \in \boldsymbol{LH}$ and $p(b_{2,[l]}) = p(a_{[i]}) \forall l \in \boldsymbol{BH}$.

d) If $\boldsymbol{U_L} \cup \boldsymbol{U_B} \neq \emptyset$, set $\boldsymbol{X}_{1,[m]}= a_{[i]}$, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$ and GOTO Step 2. Otherwise, set $\boldsymbol{X}_{1,[m]}= a_{[i]}$, $\boldsymbol{X}_{2,[m]}$ = 2, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$ and GOTO Step 4.

Step 2: If $a_{[i]}$ is the last unassigned vehicle departure option for day $p(a_{[i]})$

a) Let $k_1 = |\boldsymbol{U_L}|$and $k_2 = |\boldsymbol{U_B}|$

b) For $j = 1$ to k_1,

i) Set $m = m + 1$

ii) Set b = the k^{th} element in U_L

iii) set $\boldsymbol{X}_{1,[m]}= b$, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

c) For $l = 1\ to\ k_2$,

i) Set $m = m + 1$

ii) Set b = the l^{th} element in U_B

iii) set $\boldsymbol{X}_{1,[m]}= b$, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

d) GOTO Step 1.

Step 3: If $a_{[i]}$ is not the last unassigned vehicle departure option for day $p(a_{[i]})$

a) Randomly determine whether to assign only linehauls, only backhauls, or both linehauls and backhauls to the vehicle

i) If only linehaul customers are to be assigned to the vehicle,

(a) Randomly generate k_1 and set $k'_1 = min(k_1, |\boldsymbol{U_L}|)$

(b) For $j = 1$ to k'_1,

(1) Set $m = m + 1$

(2) Set b = the jth element in U_L

(3) Set $\boldsymbol{X}_{1,[m]}$= b, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

ii) If only backhaul customers are to be assigned to the vehicle,

(a) Randomly generate k_2 and set $k'_2 = min(k_2, |\boldsymbol{U_B}|)$

(b) For $l = 1$ to k'_2,

(1) Set $m = m + 1$

(2) Set b = the l^{th} element in U_B

(3) set $\boldsymbol{X}_{1,[m]}$= b, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

iii) If linehaul and backhaul customers are to be assigned to the vehicle,

(a) Randomly generate k_1 and set $k'_1 = min(k_1, |\boldsymbol{U_L}|)$

(b) Randomly generate k_2 and set $k'_2 = min(k_2, |\boldsymbol{U_B}|)$

(c) For $j = 1$ to k'_1,

(1) Set $m = m + 1$

(2) Set b = the jth element in U_L

(3) Set $\boldsymbol{X}_{1,[m]}$= b, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

(d) For $l = 1$ to k'_2,

(1) Set $m = m + 1$

(2) Set b = the l^{th} element in U_B

(3) set $\boldsymbol{X}_{1,[m]}$= b, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

b) GOTO Step 1.

Step 4: If all vehicles options have not been assigned, GOTO Step 1, otherwise GOTO Step 5.

Step 5: Assign all remaining unassigned elements of $\boldsymbol{LH}$ and $\boldsymbol{BH}$ as unselected depot departure options

a) Determine all unassigned elements of $\boldsymbol{LH}$ and $\boldsymbol{BH}$ and assign them to $\boldsymbol{U_L}$ and $\boldsymbol{U_B}$, respectively**.**

b) Let $k_1 = |\boldsymbol{U_L}|$and $k_2 = |\boldsymbol{U_B}|$

c) For $j = 1$ to k_1,

   a) Set $m = m + 1$

   b) Set b = the k^{th} element in U_L

   c) set $\boldsymbol{X}_{1,[m]}$= b, $\boldsymbol{X}_{2,[m]}$ = 1, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

d) For $l = 1\ to\ k_2$,

   a) Set $m = m + 1$

   b) Set b = the l^{th} element in U_B

   c) set $\boldsymbol{X}_{1,[m]}$= b, $\boldsymbol{X}_{2,[m]}$ = 3, and $\boldsymbol{X}_{3,[m]} = v(a_{[i]})$

e) GOTO Step 6

Step 6: Exit Code

A.3. Pseudocode for Crossover Operator

Step 0: Initialization: Determine the targeted reproduction day and two parents P_1 and P_2

Step 1: Identify Customers Currently Served

a) Identify customers served in parent P_1 on the targeted reproduction day.

b) Identify customers served in parent P_2 on the targeted reproduction day.

Step 2: Offspring Construction

a) Copy the base components for parents P_1 and P_2 to create offspring θ_1 and θ_2, respectively.

b) Calculate the change in the number of visits to each customer in θ_1 on the reproduction day.

c) Calculate the change in the number of visits to each customer in θ_2 on the reproduction day.

Step 3: Identify Customers Served After Construction

a) Replace the genes of θ_1 pertaining to the targeted reproduction day with the genes of P_2 pertaining to the targeted reproduction day.

b) Replace the genes of θ_2 pertaining to the targeted reproduction day with the genes of P_1 pertaining to the targeted reproduction day.

c) Calculate the net change in the number of times each customer is served in θ_1.

d) Calculate the net change in the number of times each customer is served in θ_2.

Step 4: Correct Violations to Customer Service Requirements

For each customer

a) If the customer is over-served

1. If there exists at least one day that is customer is served that is not in the tabu list

    i) Find the gene i corresponding to a selected customer visit options not in the tabu list.

    ii) Change one of the corresponding genes to an unselected customer visit option by setting its allele $\alpha_{2,i} = 3$. If more than one gene fits this condition, select one according to a uniform distribution.

    iii) GOTO Step 5

2. If all days that the customer is served is in the tabu list

    i) Find the gene i corresponding to the selected customer visit option which occurs on the day associated with the earliest entry into the tabu list.

    ii) Change the corresponding genes to an unselected customer visit option by setting its allele $\alpha_{2,i} = 3$.

3. If the vehicle that served the removed option is now empty, change the gene corresponding to its use to an unselected vehicle departure by changing the corresponding allele $\alpha_{2,i} = 2$.

    iii) GOTO Step 5.

b) If the customer is underserved,

1. If there exists at least one day not in the tabu list where the customer is not served
   i) Find the gene i corresponding to an unselected customer visit options not in the tabu list.
   ii) Change one of the corresponding genes to a selected customer visit option by setting its allele $\alpha_{2,i} = 1$. If more than one gene fits this condition, select one according to a uniform distribution.
   iii) Change the corresponding genes to an unselected customer visit option by setting its allele $\alpha_{2,i} = 3$.
   iv) GOTO Step 5.
2. If there are no days in the tabu list where the customer is not served
   i) Revert the gene i corresponding to the unselected customer visit obtained through crossover by setting its allele $\alpha_{2,i} = 3$.
   ii) GOTO Step 5.
3. If there is an unassigned vehicle on the same day as the added customer
   i) Let $\alpha_{1,j}$, $\alpha_{2,j}$, and $\alpha_{3,j}$represent an alleles of $\boldsymbol{X_1}$,$\boldsymbol{X_2}$, and $\boldsymbol{X_3}$that corresponds to an arbitrarily-selected unassigned vehicle on the same day as the added customer.
   ii) Arbitrarily select one of the extra depots, setting the respective allele to $\alpha_{2,j} = 1$.

iii) Assign the added customer to the newly added vehicle by setting $\alpha_{3,i} = \alpha_{3,j}$.

iv) If the assigned vehicle is positioned after the added customer, exchange the relative positions of the two alleles

v) GOTO Step 5.

4. If there are no unassigned vehicles on the same day as the added customer

   i) Let $\alpha_{1,j}$, $\alpha_{2,j}$, and $\alpha_{3,j}$represent an alleles of $\boldsymbol{X_1}$,$\boldsymbol{X_2}$, and $\boldsymbol{X_3}$that corresponds to an arbitrarily selected vehicles route that occurs on the same day as the added customer.

   ii) Assign the added customer to the newly added vehicle by setting $\alpha_{3,i} = \alpha_{3,j}$.

   iii) If the assigned vehicle is positioned after the added customer, exchange the relative positions of the two alleles.

   iv) Sort customers on the vehicle route such that linehauls are served before backhauls, maintaining the relative order in which linehaul and backhaul customers are served.

Step 5: Calculate updated tour travel distance, travel time, and arrival/departure times for all locations.

Step 6: Exit Pseudocode

A.4. Pseudocode for Modified Local Search Operators

A.4.1. Modified Two Opt Inter-Route Operator*

Step 0: RVNS selects Two Opt* to explore neighborhoods

Step 1: Determine if there are enough vehicle to execute Two Opt* Operator

a) Determine if more than one route is assigned on the targeted routing day

b) If only one vehicle is assigned, Goto Step 6.

Step 2: Randomly select two routes

Step 3: Select move customers from route 1

a) Select a customer and identify the customer type (linehaul or backhaul)

b) Randomly select the route segment from the selected customer to the end of route to be moved to route 2.

Step 4: Determine if Two Opt* is executed

a) If route 2 does not serve at least one customer of the same type as the customers selected from route 1.

  i) Do not move customers from route 1

  ii) Goto Step 6

Step 5: Execute Two Opt* Operator

a) Select the position of a customer of the same type as the customer type selected for route 1.

b) Move customers from the selected position to the end of tour.

c) Insert the route segment customers moved from route 2 into route 1

d) Insert the route segment customers moved from route 1 into route 2.

Step 6: Exit Two Opt* Operator

A.4.2. Modified RELOCATE Inter-Route Move Operator

Step 0: RVNS selects RELOCATE to explore neighborhoods

Step 1: Determine if the routing conditions are appropriate execute RELOCATE

a) Determine if more than one route is assigned on the targeted routing day

b) If only one vehicle is assigned, Goto Step 5.

Step 2: Randomly select two routes

Step 3: Move customers from route 1

i) Randomly select a route segment containing k customers

ii) Shift customer that remain k positions towards the start of the route

Step 4: Insert the moved customers into route 2

a) If the customers moved are all linehauls,

i) If route 2 currently serves only linehaul customers,

1. Randomly select the position of another linehaul customer as the insertion point

2. Goto Step 4.a.iv

ii) If route 2 currently serves only backhaul customers,

1. Select the position of the first backhaul customer as the insertion point
2. Goto Step 4.a.iv

iii) If route 2 currently serves both linehaul and backhaul customers,

1. Randomly select the position of another linehaul customer as the insertion point
2. Goto Step 4.a.iv

iv) Shift customers currently at or after the insertion point k positions away from the start of the route

v) Insert the move customers into route 2 at the insertion point

vi) Goto Step 5

b) If the customers moved are all backhauls,

i) If route 2 currently serves only linehaul customers,

1. Select the position of the last linehaul customer as the pre-insertion point
2. Goto Step 4.b.iv

ii) If route 2 currently serves only backhaul customers,

1. Randomly select the position of any backhaul customer as the pre-insertion point
2. Goto Step 4.b.iv

iii) If route 2 currently serves both linehaul and backhaul customers,

1. Randomly select the position of either the last linehaul customer or any backhaul customer as the pre-insertion point
2. Goto Step 4.b.iv

iv) Shift customers after the pre-insertion point backwards k positions.

v) Place the moved customers into route 2 in the position immediately after the pre-insertion point

vi) Goto Step 5

c) If moved customers are include both linehauls and backhauls,

i) If route 2 currently serves only linehaul customers,

1. Select the position after the last linehaul customer as the insertion point
2. Goto Step 4.c.iv

ii) If route 2 currently serves only backhaul customers,

1. Select the position before the first backhaul customer as the insertion point
2. Goto Step 4.c.iv

iii) If route 2 currently serves only linehaul and backhaul customers,

1. Select the position after the last linehaul customer as the insertion point
2. Goto Step 4.c.iv

iv) Shift customers currently at or after the insertion point backwards k positions.

v) Insert the move customers into route 2 at the insertion point

vi) Goto Step 5

Step 5: Exit RELOCATE Operator

A.4.3. Modified CROSS Inter-Route Operator

Step 0: RVNS selects CROSS to explore neighborhoods

a) Let k1 and

Step 1: Determine if there are enough vehicles to execute CROSS Operator

c) Determine if more than one route is assigned on the targeted routing day

d) If only one vehicle is assigned, Goto Step 6.

Step 2: Randomly select two routes

Step 3: Select move customers from route 1

a) Select a customer and identify the customer type ρ

b) Randomly select a segment containing k_1 customers of type ρ to be moved to route 2.

Step 4: Determine if CROSS is executed

a) If route 2 does not serve at least one customer of type ρ,

a. Do not move customers from route 1

b. Goto Step 6

Step 5: Execute CROSS Operator

a) Select at $k_2$customers from route 2 that are of type ρ.

b) Insert the $k_1$customers moved from route 1 into route 2 in the relative position of the first customer moved from route 2.

c) Insert the $k_2$customers moved from route 2 into route 1 in the relative position of the first customer moved from route 1.

Step 6: Exit CROSS Operator

A.4.4. Modified Two Opt Intra-Route Operator

Step 0: Two Opt Intra-route heuristic is called by intra-route procedure

Step 1: Determine if heuristic is executed

a) If no route is performed Goto Step 3.

b) Select a route. If only one customer is assigned to the route, Goto Step 3.

c) Randomly select a customer. Let ρ represent the classification of the customer as a linehaul or backhaul

d) Find all other customers of type ρ on the route

e) If there are no other customers of type ρ, Goto Step 3.

Step 2: Execute Two Opt

a) Let ε represent the sequence of customers of type ρ

b) Invert ε

Step 3: Exit Modified Two Opt

A.4.5. Modified Or Operator

Step 0: Or Intra-route heuristic is called by intra-route procedure

Step 1: Determine if Or Operator is executed

a) If no route is performed Goto Step 4.

b) Select a route. If only one customer is assigned to the route, Goto Step 4.

Step 2: Select move customers on the selected route

a) Randomly select an route segment containing k customers

b) Shift customer that remain on route 1 forward k positions

Step 3: Insert the moved customers into another position on the route

a) If the customers moved are all linehauls,

i) If the remaining customers served are all linehauls,

1. Randomly select the position of a linehaul customer as the insertion point
2. Goto Step 3.a.iv

ii) If the remaining customers served are all backhaul,

1. Select the position of the first backhaul customer as the insertion point

2. Goto Step 3.a.iv

iii) If the both linehaul and backhaul customers remain,

1. Randomly select the position of another linehaul customer as the insertion point
2. Goto Step 3.a.iv

iv) Shift customers currently at or after the insertion point backwards k positions

v) Insert the move customers into the route at the insertion point

vi) Goto Step 4

b) If the customers moved are all backhauls,

i) If the remaining customers served are all linehauls,

1. Select the position of the last linehaul customer as the pre-insertion point
2. Goto Step 3.b.iv

ii) If the remaining customers served are all backhauls,

1. Randomly select the position of any backhaul customer as the pre-insertion point
2. Goto Step 3.b.iv

iii) If both linehaul and backhaul customers remain on the route,

1. Randomly select the position of either the last linehaul customer or any backhaul customer as the pre-insertion point
2. Goto Step 3.b.iv

iv) Shift customers after the pre-insertion point backwards k positions.

v) Place the moved customers into the route in the position immediately after the pre-insertion point

vi) Goto Step 4

c) If moved customers are include both linehauls and backhauls,

i) If remaining customers served are all linehauls,

1. Select the position after the last linehaul customer as the insertion point
2. Goto Step 3.c.iv

ii) If remaining customers served are all backhauls,

1. Randomly select the position of any backhaul customer as the insertion point
2. Goto Step 3.c.iv

iii) If both linehaul and backhaul customers remain on the route,

1. Select the position after the last linehaul customer as the insertion point
2. Goto Step 3.c.iv

iv) Shift customers currently at or after the insertion point forward k positions.

v) Insert the move customers at the insertion point

vi) Goto Step 4

Step 5: Exit Or Operator

Appendix B: Food Bank Network Characteristics

B.1. Linehaul Customers

Location (Ci)	Earliest Arrival	Latest Departure	Amount (in lbs.)	Service Time (in hours)	Required Arrivals
1	2	6	9,529.00	1	1
2	4	6	9,999.00	1	1
3	5	10	9,999.00	1	1
4	3	5	1,248.00	1	1
5	0	6	9,999.00	1	1
6	1	3	2,385.00	1	1
7	2	4	9,600.00	1	1
8	3	5	9,999.00	1	1
9	0	8	6,877.00	1	1
10	1	8	7,386.00	1	1
11	0	7	1,519.00	1	1
12	2	8	1,390.00	1	1
13	3	5	3,264.00	1	1
14	3	7	8,256.00	1	1

B.2. Backhaul Customers

Location (Ci)	Earliest Arrival	Latest Departure	Amount (in lbs.)	Service Time (in hours)	Required Arrivals
4	3	5	300.00	0.5	3
5	0	6	300.00	0.5	3
6	1	3	300.00	0.5	3
7	2	4	300.00	0.5	3
8	3	5	300.00	0.5	3
9	0	8	300.00	0.5	3
10	1	8	300.00	0.5	3
11	0	7	300.00	0.5	3
12	2	8	300.00	0.5	3
13	3	5	300.00	0.5	3
14	3	7	300.00	0.5	3
15	5	7	300.00	0.5	3
16	3	7	300.00	0.5	3
17	1	7	300.00	0.5	3

Location	Earliest Arrival	Latest Departure	Amount (in lbs.)	Service Time (in hours)	Required Arrivals
18	3	10	300.00	0.5	3
19	0	6	300.00	0.5	3
20	1	8	300.00	0.5	3
21	5	7	300.00	0.5	3
22	0	6	300.00	0.5	3
23	5	7	300.00	0.5	3
24	4	6	300.00	0.5	3
25	4	6	300.00	0.5	3
26	0	8	300.00	0.5	3
27	8	10	300.00	0.5	3
28	1	6	300.00	0.5	3
29	6	8	300.00	0.5	3
30	1	3	300.00	0.5	3
31	6	8	300.00	0.5	3
32	0	8	300.00	0.5	3
33	4	9	300.00	0.5	3

Location (Ci)	Earliest Arrival	Latest Departure	Amount (in lbs.)	Service Time (in hours)	Required Arrivals
34	0	2	300.00	0.5	3
35	3	6	300.00	0.5	3
36	0	5	300.00	0.5	3
37	3	7	300.00	0.5	3
38	5	7	300.00	0.5	3
39	6	8	300.00	0.5	3
40	6	9	300.00	0.5	3
41	1	10	300.00	0.5	3
42	5	7	300.00	0.5	3

B.3. Vehicle Characteristics by Fleet Type

Fleet Type	Vehicle	Capacity (lbs.)	Fixed Cost ($/Use)	Fuel Efficiency ($/mile)	Refrigeration Cost ($/hour)
1	1 – 7	20000	150.00	0.40	1.25
	1	20000	150.00	0.40	1.25
	2	18550	125.00	0.40	1.50
	3	15000	100.00	0.35	1.75
2	4	15000	80.00	0.30	2.00
	5	12500	60.00	0.25	2.25
	6	5000	50.00	0.20	2.50
	7	1000	0.00	0.15	3.00

Homogeneous Fleet indicated by Fleet Type = 1
Heterogeneous Fleet indicated by Fleet Type = 2

B.4. Travel Time Between Locations

	To Cj							
From Ci	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
0	-	0.8833	0.4833	0.7500	0.2500	0.3000	0.2500	0.8667
1	0.8833	-	1.1833	1.1667	0.9000	0.7667	0.9333	1.2333
2	0.4833	1.1833	-	0.8667	0.4333	0.6333	0.5167	1.1167
3	0.7500	1.1667	0.8667	-	0.8000	0.5333	0.8167	1.5333
4	0.2500	0.9000	0.4333	0.8000	-	0.3167	0.2333	0.9833
5	0.3000	0.7667	0.6333	0.5333	0.3167	-	0.3667	1.0833
6	0.2500	0.9333	0.5167	0.8167	0.2333	0.3333	-	1.0167
7	0.8667	1.2333	1.1167	1.5333	0.9833	1.0833	1.0167	-
8	0.7333	1.0500	1.1333	1.4000	0.8500	0.9333	0.8833	0.7833
9	0.9333	1.2333	1.3333	1.5833	1.0500	1.1333	1.0833	0.9667
10	0.4000	0.9667	0.7833	1.0500	0.5167	0.6000	0.5333	0.5500

	To Cj							
From Ci	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
11	0.4833	1.1833	-	0.8667	0.4333	0.6333	0.5167	1.1167
12	0.7500	1.1667	0.8667	-	0.8000	0.5333	0.8167	1.5333
13	0.3000	0.7667	0.6333	0.5333	0.3167	-	0.3667	1.0833
14	0.7333	1.0500	1.1333	1.4000	0.8500	0.9333	0.8833	0.7833
15	0.8833	-	1.1833	1.1667	0.9000	0.7667	0.9333	1.2333
16	0.4833	1.1833	-	0.8667	0.4333	0.6333	0.5167	1.1167
17	0.7500	1.1667	0.8667	-	0.8000	0.5333	0.8167	1.5333
18	0.2500	0.9000	0.4333	0.8000	-	0.3167	0.2333	0.9833
19	0.3000	0.7667	0.6333	0.5333	0.3167	-	0.3667	1.0833
20	0.2500	0.9333	0.5167	0.8167	0.2333	0.3333	-	1.0167
21	0.8667	1.2333	1.1167	1.5333	0.9833	1.0833	1.0167	-
22	0.7333	1.0500	1.1333	1.4000	0.8500	0.9333	0.8833	0.7833

	To Cj							
From Ci	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
23	0.9333	1.2333	1.3333	1.5833	1.0500	1.1333	1.0833	0.9667
24	0.4000	0.9667	0.7833	1.0500	0.5167	0.6000	0.5333	0.5500
25	0.2667	0.9500	0.6500	0.9167	0.3833	0.4667	0.4000	0.7167
26	0.3000	0.7500	0.6333	0.8000	0.3333	0.3500	0.3667	0.7667
27	1.3000	0.5000	1.6500	1.6833	1.3333	1.2667	1.3833	1.3500
28	0.5500	1.2333	0.6333	1.0833	0.6000	0.6667	0.5667	0.6333
29	0.9500	1.2667	1.3500	1.6167	1.0833	1.1667	1.1000	1.0000
30	0.6833	1.0500	1.0833	1.3333	0.8000	0.8833	0.8167	0.2500
31	1.0333	0.6500	1.4000	1.5667	1.0833	1.0667	1.1333	1.0833
32	0.4833	1.0833	0.7667	1.2000	0.6667	0.7500	0.7000	0.6000
33	0.5833	0.7333	0.9667	1.2333	0.7000	0.7667	0.7167	0.6167

From Ci	To Cj 0	1	2	3	4	5	6	7
34	0.5333	0.9000	0.9167	1.1833	0.6500	0.7167	0.6667	0.5000
35	0.3500	0.7167	0.7333	1.0000	0.4667	0.5500	0.4833	0.5833
36	0.8833	1.5333	0.7000	0.5333	0.7167	0.9500	0.9000	1.6667
37	0.7500	1.4333	0.3500	0.9167	0.6833	0.8833	0.7667	1.3667
38	1.3167	2.0000	0.9167	1.5000	1.2500	1.4500	1.3333	1.9333
39	1.5500	2.2333	1.1500	1.7333	1.4833	1.6833	1.5667	2.1667
40	1.1167	1.8000	0.7000	1.2833	1.0500	1.2333	1.1333	1.7333
41	1.6500	2.3000	1.4667	1.5000	1.4833	1.7500	1.6667	2.4167
42	1.0000	1.6667	0.8167	0.9333	0.8333	1.1000	1.0333	1.7833

	To Cj							
From Ci	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
0	0.7333	0.9167	0.4000	0.4833	0.7500	0.3000	0.7333	0.8833
1	1.0500	1.2333	0.9667	1.1833	1.1667	0.7667	1.0500	-
2	1.1333	1.3333	0.7833	-	0.8667	0.6333	1.1333	1.1833
3	1.4000	1.5833	1.0500	0.8667	-	0.5333	1.4000	1.1667
4	0.8500	1.0500	0.5167	0.4333	0.8000	0.3167	0.8500	0.9000
5	0.9333	1.1333	0.6000	0.6333	0.5333	-	0.9333	0.7667
6	0.8833	1.0833	0.5333	0.5167	0.8167	0.3333	0.8833	0.9333
7	0.7833	0.9667	0.5500	1.1167	1.5333	1.0833	0.7833	1.2333
8	-	0.2333	0.5500	1.1333	1.4000	0.9333	-	1.0500
9	0.2333	-	0.7500	1.3333	1.5833	1.1333	0.2333	1.2333
10	0.5500	0.7500	-	0.7833	1.0500	0.6000	0.5500	0.9667

From Ci	To Cj 8	9	10	11	12	13	14	15
11	1.1333	1.3333	0.7833	-	0.8667	0.6333	1.1333	1.1833
12	1.4000	1.5833	1.0500	0.8667	-	0.5333	1.4000	1.1667
13	0.9333	1.1333	0.6000	0.6333	0.5333	-	0.9333	0.7667
14	-	0.2333	0.5500	1.1333	1.4000	0.9333	-	1.0500
15	1.0500	1.2333	0.9667	1.1833	1.1667	0.7667	1.0500	-
16	1.1333	1.3333	0.7833	-	0.8667	0.6333	1.1333	1.1833
17	1.4000	1.5833	1.0500	0.8667	-	0.5333	1.4000	1.1667
18	0.8500	1.0500	0.5167	0.4333	0.8000	0.3167	0.8500	0.9000
19	0.9333	1.1333	0.6000	0.6333	0.5333	-	0.9333	0.7667
20	0.8833	1.0833	0.5333	0.5167	0.8167	0.3333	0.8833	0.9333
21	0.7833	0.9667	0.5500	1.1167	1.5333	1.0833	0.7833	1.2333
22	-	0.2333	0.5500	1.1333	1.4000	0.9333	-	1.0500

From Ci	To Cj 8	9	10	11	12	13	14	15
23	0.2333	-	0.7500	1.3333	1.5833	1.1333	0.2333	1.2333
24	0.5500	0.7500	-	0.7833	1.0500	0.6000	0.5500	0.9667
25	0.6667	0.8500	0.1667	0.6500	0.9167	0.4667	0.6667	0.9500
26	0.6167	0.8000	0.5167	0.6333	0.8000	0.3500	0.6167	0.7500
27	1.1167	1.2667	1.1333	1.6500	1.6833	1.2667	1.1167	0.5000
28	0.8333	1.0167	0.4167	0.6333	1.0833	0.6667	0.8333	1.2333
29	0.2833	0.3167	0.7833	1.3500	1.6167	1.1667	0.2833	1.2667
30	0.6000	0.7833	0.3333	1.0833	1.3333	0.8833	0.6000	1.0500
31	0.8333	1.0167	0.8667	1.4000	1.5667	1.0667	0.8333	0.6500
32	0.6667	0.8500	0.2333	0.7667	1.2000	0.7500	0.6667	1.0833
33	0.3833	0.5667	0.4000	0.9667	1.2333	0.7667	0.3833	0.7333

	To Cj							
From Ci	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
34	0.3500	0.5333	0.3000	0.9167	1.1833	0.7167	0.3500	0.9000
35	0.4333	0.6167	0.3500	0.7333	1.0000	0.5500	0.4333	0.7167
36	1.5167	1.7000	1.1667	0.7000	0.5333	0.9500	1.5167	1.5333
37	1.3833	1.5667	1.0333	0.3500	0.9167	0.8833	1.3833	1.4333
38	1.9333	2.1167	1.6000	0.9167	1.5000	1.4500	1.9333	2.0000
39	2.1667	2.3667	1.8333	1.1500	1.7333	1.6833	2.1667	2.2333
40	1.7333	1.9167	1.4000	0.7000	1.2833	1.2333	1.7333	1.8000
41	2.2667	2.4667	1.8667	1.4667	1.5000	1.7500	2.2667	2.3000
42	1.6333	1.8167	1.2833	0.8167	0.9333	1.1000	1.6333	1.6667

	To Cj							
From Ci	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*
0	0.4833	0.7500	0.2500	0.3000	0.2500	0.8667	0.7333	0.9167
1	1.1833	1.1667	0.9000	0.7667	0.9333	1.2333	1.0500	1.2333
2	-	0.8667	0.4333	0.6333	0.5167	1.1167	1.1333	1.3333
3	0.8667	-	0.8000	0.5333	0.8167	1.5333	1.4000	1.5833
4	0.4333	0.8000	-	0.3167	0.2333	0.9833	0.8500	1.0500
5	0.6333	0.5333	0.3167	-	0.3667	1.0833	0.9333	1.1333
6	0.5167	0.8167	0.2333	0.3333	-	1.0167	0.8833	1.0833
7	1.1167	1.5333	0.9833	1.0833	1.0167	-	0.7833	0.9667
8	1.1333	1.4000	0.8500	0.9333	0.8833	0.7833	-	0.2333
9	1.3333	1.5833	1.0500	1.1333	1.0833	0.9667	0.2333	-
10	0.7833	1.0500	0.5167	0.6000	0.5333	0.5500	0.5500	0.7500

	To Cj							
From Ci	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*
11	-	0.8667	0.4333	0.6333	0.5167	1.1167	1.1333	1.3333
12	0.8667	-	0.8000	0.5333	0.8167	1.5333	1.4000	1.5833
13	0.6333	0.5333	0.3167	-	0.3667	1.0833	0.9333	1.1333
14	1.1333	1.4000	0.8500	0.9333	0.8833	0.7833	-	0.2333
15	1.1833	1.1667	0.9000	0.7667	0.9333	1.2333	1.0500	1.2333
16	-	0.8667	0.4333	0.6333	0.5167	1.1167	1.1333	1.3333
17	0.8667	-	0.8000	0.5333	0.8167	1.5333	1.4000	1.5833
18	0.4333	0.8000	-	0.3167	0.2333	0.9833	0.8500	1.0500
19	0.6333	0.5333	0.3167	-	0.3667	1.0833	0.9333	1.1333
20	0.5167	0.8167	0.2333	0.3333	-	1.0167	0.8833	1.0833
21	1.1167	1.5333	0.9833	1.0833	1.0167	-	0.7833	0.9667
22	1.1333	1.4000	0.8500	0.9333	0.8833	0.7833	-	0.2333

From Ci	To Cj 16	17	18	19	20	21	22	23
23	1.3333	1.5833	1.0500	1.1333	1.0833	0.9667	0.2333	-
24	0.7833	1.0500	0.5167	0.6000	0.5333	0.5500	0.5500	0.7500
25	0.6500	0.9167	0.3833	0.4667	0.4000	0.7167	0.6667	0.8500
26	0.6333	0.8000	0.3333	0.3500	0.3667	0.7667	0.6167	0.8000
27	1.6500	1.6833	1.3333	1.2667	1.3833	1.3500	1.1167	1.2667
28	0.6333	1.0833	0.6000	0.6667	0.5667	0.6333	0.8333	1.0167
29	1.3500	1.6167	1.0833	1.1667	1.1000	1.0000	0.2833	0.3167
30	1.0833	1.3333	0.8000	0.8833	0.8167	0.2500	0.6000	0.7833
31	1.4000	1.5667	1.0833	1.0667	1.1333	1.0833	0.8333	1.0167
32	0.7667	1.2000	0.6667	0.7500	0.7000	0.6000	0.6667	0.8500
33	0.9667	1.2333	0.7000	0.7667	0.7167	0.6167	0.3833	0.5667

From Ci	To Cj 16	17	18	19	20	21	22	23
34	0.9167	1.1833	0.6500	0.7167	0.6667	0.5000	0.3500	0.5333
35	0.7333	1.0000	0.4667	0.5500	0.4833	0.5833	0.4333	0.6167
36	0.7000	0.5333	0.7167	0.9500	0.9000	1.6667	1.5167	1.7000
37	0.3500	0.9167	0.6833	0.8833	0.7667	1.3667	1.3833	1.5667
38	0.9167	1.5000	1.2500	1.4500	1.3333	1.9333	1.9333	2.1167
39	1.1500	1.7333	1.4833	1.6833	1.5667	2.1667	2.1667	2.3667
40	0.7000	1.2833	1.0500	1.2333	1.1333	1.7333	1.7333	1.9167
41	1.4667	1.5000	1.4833	1.7500	1.6667	2.4167	2.2667	2.4667
42	0.8167	0.9333	0.8333	1.1000	1.0333	1.7833	1.6333	1.8167

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
0	0.4000	0.2667	0.3000	1.3000	0.5500	0.9500	0.6833	1.0333
1	0.9667	0.9500	0.7500	0.5000	1.2333	1.2667	1.0500	0.6500
2	0.7833	0.6500	0.6333	1.6500	0.6333	1.3500	1.0833	1.4000
3	1.0500	0.9167	0.8000	1.6833	1.1333	1.6167	1.3333	1.5667
4	0.5167	0.3833	0.3333	1.3333	0.6000	1.0833	0.8000	1.0833
5	0.6000	0.4667	0.3500	1.2667	0.6667	1.1667	0.8833	1.0667
6	0.5333	0.4000	0.3667	1.3833	0.5667	1.1000	0.8167	1.1333
7	0.5500	0.7167	0.7667	1.3500	0.6333	1.0000	0.2500	1.0833
8	0.5500	0.6667	0.6167	1.1167	0.8333	0.2833	0.6000	0.8333
9	0.7500	0.8500	0.8000	1.2667	1.0167	0.3167	0.7833	1.0167
10	-	0.1667	0.5167	1.1333	0.4167	0.7833	0.3333	0.8667

From Ci	To Cj 24	25	26	27	28	29	30	31
11	0.7833	0.6500	0.6333	1.6500	0.6333	1.3500	1.0833	1.4000
12	1.0500	0.9167	0.8000	1.6833	1.1333	1.6167	1.3333	1.5667
13	0.6000	0.4667	0.3500	1.2667	0.6667	1.1667	0.8833	1.0667
14	0.5500	0.6667	0.6167	1.1167	0.8333	0.2833	0.6000	0.8333
15	0.9667	0.9500	0.7500	0.5000	1.2333	1.2667	1.0500	0.6500
16	0.7833	0.6500	0.6333	1.6500	0.6333	1.3500	1.0833	1.4000
17	1.0500	0.9167	0.8000	1.6833	1.1333	1.6167	1.3333	1.5667
18	0.5167	0.3833	0.3333	1.3333	0.6000	1.0833	0.8000	1.0833
19	0.6000	0.4667	0.3500	1.2667	0.6667	1.1667	0.8833	1.0667
20	0.5333	0.4000	0.3667	1.3833	0.5667	1.1000	0.8167	1.1333
21	0.5500	0.7167	0.7667	1.3500	0.6333	1.0000	0.2500	1.0833
22	0.5500	0.6667	0.6167	1.1167	0.8333	0.2833	0.6000	0.8333

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
23	0.7500	0.8500	0.8000	1.2667	1.0167	0.3167	0.7833	1.0167
24	-	0.1667	0.5167	1.1333	0.4167	0.7833	0.3333	0.8667
25	0.1667	-	0.3500	1.2833	0.6167	0.9333	0.5333	1.0167
26	0.5167	0.3500	-	1.1833	0.6833	0.8500	0.5667	0.9167
27	1.1333	1.2833	1.1833	-	1.4000	1.3500	1.1500	0.2500
28	0.4167	0.6167	0.6833	1.4000	-	1.0833	0.5833	1.1500
29	0.7833	0.9333	0.8500	1.3500	1.0833	-	0.8333	1.0833
30	0.3333	0.5333	0.5667	1.1500	0.5833	0.8333	-	0.8833
31	0.8667	1.0167	0.9167	0.2500	1.1500	1.0833	0.8833	-
32	0.2333	0.4500	0.6167	1.2167	0.2667	0.9000	0.4333	0.9500
33	0.4000	0.6000	0.4667	0.7667	0.7000	0.6333	0.4167	0.5000

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
34	0.3000	0.4500	0.4167	0.9667	0.6000	0.6000	0.3000	0.7000
35	0.3500	0.3333	0.2500	0.9833	0.6500	0.6667	0.3833	0.7167
36	1.1667	1.0167	0.9833	1.9833	1.2500	1.7500	1.4667	1.7167
37	1.0333	0.8833	0.8833	1.8833	0.9333	1.6167	1.3333	1.6167
38	1.6000	1.4500	1.4500	2.4500	1.5000	2.1833	1.9000	2.1833
39	1.8333	1.6833	1.6833	2.6833	1.7333	2.4167	2.1333	2.4167
40	1.4000	1.2500	1.2500	2.2500	1.3000	1.9833	1.6833	1.9833
41	1.8667	1.7833	1.7500	2.7500	2.0167	2.5167	2.2333	2.4833
42	1.2833	1.1500	1.1167	2.1000	1.3667	1.8667	1.5833	1.8500

From Ci	To Cj 32	33	34	35	36	37	38	39
0	0.4833	0.5833	0.5333	0.3500	0.8833	0.7500	1.3167	1.5500
1	1.0833	0.7333	0.9000	0.7167	1.5333	1.4333	2.0000	2.2333
2	0.7667	0.9667	0.9167	0.7333	0.7000	0.3500	0.9167	1.1500
3	1.2000	1.2333	1.1833	1.0000	0.5333	0.9167	1.5000	1.7333
4	0.6667	0.7000	0.6500	0.4667	0.7167	0.6833	1.2500	1.4833
5	0.7500	0.7667	0.7167	0.5500	0.9500	0.8667	1.4500	1.6833
6	0.7000	0.7167	0.6667	0.4833	0.9000	0.7667	1.3333	1.5667
7	0.6000	0.6167	0.5000	0.5833	1.6667	1.4833	1.9333	2.1667
8	0.6667	0.3833	0.3500	0.4333	1.5167	1.3833	1.9333	2.1667
9	0.8500	0.5667	0.5333	0.6167	1.7000	1.5667	2.1167	2.3667
10	0.2333	0.4000	0.3000	0.3500	1.1667	1.0333	1.6000	1.8333

From Ci	To Cj 32	33	34	35	36	37	38	39
11	0.7667	0.9667	0.9167	0.7333	0.7000	0.3500	0.9167	1.1500
12	1.2000	1.2333	1.1833	1.0000	0.5333	0.9167	1.5000	1.7333
13	0.7500	0.7667	0.7167	0.5500	0.9500	0.8667	1.4500	1.6833
14	0.6667	0.3833	0.3500	0.4333	1.5167	1.3833	1.9333	2.1667
15	1.0833	0.7333	0.9000	0.7167	1.5333	1.4333	2.0000	2.2333
16	0.7667	0.9667	0.9167	0.7333	0.7000	0.3500	0.9167	1.1500
17	1.2000	1.2333	1.1833	1.0000	0.5333	0.9167	1.5000	1.7333
18	0.6667	0.7000	0.6500	0.4667	0.7167	0.6833	1.2500	1.4833
19	0.7500	0.7667	0.7167	0.5500	0.9500	0.8667	1.4500	1.6833
20	0.7000	0.7167	0.6667	0.4833	0.9000	0.7667	1.3333	1.5667
21	0.6000	0.6167	0.5000	0.5833	1.6667	1.4833	1.9333	2.1667
22	0.6667	0.3833	0.3500	0.4333	1.5167	1.3833	1.9333	2.1667

From Ci	To Cj							
	32	*33*	*34*	*35*	*36*	*37*	*38*	*39*
23	0.8500	0.5667	0.5333	0.6167	1.7000	1.5667	2.1167	2.3667
24	0.2333	0.4000	0.3000	0.3500	1.1667	1.0333	1.6000	1.8333
25	0.4500	0.6000	0.4500	0.3333	1.0167	0.8833	1.4500	1.6833
26	0.6167	0.4667	0.4167	0.2500	0.9833	0.8833	1.4500	1.6833
27	1.2167	0.7667	0.9667	0.9833	1.9833	1.8833	2.4500	2.6833
28	0.2667	0.7000	0.6000	0.6500	1.2500	0.9333	1.5000	1.7333
29	0.9000	0.6333	0.6000	0.6667	1.7500	1.6167	2.1833	2.4167
30	0.4333	0.4167	0.3000	0.3833	1.4667	1.3333	1.9000	2.1333
31	0.9500	0.5000	0.7000	0.7167	1.7167	1.6167	2.1833	2.4167
32	-	0.5000	0.4000	0.4500	1.3667	1.0333	1.6000	1.8333
33	0.5000	-	0.2667	0.2833	1.3667	1.2333	1.8000	2.0333

	To Cj							
From Ci	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*
34	0.4000	0.2667	-	0.2333	1.2833	1.1500	1.7167	1.9500
35	0.4500	0.2833	0.2333	-	1.1333	1.0000	1.5667	1.8000
36	1.3667	1.3667	1.2833	1.1333	-	0.5833	1.1667	1.4000
37	1.0333	1.2333	1.1500	1.0000	0.5833	-	0.6333	0.8667
38	1.6000	1.8000	1.7167	1.5667	1.1667	0.6333	-	0.4000
39	1.8333	2.0333	1.9500	1.8000	1.4000	0.8667	0.4000	-
40	1.4000	1.6000	1.5167	1.3667	0.9500	0.4333	0.7667	0.4667
41	1.1167	2.1333	2.0500	1.9000	1.1667	1.4167	1.4333	1.0333
42	1.4833	1.4833	1.4167	1.2500	0.6000	0.7667	1.3000	0.8833

	To Cj		
From Ci	*40*	*41*	*42*
0	1.1167	1.6500	1.0000
1	1.8000	2.3000	1.6667
2	0.7000	1.4667	0.8167
3	1.2833	1.5000	0.9333
4	1.0500	1.4833	0.8333
5	1.2333	1.7500	1.1000
6	1.1333	1.6667	1.0333
7	1.7333	2.4167	1.7833
8	1.7333	2.2667	1.6333
9	1.9167	2.4667	1.8167
10	1.4000	1.8667	1.2833

	To Cj		
From Ci	*40*	*41*	*42*
11	0.7000	1.4667	0.8167
12	1.2833	1.5000	0.9333
13	1.2333	1.7500	1.1000
14	1.7333	2.2667	1.6333
15	1.8000	2.3000	1.6667
16	0.7000	1.4667	0.8167
17	1.2833	1.5000	0.9333
18	1.0500	1.4833	0.8333
19	1.2333	1.7500	1.1000
20	1.1333	1.6667	1.0333
21	1.7333	2.4167	1.7833

	To Cj		
From Ci	*40*	*41*	*42*
22	1.7333	2.2667	1.6333
23	1.9167	2.4667	1.8167
24	1.4000	1.8667	1.2833
25	1.2500	1.7833	1.1500
26	1.2500	1.7500	1.1167
27	2.2500	2.7500	2.1000
28	1.3000	2.0167	1.3667
29	1.9833	2.5167	1.8667
30	1.6833	2.2333	1.5833
31	1.9833	2.4833	1.8500
32	1.4000	1.1167	1.4833

	To Cj		
From Ci	*40*	*41*	*42*
33	1.6000	2.1333	1.4833
34	1.5167	2.0500	1.4167
35	1.3667	1.9000	1.2500
36	0.9500	1.1667	0.6000
37	0.4333	1.4167	0.7667
38	0.7667	1.4333	1.3000
39	0.4667	1.0333	0.8833
40	-	1.3500	0.7000
41	1.3500	-	0.6833
42	0.7000	0.6833	-

B.5. Travel Distance Between Locations

	To Cj							
From Ci	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
0	0	37	26	42	8	12	7	51
1	37	0	58	55	40	28	40	57
2	26	58	0	43	22	35	26	51
3	42	55	43	0	44	30	43	88
4	8	40	22	44	0	15	8	58
5	12	28	35	30	15	0	15	60
6	7	40	26	43	8	15	0	56
7	51	57	51	88	58	60	56	0
8	53	46	68	81	50	52	49	45
9	50	50	75	87	57	58	55	52
10	21	46	46	61	28	32	26	23

	To Cj							
From Ci	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
11	26	58	0	43	22	35	26	51
12	42	55	43	0	44	30	43	88
13	12	28	35	30	15	0	15	60
14	53	46	68	81	50	52	49	45
15	37	0	58	55	40	28	40	57
16	26	58	0	43	22	35	26	51
17	42	55	43	0	44	30	43	88
18	8	40	22	44	0	15	8	58
19	12	28	35	30	15	0	15	60
20	7	40	26	43	8	15	0	56
21	51	57	51	88	58	60	56	0
22	53	46	68	81	50	52	49	45

	To Cj							
From Ci	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
23	50	50	75	87	57	58	55	52
24	21	46	46	61	28	32	26	23
25	12	35	38	53	20	24	18	29
26	10	28	34	46	16	17	15	44
27	63	14	70	69	52	45	51	65
28	26	60	27	63	31	34	26	30
29	57	59	82	94	64	66	62	59
30	39	46	64	77	46	48	44	12
31	55	24	68	80	50	45	50	57
32	17	54	35	58	25	29	22	25
33	32	29	57	70	39	41	38	34

From Ci	To Cj 0	1	2	3	4	5	6	7
34	29	33	54	67	36	38	34	26
35	20	31	45	58	27	29	26	32
36	44	75	32	27	36	56	43	93
37	44	76	20	58	40	53	44	68
38	78	109	53	92	73	86	77	101
39	86	118	61	100	82	95	86	110
40	64	96	40	79	60	73	64	88
41	92	123	80	80	84	98	92	141
42	58	89	46	57	50	64	58	107

	To Cj							
From Ci	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
0	43	50	21	26	42	12	43	37
1	46	50	46	58	55	28	46	0
2	68	75	46	0	43	35	68	58
3	81	87	61	43	0	30	81	55
4	50	57	28	22	44	15	50	40
5	52	58	32	35	30	0	52	28
6	49	55	26	26	43	15	49	40
7	45	52	23	51	88	60	45	57
8	0	7	32	68	81	52	0	46
9	7	0	28	75	87	58	7	50
10	32	28	0	46	61	32	32	46

From Ci	To Cj 8	9	10	11	12	13	14	15
11	68	75	46	0	43	35	68	58
12	81	87	61	43	0	30	81	55
13	52	58	32	35	30	0	52	28
14	0	7	32	68	81	52	0	46
15	46	50	46	58	55	28	46	0
16	68	75	46	0	43	35	68	58
17	81	87	61	43	0	30	81	55
18	50	57	28	22	44	15	50	40
19	52	58	32	35	30	0	52	28
20	49	55	26	26	43	15	49	40
21	45	52	23	51	88	60	45	57
22	0	7	32	68	81	52	0	46

	To Cj							
From Ci	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
23	7	0	28	75	87	58	7	50
24	32	28	0	46	61	32	32	46
25	35	42	5	38	53	24	35	35
26	36	42	28	34	46	17	36	28
27	50	45	51	70	69	45	50	14
28	52	58	23	27	63	34	52	60
29	15	11	46	82	94	66	15	59
30	34	41	13	64	77	48	34	46
31	43	37	44	68	80	45	43	24
32	39	45	11	35	58	29	39	54
33	19	26	20	57	70	41	19	29

	To Cj							
From Ci	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
34	20	26	14	54	67	38	20	33
35	24	30	17	45	58	29	24	31
36	86	92	63	32	27	56	86	75
37	86	93	64	20	58	53	86	76
38	119	126	97	53	92	86	119	109
39	128	134	105	61	100	95	128	118
40	106	112	84	40	79	73	106	96
41	134	140	111	80	80	98	134	123
42	100	106	77	46	57	64	100	89

From Ci	To Cj 16	17	18	19	20	21	22	23
0	26	42	8	12	7	51	43	50
1	58	55	40	28	40	57	46	50
2	0	43	22	35	26	51	68	75
3	43	0	44	30	43	88	81	87
4	22	44	0	15	8	58	50	57
5	35	30	15	0	15	60	52	58
6	26	43	8	15	0	56	49	55
7	51	88	58	60	56	0	45	52
8	68	81	50	52	49	45	0	7
9	75	87	57	58	55	52	7	0
10	46	61	28	32	26	23	32	28

	To Cj							
From Ci	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*
11	0	43	22	35	26	51	68	75
12	43	0	44	30	43	88	81	87
13	35	30	15	0	15	60	52	58
14	68	81	50	52	49	45	0	7
15	58	55	40	28	40	57	46	50
16	0	43	22	35	26	51	68	75
17	43	0	44	30	43	88	81	87
18	22	44	0	15	8	58	50	57
19	35	30	15	0	15	60	52	58
20	26	43	8	15	0	56	49	55
21	51	88	58	60	56	0	45	52
22	68	81	50	52	49	45	0	7

	To Cj							
From Ci	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*
23	75	87	57	58	55	52	7	0
24	46	61	28	32	26	23	32	28
25	38	53	20	24	18	29	35	42
26	34	46	16	17	15	44	36	42
27	70	69	52	45	51	65	50	45
28	27	63	31	34	26	30	52	58
29	82	94	64	66	62	59	15	11
30	64	77	46	48	44	12	34	41
31	68	80	50	45	50	57	43	37
32	35	58	25	29	22	25	39	45
33	57	70	39	41	38	34	19	26

	To Cj							
From Ci	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*
34	54	67	36	38	34	26	20	26
35	45	58	27	29	26	32	24	30
36	32	27	36	56	43	93	86	92
37	20	58	40	53	44	68	86	93
38	53	92	73	86	77	101	119	126
39	61	100	82	95	86	110	128	134
40	40	79	60	73	64	88	106	112
41	80	80	84	98	92	141	134	140
42	46	57	50	64	58	107	100	106

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
0	21	13	10	63	26	57	37	55
1	46	35	28	14	60	59	46	24
2	46	38	34	70	27	82	64	68
3	61	53	46	69	63	94	77	80
4	28	20	16	52	31	64	46	50
5	32	24	17	45	34	66	48	45
6	26	18	15	51	26	62	44	50
7	23	29	44	65	30	59	12	57
8	32	35	36	50	52	15	34	43
9	28	42	42	45	58	11	41	37
10	0	5	28	51	23	46	13	44

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
11	46	38	34	70	27	82	64	68
12	61	53	46	69	63	94	77	80
13	32	24	17	45	34	66	48	45
14	32	35	36	50	52	15	34	43
15	46	35	28	14	60	59	46	24
16	46	38	34	70	27	82	64	68
17	61	53	46	69	63	94	77	80
18	28	20	16	52	31	64	46	50
19	32	24	17	45	34	66	48	45
20	26	18	15	51	26	62	44	50
21	23	29	44	65	30	59	12	57
22	32	35	36	50	52	15	34	43

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
23	28	42	42	45	58	11	41	37
24	0	5	28	51	23	46	13	44
25	5	0	12	57	29	52	12	50
26	28	12	0	40	36	50	32	48
27	51	57	40	0	71	65	56	7
28	23	29	36	71	0	67	31	64
29	46	52	50	65	67	0	48	56
30	13	12	32	56	31	48	0	46
31	44	50	48	7	64	56	46	0
32	11	11	35	59	14	53	18	51
33	20	27	25	31	41	33	23	24

	To Cj							
From Ci	*24*	*25*	*26*	*27*	*28*	*29*	*30*	*31*
34	14	13	22	39	35	34	14	32
35	17	10	13	43	38	38	21	36
36	63	56	51	52	66	99	82	85
37	64	56	52	93	42	100	83	86
38	97	90	85	126	76	113	116	119
39	105	98	94	135	84	141	124	127
40	84	76	72	113	62	120	102	105
41	111	104	99	140	114	147	130	133
42	77	70	65	106	80	113	96	99

	To Cj							
From Ci	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*
0	17	32	29	20	44	44	78	86
1	54	29	33	31	75	76	109	118
2	35	57	54	45	32	20	53	61
3	58	70	67	58	27	58	92	100
4	25	39	36	27	36	40	73	82
5	29	41	38	29	56	53	86	95
6	22	38	34	26	43	44	77	86
7	25	34	26	32	93	68	101	110
8	39	19	20	24	86	86	119	128
9	45	26	26	30	92	93	126	134
10	11	20	14	17	63	64	97	105

	To Cj							
From Ci	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*
11	35	57	54	45	32	20	53	61
12	58	70	67	58	27	58	92	100
13	29	41	38	29	56	53	86	95
14	39	19	20	24	86	86	119	128
15	54	29	33	31	75	76	109	118
16	35	57	54	45	32	20	53	61
17	58	70	67	58	27	58	92	100
18	25	39	36	27	36	40	73	82
19	29	41	38	29	56	53	86	95
20	22	38	34	26	43	44	77	86
21	25	34	26	32	93	68	101	110
22	39	19	20	24	86	86	119	128

From Ci	To Cj 32	33	34	35	36	37	38	39
23	45	26	26	30	92	93	126	134
24	11	20	14	17	63	64	97	105
25	11	27	13	10	56	56	90	98
26	35	25	22	13	51	52	85	94
27	59	31	39	43	52	93	126	135
28	14	41	35	38	66	42	76	84
29	53	33	34	38	99	100	113	141
30	18	23	14	21	82	83	116	124
31	51	24	32	36	85	86	119	127
32	0	28	22	25	61	52	85	94
33	28	0	9	13	75	76	109	117

	To Cj							
From Ci	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*
34	22	9	0	11	71	72	105	114
35	25	13	11	0	63	63	96	105
36	61	75	71	63	0	34	68	76
37	52	76	72	63	34	0	35	43
38	85	109	105	96	68	35	0	16
39	94	117	114	105	76	43	16	0
40	72	95	92	83	54	21	29	22
41	109	123	119	111	57	80	66	50
42	75	89	85	77	33	46	50	34

	To Cj		
From Ci	*40*	*41*	*42*
0	64	92	58
1	96	123	89
2	40	80	46
3	79	80	57
4	60	84	50
5	73	98	64
6	64	92	58
7	88	141	107
8	106	134	100
9	112	140	106
10	84	111	77

	To Cj		
From Ci	*40*	*41*	*42*
11	40	80	46
12	79	80	57
13	73	98	64
14	106	134	100
15	96	123	89
16	40	80	46
17	79	80	57
18	60	84	50
19	73	98	64
20	64	92	58
21	88	141	107

	To Cj		
From Ci	*40*	*41*	*42*
22	106	134	100
23	112	140	106
24	84	111	77
25	76	104	70
26	72	99	65
27	113	140	106
28	62	114	80
29	120	147	113
30	102	130	96
31	105	133	99
32	72	109	75

	To Cj		
From Ci	*40*	*41*	*42*
33	95	123	89
34	92	119	85
35	83	111	77
36	54	57	33
37	21	80	46
38	29	66	50
39	22	50	34
40	0	55	21
41	55	0	35
42	21	35	0

Appendix C: Schedules Associated with Best Solutions for Different Routing Problems

C.1. PVRPB Solutions Obtained Using Basic Model Formulation

Test Scenario #1

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	1	4	5	0		
Tuesday	1	1	3	5	4	0
Friday	6	2	4	5	0	

Test Scenario #2

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	1	6	5	0		
Thursday	1	1	3	5	6	0
Friday	2	2	4	6	5	0

Test Scenario #3

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	4	5	6	7	0	
Wednesday	4	3	1	7	6	0
Thursday	7	4	2	7	6	0

Test Scenario #4

		Tour Stop					
Weekday	**Vehicle**	1	2	3	4	5	6
Wednesday	7	5	8	7	0		
Thursday	7	3	1	8	7	0	
Friday	3	6	4	2	7	8	0

Test Scenario #5

		Tour Stop					
Weekday	**Vehicle**	1	2	3	4	5	6
Monday	1	3	1	8	9	0	
Wednesday	6	6	4	5	8	9	0
Thursday	6	2	7	8	9	0	

Test Scenario #6

		Tour Stop					
Weekday	**Vehicle**	1	2	3	4	5	6
Monday	7	6	4	1	9	10	0
Tuesday	6	2	7	9	10	0	
Wednesday	2	8	9	10	0		
	3	5	3	0			

Test Scenario #7

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	1	5	1	15	0			
	3	10	7	14	11	12	13	0
Tuesday	3	6	9	8	14	15	0	
	7	4	2	11	12	13	0	
Friday	4	3	12	13	15	14	11	0

Test Scenario #8

		Tour Stop								
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9
Monday	2	1	14	15	20	11	0			
	5	3	17	12	13	16	18	0		
Tuesday	3	4	2	11	16	20	14	15	0	
	4	10	7	13	0					
Wednesday	4	18	20	19	15	14	12	11	16	0
Thursday	7	5	19	13	17	0				
Friday	5	6	8	9	17	12	18	19	0	

Test Scenario #9

		Tour Stop								
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9
Monday	5	1	15	19	0					
	6	24	22	23	21	16	17	18	20	0
Tuesday	4	9	23	22	15	25	0			
Wednesday	2	10	7	21	22	15	25	0		
	6	13	5	4	18	20	25	0		
Thursday	3	6	2	11	12	17	19	0		
	4	14	8	23	24	18	16	0		
Friday	3	3	16	21	24	25	20	0		

C.2. PVRPB Solutions Obtained by Adding Tour Limitation Constraints to Model Formulation

Test Scenario #1

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	3	5	4	0
Tuesday	1	1	5	4	0
Friday	1	2	4	5	0

Test Scenario #2

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	1	3	5	6	0	
Tuesday	4	4	6	0		
Thursday	1	1	5	0		
Friday	4	2	6	5	0	

Test Scenario #3

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	3	6	7	0
Tuesday	6	1	7	0	
Wednesday	1	2	7	0	
Thursday	7	5	6	0	
Friday	3	4	6	0	

Test Scenario #4

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	6	0		
	2	1	8	7	0
Tuesday	1	3	8	7	0
	5	5	0		
Thursday	3	4	0		
Friday	6	2	7	8	0

Test Scenario #5

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	2	0		
	2	7	8	9	0
	7	6	0		
Wednesday	3	4	0		
Thursday	1	1	8	9	0
Friday	2	3	8	9	0
	6	5	0		

Test Scenario #6

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	1	0		
	4	3	9	10	0
	5	6	0		
Tuesday	4	8	9	10	0
Wednesday	4	4	0		
	5	7	9	10	0
Thursday	5	5	0		
Friday	3	2	0		

Test Scenario # 7

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	2	7	14	0		
Tuesday	4	9	14	15	0	
Wednesday	4	1	15	0		
	6	5	13	12	11	0
Thursday	1	2	11	12	13	0
	5	4	0			
	7	6	0			
Friday	1	8	14	15	13	0
	5	10	0			
	7	3	12	11	0	

Test Scenario #8

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
	1	10	0					
Monday	3	8	14	15	19	13	0	
	5	6	20	0				
Tuesday	2	4	18	20	0			
	5	9	14	15	19	13	0	
Wednesday	6	3	17	12	16	11	18	0
Thursday	5	1	15	17	12	16	11	0
	1	7	14	20	0			
Friday	4	5	13	19	12	17	0	
	7	2	11	16	18	0		

Test Scenario #9

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	1	2	16	18	20	0		
	3	8	22	23	15	0		
	6	12	17	19	0			
Tuesday	5	7	21	24	25	0		
	6	5	19	0				
	7	11	16	17	0			
Wednesday	1	6	20	0				
	2							
	3	10	21	24	25	0		
	6	3	17	16	18	0		
Thursday	2	4	18	20	0			
	6	14	22	23	15	0		

Test Scenario #9 (cont'd.)

Friday	4	13	19	0				
	7	9	23	22	21	24	25	0

Test Scenario #10

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	2	5	19	17	16	18	-	
	5	8	22	29	23	24	25	-
Tuesday	4	6	20	-				
	7	10	24	30	21	28	-	
Wednesday	5	3	17	19	26	0		
	6	2	16	18	20	0		

Test Scenario #10 (cont'd.)

Thursday	2	12	17	26	0			
	5	7	21	30	25	24	28	0
	6	9	23	29	22	27	15	0
Friday	3	11	16	28	21	30	25	0
	4	13	19	0				
	5	4	18	20	0			
	6	14	22	29	23	27	15	0

Test Scenario #11

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	4	7	21	29	22	24	25	0
	7	4	26	15	31	33	35	0

Test Scenario #11 (cont'd.)

Tuesday	3	3	16	18	26	0		
	4	9	23	34	30	28	0	
	3	10	24	30	21	16	18	0
Wednesday	5	12	17	31	27	15	20	0
	7	8	22	35	29	33	19	0
	1	6	20	25	0			
Thursday	2	14	22	23	35	24	32	0
	7	3	17	19	34	27	31	0
Friday	2	11	28	29	23	34	32	0
	6	5	16	17	19	18	20	0

C.3. PVRPB Solutions Obtained Using MULTI-HGA-ROUTE

Test Scenario #1

		Tour Stops			
Weekday	**Vehicle**	1	2	3	4
Tuesday	2	2	4	5	0
Wednesday	5	3	5	4	0
Thursday	1	1	5	4	

Test Scenario #2

		Tour Stops				
Weekday	**Vehicle**	1	2	3	4	5
Tuesday	2	1	5	6	0	
Thursday	7	2	6	5	0	
Friday	4	4	3	5	6	0

Test Scenario #3

		Tour Stops				
Weekday	**Vehicle**	1	2	3	4	5
Tuesday	4	3	4	6	7	0
Wednesday	7	2	1	7	6	0
Friday	7	5	6	7	0	

Test Scenario #4

		Tour Stops				
Weekday	**Vehicle**	1	2	3	4	5
Tuesday	2	2	1	8	7	0
Wednesday	6	6	5	8	7	0
Friday	6	4	3	8	7	0

Test Scenario #5

		Tour Stops					
Weekday	**Vehicle**	1	2	3	4	5	6
Wednesday	3	6	5	8	9	0	
Thursday	2	3	1	8	9	0	
	4	4	2	0			
Friday	6	7	8	9	0		

Test Scenario #6

		Tour Stops				
Weekday	**Vehicle**	1	2	3	4	5
Monday	6	4	3	9	10	0
Tuesday	3	6	5	0		
Thursday	1	2	1	9	10	0

Test Scenario #6 (cont'd.)

Friday	5	7	8	9	10	0

Test Scenario #7

		Tour Stops							
Weekday	**Vehicle**	1	2	3	4	5	6	7	8
	1	5	1	15	12	13	0		
Monday	2	6	2	11	0				
	7	10	9	14	0				
Thursday	2	4	3	12	11	14	15	13	0
Friday	1	7	8	14	15	13	12	11	0

Test Scenario #8

		Tour Stops								
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9
Monday	1	2	1	15	14	0				
	2	20	18	16	11	12	17	19	13	0
Wednesday	1	8	7	0						
Thursday	5	3	17	12	11	16	0			
	7	4	18	20	14	15	19	13	0	
Friday	2	5	6	20	18	16	11	0		
	5	10	9	14	15	12	17	13	19	0

Test Scenario #9

		Tour Stops								
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9
Monday	2	6	5	19	15	17	16	18	20	0
	6	8	7	21	22	23	24	25	0	
Tuesday	4	20	18	16	17	19	0			
	6	10	9	23	22	15	21	24	25	0
Wednesday	1	13	19	0						
	5	1	14	22	23	0				
	6	2	21	24	25	20	0			
Thursday	4	11	12	17	16	18	15	0		
Friday	5	3	4	0						

Test Scenario #10

		Tour Stops										
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10	11
	3	13	3	17	16	0						
Monday	4	4	18	20	28	21	30	24	25	26	19	0
	7	14	22	29	23	27	15	0				
Tuesday	3	7	8	29	27	30	28	0				
Wednesday	6	6	5	19	15	17	16	18	20	0		
Thursday	4	10	9	23	22	21	24	25	26	0		
	2	20	18	0								
Friday	3	2	11	16	28	0						
	4	12	17	19	26	25	24	0				
	5	1	15	27	23	29	22	21	30	0		

Test Scenario #11

		Tour Stops									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Monday	1	14	22	29	23	31	27	33	34	35	0
	5	6	20	18	19	26	0				
	7	11	28	21	30	25	24	32	0		
Tuesday	2	13	3	17	0						
	3	16	21	25	24	23	22	15	26	0	
Wednesday	3	4	18	19	20	28	30	32	0		
	7	10	8	29	33	27	31	34	35	0	
Thursday	1	9	7	21	22	23	15	0			
	2	12	17	16	0						
Friday	3	20	18	17	19	26	35	0			
	4	5	1	15	27	31	33	29	34	0	
	5	2	16	28	32	30	24	25	0		

Test Scenario #12

		Tour Stops										
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10	11
	5	3	17	36	37	38	16	0				
Monday	6	4	18	0								
	7	9	33	34	35	15	19	20	0			
	2	11	21	34	33	35	25	26	0			
Tuesday	5	37	40	38	39	41	42	36	0			
	6	8	22	23	24	0						
	1	32	28	21	30	34	25	0				
Wednesday	3	5	6	38	37	36	0					
	4	10	24	22	29	23	31	27	33	35	26	0
	1	2	16	40	39	41	42	19	0			
Thursday	2	14	29	28	20	18	0					
	6	13	12	17	15	27	31	30	32	0		

Test Scenario #12 (cont'd.)

Friday	1	19	18	42	41	39	40	0	
	3	26	25	32	28	0			
	6	1	31	27	15	17	16	20	0
	7	7	21	30	22	29	23	24	0

C.4. PVRPBTW Solutions Obtained Using Basic Model Formulation

Test Scenario #1

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	2	2	3	0	
Tuesday	1	1	4	5	0
Wednesday	7	4	5	0	
Friday	1	4	5	0	

Test Scenario #2

		Vehicle Stop		
Weekday	**Vehicle**	1	2	3
	1	6	5	0
Monday	3	1	3	0
	4	4	2	0
Tuesday	6	6	5	0
Wednesday	1	6	5	0

Test Scenario #3

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	6	7	0	
Tuesday	1	5	6	7	0
	2	1	3	0	
Thursday	3	6	7	0	
Friday	4	4	2	0	

Test Scenario #4

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
	1	7	8	0	
Monday	6	1	3	0	
	7	6	4	2	0
Tuesday	1	5	7	8	0
Wednesday	4	7	8	0	

Test Scenario #5

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	5	7	8	9	0	
Tuesday	1	8	9	0		
	1	5	1	8	9	0
Thursday	5	2	3	0		
	7	6	4	0		

Test Scenario #6

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	6	6	4	10	9	0
Wednesday	3	7	8	9	10	0
Friday	2	5	1	9	10	0
	3	2	3	0		

Test Scenario #7

		Vehicle Stop					
Weekday	**Vehicle**	1	2	3	4	5	6
Monday	5	9	14	13	0		
Tuesday	6	5	1	13	0		
	2	3	12	0			
Wednesday	3	8	14	15	0		
	4	6	4	2	11	0	
Thursday	1	10	7	14	11	12	0
Friday	1	11	12	13	15	0	

Test Scenario #8

		Vehicle Stop								
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9
Monday	4	6	20	19	13	0				
	5	4	2	11	16	18	0			
	6	1	3	17	12	0				
Tuesday	4	12	17	16	11	18	20	0		
	6	9	8	14	15	0				
Wednesday	7	5	13	19	14	15	0			
Friday	3	20	19	13	17	12	11	16	18	0
	4	10	7	14	15	0				

Test Scenario #9

		Vehicle Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	1	9	14	22	23	21	0	
	4	5	19	17	15	18	20	0
Tuesday	4	4	1	15	0			
	5	6	20	16	18	22	23	0
Wednesday	2	10	7	25	24	23	0	
Thursday	2	13	19	17	16	18	0	
	5	25	24	21	0			
Friday	1	11	2	16	20	0		
	2	19	25	24	21	0		
	4	8	22	15	0			
	6	12	3	17	0			

C.5. PVRPBTW Solutions Obtained Adding Tour Limitation Constraints to Model Formulation

Test Scenario #1

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	4	5	0	
Tuesday	3	2	0		
Wednesday	5	3	0		
Thursday	6	4	5	0	
Friday	4	1	4	5	0

Test Scenario #2

		Vehicle Stop		
Weekday	**Vehicle**	1	2	3
Monday	5	6	0	
	6	4	5	0
Tuesday	1	1	5	0
	2	6	0	
	3	2	0	
Wednesday	5	3	0	
Thursday	1	6	5	0

Test Scenario #3

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
	1	3	0		
Monday	4	4	0		
	6	6	7	0	
Tuesday	6	5	6	7	0
Wednesday	5	1	0		
Thursday	5	5	6	7	0
Friday	7	2	0		

Test Scenario #4

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	4	0		
	3	7	8	0	
Tuesday	1	2	0		
Wednesday	5	1	0		
	7	6	7	8	0
Thursday	7	3	0		
Friday	4	5	7	8	0

Test Scenario #5

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	5	8	9	0
Tuesday	1	1	8	9	0
	3	4	0		
Wednesday	5	3	0		
Thursday	1	7	8	9	0
	6	6	0		
Friday	5	2	0		

Test Scenario #6

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	1	7	9	10	0

Test Scenario #6 (cont'd.)

Tuesday	1	5	0
Wednesday	1	8	9
	4	2	0
Thursday	2	3	0
	3	4	0
Friday	1	1	9
	3	6	0

Test Scenario #7

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	1	6	13	12	11	0
	3	7	14	15	0	

Test Scenario #7 (cont'd.)

	3	9	14	13	0	
Tuesday	6	10	0			
	7	2	11	0		
Thursday	7	5	13	12	11	0
	1	4	0			
Friday	2	8	14	15	0	
	7	3	12	0		

Test Scenario #8

		Vehicle Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	4	5	19	13	12	17	0	
Tuesday	1	4	18	16	11	0		
	2	6	20	13	19	0		
Wednesday	3	3	17	12	0			
	5	10	14	15	0			
Thursday	2	8	14	15	0			
	5	7	16	11	18	20	0	
Friday	1	9	14	13	19	17	12	0
	4	2	11	16	18	20	0	

Test Scenario #9

		Vehicle Stop					
Weekday	**Vehicle**	1	2	3	4	5	6
Monday	1	4	18	0			
	2	3	17	0			
	3	1	19	25	24	20	0
	6	14	22	23	21	0	
Tuesday	2	2	16	18	0		
	4	10	24	25	15	0	
Wednesday	1	5	19	15	0		
	2	7	21	0			
	5	12	17	16	18	20	0
Thursday	1	8	22	23	21	0	

Test Scenario #9 (cont'd.)

	1	9	23	0			
	2	6	20	15	0		
Friday	4	13	24	25	0		
	6	11	22	19	17	16	0

C.6. PVRPBTW Solutions Obtained Using MULTI-HGA-ROUTE

Test Scenario #1

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Tuesday	6	5	4	0	
Wednesday	1	1	5	4	0
	6	2	3	0	
Friday	2	5	4	0	

Test Scenario #2

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	4	1	5	0		
Tuesday	4	4	3	5	6	0
Wednesday	5	2	6	5	0	
Friday	7	6	0			

Test Scenario #3

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	5	5	4	6	7	0

Test Scenario #3 (cont'd.)

Tuesday	5	1	0	
Wednesday	5	3	6	0
	7	7	0	
Thursday	5	2	7	6

Test Scenario #4

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Wednesday	3	7	8	0	
Thursday	7	7	8	0	
	2	1	3	0	
Friday	6	6	4	2	0
	7	5	7	8	0

Test Scenario #5

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	2	8	9	0		
	3	5	1	8	9	0
Tuesday	5	2	3	0		
	6	6	4	0		
Thursday	6	7	8	9	0	

Test Scenario #6

		Vehicle Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	3	4	2	10	9	0
Wednesday	2	7	8	9	10	0
	7	1	3	0		
Thursday	6	5	6	10	9	0

Test Scenario #7

		Vehicle Stop					
Weekday	**Vehicle**	1	2	3	4	5	6
Monday	2	7	8	14	15	0	
	5	11	12	13	0		
Tuesday	2	10	9	14	15	0	
	6	6	5	13	12	11	0
Thursday	3	1	2	11	12	0	
Friday	2	13	15	14	0		
	4	4	3	0			

Test Scenario #8

		Vehicle Stop									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Monday	4	11	12	13	0						
	5	7	8	14	0						
Tuesday	3	5	19	17	15	20	0				
	6	2	16	18	0						
Wednesday	4	1	15	14	0						
	7	11	12	13	0						
Thursday	1	10	9	14	15	0					
	5	6	20	19	13	12	17	16	11	18	0
Friday	1	19	20	16	18	0					
	5	4	3	17	0						

Test Scenario #9

		Vehicle Stop							
Weekday	**Vehicle**	1	2	3	4	5	6	7	8
Monday	2	20	18	16	17	19	15	0	
	6	10	9	23	22	21	24	25	0
Tuesday	1	13	3	12	17	19	15	0	
	4	14	22	23	21	24	25	0	
	7	4	11	16	18	20	0		
Wednesday	2	2	6	20	19	0			
	7	5	1	23	22	21	24	25	0
Thursday	3	7	8	15	17	16	18	0	

Test Scenario #10

		Vehicle Stop									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Monday	2	26	23	22	21	24	25	0			
	1	4	3	17	19	0					
Tuesday	2	20	18	16	28	24	21	30	25	26	0
	7	10	9	23	29	22	27	15	0		
	2	13	19	17	16	18	20	0			
Wednesday	4	14	8	29	30	27	15	0			
	6	7	28	0							
Thursday	6	6	5	26	23	22	21	24	25	0	
	1	11	2	16	18	20	0				
Friday	2	12	17	19	0						
	3	1	15	27	29	30	28	0			

Test Scenario #11

		Vehicle Stop							
Weekday	**Vehicle**	1	2	3	4	5	6	7	8
Monday	4	1	15	31	27	0			
	5	14	8	33	29	0			
	6	30	34	35	0				
	7	32	28	16	17	0			
Tuesday	2	34	30	21	33	31	27	0	
	3	11	28	32	24	25	0		
	5	20	18	0					
	7	19	26	35	22	23	29	0	
Wednesday	2	6	28	24	21	33	31	27	0
	3	32	30	35	25	0			
	4	13	2	0					
	6	26	34	22	23	29	0		

Test Scenario #11 (cont'd.)

	3	5	3	17	0			
Thursday	4	20	16	18	19	15	0	
	5	9	22	23	21	0		
	1	4	18	16	20	0		
Friday	4	12	17	19	0			
	6	10	7	24	25	15	26	0

Test Scenario #12

		Vehicle Stop								
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9
	1	1	35	33	31	27	0			
	2	7	21	40	0					
	3	34	30	32	28	24	25	0		
Monday	4	2	3	20	0					
	5	26	41	42	0					
	6	4	36	37	38	39	0			
	7	8	22	23	29	0				
	1	10	9	23	22	0				
	2	26	19	15	27	0				
Tuesday	3	5	12	17	36	41	40	0		
	4	6	20	18	38	39	0			
	6	32	34	30	24	25	35	33	31	0

Test Scenario #12 (cont'd.)

Tuesday	7	11	16	37	28	21	29	0
	2	28	16	37	40	41	0	
	3	18	24	25	35	26	27	0
Wednesday	4	20	30	21	15	31	0	
	5	19	17	36	42	38	39	0
	6	32	34	33	22	23	29	0
Thursday	2	42	0					
Friday	4	14	15	0				
	7	13	19	17	16	18	0	

C.7. HPVRPB Solutions Obtained Using Basic Model Formulation

Test Scenario #1

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	7	5	4	0	
Tuesday	5	2	4	0	
Wednesday	5	3	5	0	
Thursday	5	1	5	4	0

Test Scenario #2

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	7	6	5	0	
Tuesday	5	1	5	0	
Thursday	3	3	5	6	0
Friday	5	2	4	6	0

Test Scenario #3

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	1	7	0	
Tuesday	5	5	6	0	
Wednesday	5	3	6	0	
	7	7	0		
Friday	5	2	4	6	0
	7	7	0		

Test Scenario #4

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	5	6	0	
	7	7	0		
Tuesday	5	2	4	6	0
	7	7	0		
Thursday	5	1	7	0	
Friday	5	3	6	0	

Test Scenario #5

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	3	6	0	
	7	8	9	0	
Tuesday	5	5	0		
Wednesday	5	7	8	9	0
Thursday	5	2	4	0	
Friday	5	1	8	9	0

Test Scenario #6

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	2	4	0	
Tuesday	5	3	6	0	
Wednesday	4	5	0		
	5	8	9	10	0
Thursday	5	7	9	10	0
Friday	5	1	9	10	0

Test Scenario #7

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	4	5	13	0	
	5	7	14	15	0
	7	12	11	0	
Tuesday	5	6	1	15	0
	7	13	12	11	0
Wednesday	5	3	12	13	0
Thursday	5	8	14	15	0
Friday	4	10	9	14	0
	5	4	2	11	0

Test Scenario #8

		Vehicle Path									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Monday	5	5	19	13	12	17	16	11	18	20	0
Tuesday	5	7	14	15	13	19	0				
	7	11	16	0							
Wednesday	4	10	9	14	20	0					
	5	1	15	13	19	0					
Thursday	5	3	17	12	16	11	18	20	0		
Friday	4	6	2	4	18	0					
	5	8	14	15	0						
	7	17	12	0							

Test Scenario #9

		Tour Stop							
Weekday	**Vehicle**	1	2	3	4	5	6	7	8
Monday	4	10	9	23	22	25	0		
	5	1	15	0					
	7	16	17	19	0				
Tuesday	5	5	4	18	20	0			
	7	24	21	15	0				
Wednesday	5	8	22	23	21	24	25	0	
	7	16	17	19	0				
Thursday	5	12	3	17	0				
	6	6	11	16	18	20	19	0	
Friday	1	2	7	21	24	25	0		
	5	13	14	22	23	15	18	20	0

Test Scenario #10

		Tour Stop									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Monday	1	14	7	21	30	25	24	0			
	3	10	9	23	27	15	26	0			
	5	5	0								
	7	20	18	16	0						
Tuesday	5	12	3	17	19	0					
	6	6	20	18	27	15	26	21	30	25	0
Wednesday	5	8	22	29	23	24	28	16	0		
	7	19	20	18	0						
Thursday	3	4	11	2	16	0					
	6	28	21	30	29	22	24	25	0		
	7	17	0								

Test Scenario #10 (cont'd.)

		Tour Stop									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Friday	5	1	15	22	29	23	27	26	0		
	6	13	19	17	28	0					

C.8. HPVRPB Solutions Obtained by Adding Tour Limitation Constraints to Model Formulation

Test Scenario #1

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	2	4	0	
Tuesday	5	1	5	0	
Wednesday	5	3	5	4	0
Friday	7	4	5	0	

Test Scenario #2

		Vehicle Path			
Weekday	**Vehicle**	1	2	3	4
Monday	5	2	6	0	
Tuesday	1	1	5	6	0
Wednesday	5	3	5	0	
Friday	6	4	6	5	0

Test Scenario #3

		Tour Stop		
Weekday	**Vehicle**	1	2	3
Monday	5	2	6	0
	7	7	0	
Tuesday	5	5	0	
Wednesday	7	7	0	
Thursday	6	4	6	0
Friday	5	3	6	0

Test Scenario #4

		Tour Stop		
Weekday	**Vehicle**	1	2	3
Monday	5	1	0	
	6	6	0	
Tuesday	7	8	7	0
Wednesday	5	3	0	
	7	8	7	0
Thursday	5	5	0	
	6	4	0	

Test Scenario #4 (cont'd.)

Friday	5	2	0	
	7	8	7	0

Test Scenario #5

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	5	0		
	6	6	0		
	7	8	9	0	
Tuesday	5	3	0		
Wednesday	5	2	0		
Thursday	5	7	8	9	0
	6	4	0		
Friday	5	1	8	9	0

Test Scenario #6

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	8	9	10	0
Tuesday	4	5	0		
	5	3	0		
Wednesday	5	1	9	10	0
Thursday	5	2	0		
	6	4	0		
Friday	5	7	9	10	0
	6	6	0		

Test Scenario #7

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
	4	5	13	0	
Monday	5	7	14	15	0
	7	11	12	0	
Tuesday	5	3	12	13	0
	4	10	0		
Wednesday	5	1	15	0	
	6	4			
	7	11			
Thursday	5	8	14	0	
	6	6	0		
Friday	4	2	11	0	
	5	9	14	15	0

Test Scenario #8

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
	4	5	13	19	0			
Monday	5	3	17	12	11	16	18	0
	6	6	20	0				
	4	10	0					
Tuesday	5	7	14	20	0			
	6	4	18	16	11	12	17	0
Wednesday	5	1	15	17	12	19	13	0
Thursday	5	8	14	15	19	13	0	
Friday	4	2	11	16	18	20	0	
	5	9	14	15	0			

Test Scenario #9

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
	4	1	15	0				
Monday	5	7	21	24	25	0		
	6	12	17	16	18	0		
Tuesday	5	9	23	22	21	24	25	0
	6	13	19	17	15	0		
	4	2	16	0				
Wednesday	5	8	22	21	24	0		
	6	6	20	0				
	4	10	25	0				
Thursday	5	3	17	19	0			
	6	11	16	18	20	0		

Test Scenario #9 (cont'd.)

	4	5	19	0			
Friday	5	14	22	23	15	0	
	6	4	18	20	0		

Test Scenario #10

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
	5	8	22	29	23	27	15	0
Monday	6	11	16	18	20	26	0	
	7	30	21	28	0			
	4	9	23	29	22	26	0	
	5	3	17	19	0			
Tuesday	6	4	18	20	0			
	7	30	21	28	0			

Test Scenario #10 (cont'd.)

	4	2	16	18	0	
Wednesday	5	7	21	30	24	25
	6	13	19	0		
	7	15	27	17	0	
	4	10	24	25	0	
Thursday	5	1	15	27	26	0
	6	6	20	0		
	4	5	19	0		
Friday	5	14	22	29	23	24
	6	12	17	16	28	0

Test Scenario #11

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	4	1	15	26	0			
	5	8	22	23	29	33	35	0
	6	11	16	17	19	0		
	7	32	28	20	0			
Tuesday	5	9	23	29	22	35	26	0
	6	12	17	16	28	32	0	
	7	15	27	31	0			
Wednesday	4	5	19	20	0			
	5	14	22	31	27	26	0	
	6	4	18	34	30	21	24	0
	7	25	32	28	0			

Test Scenario #11 (cont'd.)

	4	2	18	0				
Thursday	5	7	21	30	24	25	0	
	6	6	34	23	29	33	35	0
	4	10	30	21	24	25	0	
Friday	5	3	17	19	0			
	6	13	15	27	31	33	34	0
	7	16	18	20	0			

C.9. HPVRPB Solutions Obtained Using MULTI-HGA-ROUTE

Test Scenario #1

		Tour Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	2	0		
Tuesday	5	3	5	4	0
Wednesday	4	1	5	4	0
Friday	7	4	5	0	

Test Scenario #2

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	4	1	5	0		
Tuesday	4	4	3	5	6	0
Wednesday	5	2	6	5	0	
Friday	7	6	0			

Test Scenario #3

		Vehicle Stop			
Weekday	**Vehicle**	1	2	3	4
Monday	5	5	4	6	7
Tuesday	5	1	0		
Wednesday	5	3	6	0	
	7	7	0		
Thursday	5	2	7	6	0

Test Scenario #4

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	6	4	0			
Tuesday	2	8	7	0		
Wednesday	5	6	5	8	7	0
Thursday	5	3	0			
Friday	1	2	1	8	7	

Test Scenario #5

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	5	2	4	8	9	0
Tuesday	5	3	0			
	7	8	9	0		
Thursday	2	6	1	0		
	5	7	0			
Friday	4	5	8	9	0	

Test Scenario #6

		Tour Stop				
Weekday	**Vehicle**	1	2	3	4	5
Monday	4	6	0			
Wednesday	2	2	0			
	4	1	9	10	0	
Thursday	5	4	5	9	10	0
Friday	1	7	8	9	10	0
	5	3	0			

Test Scenario #7

		Tour Stop						
Weekday	**Vehicle**	1	2	3	4	5	6	7
Monday	1	8	9	6	0			
	5	10	7	15	0			
Tuesday	2	2	0					
Wednesday	5	4	11	12	13	15	14	0
Thursday	4	5	13	0				
	5	1	15	14	12	11	0	
Friday	2	3	12	11	14	13	0	

Test Scenario #8

		Vehicle Stop										
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10	11
Tuesday	1	4	3	17	19	0						
	6	14	15	18	16	20	0					
Wednesday	1	5	1	15	14	18	20	0				
	4	6	2	16	11	12	17	13	19	0		
	5	14	15	12	17	13	19	18	11	16	20	0
Thursday	1	8	9	0								
	5	10	7	11	12	13	0					

Test Scenario #9

		Tour Stop									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Tuesday	3	5	0								
	5	8	0								
Wednesday	1	14	9	23	22	21	24	25	0		
	5	13	12	17	19	15	16	18	20	0	
Thursday	1	10	7	11	16	18	0				
	4	1	15	0							
	6	6	20	17	19	23	22	21	24	25	0
Friday	3	4	2	16	18	20	24	21	25	0	
	4	3	17	19	15	22	23	0			

Test Scenario #10

		Tour Stop										
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10	11
Monday	6	15	19	17	16	18	20	0				
Tuesday	1	5	3	0								
	5	4	14	22	23	29	0					
	6	13	15	27	24	30	21	28	0			
	7	16	26	25	0							
Wednesday	3	6	2	28	0							
	4	1	27	19	0							
	5	7	21	30	29	23	22	24	25	26	20	0
Thursday	3	11	12	17	19	20	0					
	7	16	18	15	0							

Test Scenario #10 (cont'd.)

	2	8	9	23	29	22	24	0
Friday	5	10	25	30	21	28	0	
	6	18	17	27	26	0		

Test Scenario #11

		Vehicle Path										
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10	11
Monday	2	10	8	22	23	21	24	25	20	18	0	
	5	26	27	15	19	17	16	28	0			
Tuesday	4	7	0									
	2	5	13	19	17	18	26	35	25	0		
Wednesday	3	14	22	23	29	33	31	27	15	0		
	6	4	20	16	28	32	21	30	34	24	0	

Test Scenario #11 (cont'd.)

	1	1	15	27	31	33	22	23	29	0		
Thursday	3	11	16	28	32	24	21	30	34	35	25	0
	5	3	17	19	26	18	20	0				
Friday	4	6	12	2	32	0						
	5	9	29	33	31	34	30	35	0			

Test Scenario #12

		Vehicle Stop									
Weekday	**Vehicle**	1	2	3	4	5	6	7	8	9	10
Monday	1	18	20	26	15	23	22	21	24	25	0
	5	16	40	39	41	42	17	19	0		
Tuesday	1	5	0								
	4	10	32	30	34	33	31	27	15	0	
	6	17	36	37	38	16	18	0			
	7	28	29	35	0						
Wednesday	1	8	9	23	29	33	34	35	0		
	4	6	4	2	28	32	24	25	0		
	5	7	30	31	27	26	0				
	6	36	42	41	39	38	40	37	0		

Test Scenario #12 (cont'd.)

Thursday	1	20	19	15	27	31	21	35	0		
	2	3	17	36	18	24	25	0			
	4	11	16	37	40	38	39	41	42	0	
	6	26	29	23	22	33	34	30	32	28	0
Friday	3	1	12	13	19	20	0				
	4	14	22	21	0						