SUPPLY CHAIN NETWORK CONFIGURATION: DYNAMICITY AND SUSTAINABILITY

A Dissertation by

Shokoufeh Mirzaei

Master of Science, Isfahan University of Technology, 2009

Bachelor of Science, Iran University of Science and Technology, 2006

Submitted to the Department of Industrial and Manufacturing Engineering
and the faculty of the Graduate School of
Wichita State University
in partial fulfillment of
the requirement for the degree of
Doctor of Philosophy

May 2012
SUPPLY CHAIN NETWORK CONFIGURATION: SUSTAINABILITY AND DYNAMICITY

The following faculty have examined the final copy of this dissertation for form and content, and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy with a major in Industrial Engineering.

_____________________________________________________
Krishna Krishnan, Committee Chair

_____________________________________________________
Don Malzahn, Committee Member

_____________________________________________________
Ramazan Asmatulu, Committee Member

_____________________________________________________
Pingfeng Wang, Committee Member

_____________________________________________________
Vinod Namboodiri, Committee Member

Accepted for the College of Engineering

_____________________________________________________
Zulma Toro-Ramos, Dean

Accepted for the Graduate School

_____________________________________________________
J. David McDonald, Dean
DEDICATION

To my parents, sister, and dearest friend
for their unconditional love and support
ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Professor Krishna Krishnan, who gave me freedom to explore on my own and yet guidance when my steps faltered. I have been incredibly fortunate to have him as my advisor. His patience and support, especially with my writing, helped me overcome many crisis situations and finish this dissertation. I hope that one day I will become as good a person as my advisor has been to me.

I would like to thank my committee members, Don Malzahn, Ramazan Asmatulu, Pingfeng Wang, and Vinod Namboodiri. I also thank Dr. Neeraj Jaggi, for attending my dissertation proposal meeting. Many thanks to all the members of our research group over the past three year for their friendship and help.

Most importantly, none of this would have been possible without the love and patience of my family and friend, especially Armin Ghoddoussi. My family, to whom this dissertation is dedicated, has been a constant source of love, concern, support, and strength all these years. I express heart-felt gratitude to my family.
ABSTRACT

This dissertation consists of five submission-ready accepted/submitted papers that address some of the key supply chain problems. Supply chain problems, in terms of the area that they address, can be classified into four major groups: location-allocation problem, transportation problem, manufacturing problem, and inventory problem. In this dissertation, location-allocation and location-routing problems, also called LRPs, are studied using two approaches. In the first approach, presented in Chapters 2 and 3, it is assumed that the value of some parameters of the network are dynamically changing. The objective here is to minimize the total system cost by finding the best location-allocation and routing plan when demand and travel times are dynamic. The dynamic nature of demand/travel time is presented by functions obtained from historical data. In the second approach, the sustainability perspective of the LRP is considered. The objective here is also to minimize the total network cost. However, the total cost is presented in terms of energy cost because of the lack of literature investigating the energy effectiveness of a location-routing plan. Traditionally, the objective function of the LRP is expressed in terms of distance minimization, although distance is not the only factor that contributes to energy consumption in an LRP. This perspective is thoroughly discussed in chapters 4 and 5.

Due to the rising price of fuel, industries are concerned more than ever about their transportation costs and modes. In the current economic atmosphere, railway transportation is extremely in demand. Hence, to continue the sustainability part of this dissertation, a rail freight transportation system is investigated. The objective here is to develop a heuristic algorithm that can provide a cost-effective train scheduling plan in a matter of seconds. The main contribution in this section is the integration of a pool of business cost elements and constraints existing in practical train-scheduling problems for obtaining results.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>INTRODUCTION</td>
</tr>
<tr>
<td>1.1</td>
<td>Dynamicity</td>
</tr>
<tr>
<td>1.2</td>
<td>Sustainability</td>
</tr>
<tr>
<td>1.3</td>
<td>Research Objectives and Outlines</td>
</tr>
<tr>
<td>1.4</td>
<td>References</td>
</tr>
<tr>
<td>2.</td>
<td>GENERIC FORMULATION OF LOCATION-ROUTING PROBLEM WITH DYNAMIC DEMAND</td>
</tr>
<tr>
<td>2.1</td>
<td>Abstract</td>
</tr>
<tr>
<td>2.2</td>
<td>Introduction</td>
</tr>
<tr>
<td>2.3</td>
<td>Problem Statement</td>
</tr>
<tr>
<td>2.4</td>
<td>Mathematical Formulation of Problem</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Formulation of LRP with Time-Dependent Customer Demands</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Formulation of LRP with Time Windows</td>
</tr>
<tr>
<td>2.4.2.1</td>
<td>Formulation of LRP with Hard Time Window</td>
</tr>
<tr>
<td>2.4.2.2</td>
<td>Formulation of LRP with Soft Time Window</td>
</tr>
<tr>
<td>2.5</td>
<td>Illustrative Examples</td>
</tr>
<tr>
<td>2.5.1</td>
<td>Example of LRP with Time-Dependent Customer Demands</td>
</tr>
<tr>
<td>2.5.2</td>
<td>Example of VRP with Hard Time Window</td>
</tr>
<tr>
<td>2.5.3</td>
<td>Example of VRP with Soft Time Window</td>
</tr>
<tr>
<td>2.6</td>
<td>Additional Case Studies and Constraint Relaxations for Infeasibility</td>
</tr>
<tr>
<td>2.7</td>
<td>Linearization and Computation Results</td>
</tr>
<tr>
<td>2.8</td>
<td>Conclusions</td>
</tr>
<tr>
<td>2.9</td>
<td>References</td>
</tr>
<tr>
<td>3.</td>
<td>LOCATION-ROUTING PROBLEM WITH TIME-DEPENDENT TRAVEL TIMES</td>
</tr>
<tr>
<td>3.1</td>
<td>Abstract</td>
</tr>
<tr>
<td>3.2</td>
<td>Introduction</td>
</tr>
<tr>
<td>3.3</td>
<td>Problem Statement</td>
</tr>
<tr>
<td>3.4</td>
<td>Mathematical Formulation of Problem</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Formulation of TDLRP without Time Windows</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Formulation of TDLRP with Hard Time Window</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Formulation of TDLRP with Soft Time Window</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Formulation of TDLRP with Dynamic Travel Time and Demand</td>
</tr>
<tr>
<td>3.5</td>
<td>Illustrative Examples</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Example of TDLRP without Time Window Constraints</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Example of TDLRP with Hard Time Window Constraints</td>
</tr>
<tr>
<td>3.5.3</td>
<td>Example of TDLRP with Soft Time Window Constraints</td>
</tr>
<tr>
<td>3.5.4</td>
<td>Example of TDLRP with Time-Dependent Demand</td>
</tr>
<tr>
<td>3.6</td>
<td>Linearization and Solution Approach</td>
</tr>
<tr>
<td>3.7</td>
<td>Conclusions</td>
</tr>
<tr>
<td>3.8</td>
<td>References</td>
</tr>
<tr>
<td>4.</td>
<td>ENERGY-EFFICIENT LOCATION-ROUTING PROBLEM (EELRP): GENERIC FORMULATION, HARD AND SOFT TIME WINDOWS, AND TIME-DEPENDENT DEMAND</td>
</tr>
<tr>
<td>4.1</td>
<td>Abstract</td>
</tr>
<tr>
<td>4.2</td>
<td>Introduction</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS (continued)

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3</td>
<td>Problem Statement</td>
</tr>
<tr>
<td>4.4</td>
<td>Mathematical Formulation of Problem</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Formulation of EELRP</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Formulation of EELRP with Time Window</td>
</tr>
<tr>
<td>4.4.2.1</td>
<td>Formulation of EELRP with Hard Time Window</td>
</tr>
<tr>
<td>4.4.2.2</td>
<td>Formulation of EELRP with Soft Time Window</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Formulation of EELRP with Time-Dependent Customer Demand</td>
</tr>
<tr>
<td>4.5</td>
<td>Illustrative Examples</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Example of EELRP</td>
</tr>
<tr>
<td>4.5.2</td>
<td>Example of EELRP with Hard Time Window</td>
</tr>
<tr>
<td>4.5.3</td>
<td>Example of EELRP with Soft Time Window</td>
</tr>
<tr>
<td>4.5.4</td>
<td>Example of EELRP with Time-Dependent Demand</td>
</tr>
<tr>
<td>4.6</td>
<td>Conclusions</td>
</tr>
<tr>
<td>4.7</td>
<td>References</td>
</tr>
<tr>
<td>5.</td>
<td>HEURISTIC ALGORITHM FOR ENERGY-EFFICIENT VEHICLE-ROUTING PROBLEM</td>
</tr>
<tr>
<td>5.1</td>
<td>Abstract</td>
</tr>
<tr>
<td>5.2</td>
<td>Introduction</td>
</tr>
<tr>
<td>5.3</td>
<td>Clarke-Wright Savings Algorithm</td>
</tr>
<tr>
<td>5.4</td>
<td>Energy-Efficient Clarke-Wright Savings Algorithm</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Calculation of Savings Values for EECW Algorithm</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Modified Clarke-Wright Savings Algorithm</td>
</tr>
<tr>
<td>5.5</td>
<td>Computation Results</td>
</tr>
<tr>
<td>5.6</td>
<td>Conclusions</td>
</tr>
<tr>
<td>5.7</td>
<td>References</td>
</tr>
<tr>
<td>6.</td>
<td>HEURISTIC ALGORITHM FOR BLOCK-TO-TRAIN ASSIGNMENT AND TRAIN-ROUTING PROBLEMS</td>
</tr>
<tr>
<td>6.1</td>
<td>Abstract</td>
</tr>
<tr>
<td>6.2</td>
<td>Introduction</td>
</tr>
<tr>
<td>6.3</td>
<td>Heuristic Algorithm for Block-to-Train Assignment and Train-Routing Problem</td>
</tr>
<tr>
<td>6.4</td>
<td>Conclusions</td>
</tr>
<tr>
<td>6.5</td>
<td>References</td>
</tr>
<tr>
<td>7.</td>
<td>CONCLUSIONS</td>
</tr>
<tr>
<td>7.1</td>
<td>Summary and Conclusions</td>
</tr>
<tr>
<td>7.2</td>
<td>Future Work</td>
</tr>
<tr>
<td>APPENDIX</td>
<td></td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Customers’ Demand Information for Two-Layer Network Problem with Four Customers (Nodes 1–4) and Two DCs (Nodes 5–6)</td>
</tr>
<tr>
<td>2.2</td>
<td>Transportation Cost between Nodes ($ Thousands) for Two-Layer Network Problem with Four Customers (Nodes 1–4) and Two DCs (Nodes 5–6)</td>
</tr>
<tr>
<td>2.3</td>
<td>Transportation Time between Nodes (Hours) for Two-Layer Network Problem with Four Customers (Nodes 1–4) and Two DCs (Nodes 5–6)</td>
</tr>
<tr>
<td>2.4</td>
<td>Time Interval Assigned to Each Customer for VRP with Hard Time Window</td>
</tr>
<tr>
<td>2.5</td>
<td>Time Interval Assigned to Each Customer for VRP with Soft Time Window</td>
</tr>
<tr>
<td>2.6</td>
<td>Actual and Recorded Arrival Times for Proposed and Arc Formulations</td>
</tr>
<tr>
<td>2.7</td>
<td>Customers’ Demand Information for Two-Layer Network Problem with Five Customers (Nodes 1–5) and Two DCs (Nodes 6–7)</td>
</tr>
<tr>
<td>2.8</td>
<td>Transportation Cost Between Nodes ($ Thousands) for Two-Layer Network Problem with Five Customers (Nodes 1–5) and Two DCs (Nodes 6–7)</td>
</tr>
<tr>
<td>2.9</td>
<td>Transportation Time Between Nodes (Hours) for Two-Layer Network Problem with Five Customers (Nodes 1–5) and Two DCs (Nodes 6–7)</td>
</tr>
<tr>
<td>2.10</td>
<td>Results of Six-Node Example #1 when Objective Function Value = $3.4957 \times 10^2$</td>
</tr>
<tr>
<td>2.11</td>
<td>Results of Seven-Node Example #2 when Objective Function Value = $3.4442 \times 10^2$</td>
</tr>
<tr>
<td>2.12</td>
<td>Results of Eight-Node Example #3 when Objective Function Value = $3.5055 \times 10^2$</td>
</tr>
<tr>
<td>2.13</td>
<td>Results of Six-Node Example #1 when Objective Function Value = $3.4957 \times 10^2$</td>
</tr>
<tr>
<td>2.14</td>
<td>Results of Seven-Node Example #2 when Objective Function Value = $2.675 \times 10^2$</td>
</tr>
<tr>
<td>3.1</td>
<td>Results of TDLRP without Time Window Constraints</td>
</tr>
<tr>
<td>3.2</td>
<td>Time Interval Assigned to Each Customer for TDLRP with Hard Time Window</td>
</tr>
<tr>
<td>3.3</td>
<td>Results of TDLRP with Hard Time Window</td>
</tr>
<tr>
<td>3.4</td>
<td>Time Interval, Violation, and Penalty Cost Assigned to Each Customer for TDLRP with Soft Time Window</td>
</tr>
<tr>
<td>3.5</td>
<td>Results of TDLRP with Soft Time Window</td>
</tr>
<tr>
<td>3.6</td>
<td>Customers’ Demand Information for TDLRP with Time-Dependent Demand</td>
</tr>
<tr>
<td>3.7</td>
<td>Results of TDLRP with Time-Dependent Demand</td>
</tr>
<tr>
<td>3.8</td>
<td>Information Regarding Examples Solved</td>
</tr>
<tr>
<td>3.9</td>
<td>Results of Six-Node Example #1 when Objective Function Value = $5.350 \times 10^2$</td>
</tr>
<tr>
<td>3.10</td>
<td>Results of Seven-Node Example #2 when Objective Function Value = $2.675 \times 10^2$</td>
</tr>
</tbody>
</table>
LIST OF TABLES (continued)

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.11</td>
<td>56</td>
</tr>
<tr>
<td>3.12</td>
<td>56</td>
</tr>
<tr>
<td>4.1</td>
<td>72</td>
</tr>
<tr>
<td>4.2</td>
<td>73</td>
</tr>
<tr>
<td>4.3</td>
<td>73</td>
</tr>
<tr>
<td>4.4</td>
<td>73</td>
</tr>
<tr>
<td>4.5</td>
<td>73</td>
</tr>
<tr>
<td>4.6</td>
<td>74</td>
</tr>
<tr>
<td>4.7</td>
<td>74</td>
</tr>
<tr>
<td>4.8</td>
<td>74</td>
</tr>
<tr>
<td>4.9</td>
<td>75</td>
</tr>
<tr>
<td>4.10</td>
<td>75</td>
</tr>
<tr>
<td>4.11</td>
<td>76</td>
</tr>
<tr>
<td>4.12</td>
<td>76</td>
</tr>
<tr>
<td>4.13</td>
<td>77</td>
</tr>
<tr>
<td>4.14</td>
<td>77</td>
</tr>
<tr>
<td>4.15</td>
<td>78</td>
</tr>
<tr>
<td>4.16</td>
<td>78</td>
</tr>
<tr>
<td>4.17</td>
<td>79</td>
</tr>
<tr>
<td>4.18</td>
<td>79</td>
</tr>
<tr>
<td>4.19</td>
<td>80</td>
</tr>
<tr>
<td>4.20</td>
<td>80</td>
</tr>
<tr>
<td>4.21</td>
<td>80</td>
</tr>
<tr>
<td>4.22</td>
<td>81</td>
</tr>
<tr>
<td>5.1</td>
<td>93</td>
</tr>
</tbody>
</table>
# LIST OF TABLES (continued)

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2</td>
<td>Results of Benchmark Problems from Li et al.</td>
</tr>
<tr>
<td>6.1</td>
<td>Condition of Routes I and J Based on Similarity Indices</td>
</tr>
<tr>
<td>6.2</td>
<td>Input Summary of First Case Study</td>
</tr>
<tr>
<td>6.3</td>
<td>Input Summary of Second Case Study</td>
</tr>
<tr>
<td>6.4</td>
<td>Output Summary of First Case Study</td>
</tr>
<tr>
<td>6.5</td>
<td>Output Summary of Second Case Study</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Two-layer network problem with four customers (nodes 1–4) and two DCs (nodes 5–6)</td>
<td>16</td>
</tr>
<tr>
<td>2.2</td>
<td>Network configuration solution for two-layer LRPD with four customers and two DCs</td>
<td>18</td>
</tr>
<tr>
<td>2.3</td>
<td>Network configuration solution for VRPTW from proposed and existing models</td>
<td>19</td>
</tr>
<tr>
<td>2.4</td>
<td>Network configuration solution for VRP with soft time window using proposed formulation</td>
<td>22</td>
</tr>
<tr>
<td>2.5</td>
<td>Two-layer network problem with five customers (nodes 1–5) and two DCs (nodes 6–7)</td>
<td>23</td>
</tr>
<tr>
<td>2.6</td>
<td>Network configuration solution for LRPTD with 30- and 40-vehicle capacity</td>
<td>25</td>
</tr>
<tr>
<td>2.7</td>
<td>Network configuration solution for LRP with three 30-unit vehicles</td>
<td>26</td>
</tr>
<tr>
<td>2.8</td>
<td>Network configuration solution for VRP when ω = $0.2$ and assignment of all customers is relaxed</td>
<td>27</td>
</tr>
<tr>
<td>3.1</td>
<td>Travel time functions between customers and DCs</td>
<td>48</td>
</tr>
<tr>
<td>3.2</td>
<td>Travel time functions between each pair of customers in network</td>
<td>49</td>
</tr>
<tr>
<td>4.1</td>
<td>Three-layer network problem with one plant (P), four customers (nodes 1–4), and 2 DCs (nodes 5–6)</td>
<td>72</td>
</tr>
<tr>
<td>4.2</td>
<td>Network configuration solution for EELRP</td>
<td>75</td>
</tr>
<tr>
<td>4.3</td>
<td>Network configuration solution for LRP</td>
<td>75</td>
</tr>
<tr>
<td>4.4</td>
<td>Network configuration solution for EELRP with hard time window</td>
<td>76</td>
</tr>
<tr>
<td>4.5</td>
<td>Network configuration solution for LRP with hard time window</td>
<td>77</td>
</tr>
<tr>
<td>4.6</td>
<td>Network configuration solution for EELRP with soft time window</td>
<td>78</td>
</tr>
<tr>
<td>4.7</td>
<td>Network configuration solution for LRP with soft time window</td>
<td>79</td>
</tr>
<tr>
<td>4.8</td>
<td>Network configuration solution for EELRP with time-dependent demand</td>
<td>80</td>
</tr>
<tr>
<td>4.9</td>
<td>Network configuration solution for LRP with time-dependent demand</td>
<td>81</td>
</tr>
<tr>
<td>5.1</td>
<td>Merging routes in savings method</td>
<td>88</td>
</tr>
<tr>
<td>5.2</td>
<td>Comparison of CW and EECW computation times</td>
<td>94</td>
</tr>
<tr>
<td>6.1</td>
<td>Sequence of route construction and route improvement steps to provide near-optimal results</td>
<td>103</td>
</tr>
</tbody>
</table>
### LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BTA</td>
<td>Block-to-Train Assignment</td>
</tr>
<tr>
<td>BTRP</td>
<td>Block-to-Train-Routing Problem</td>
</tr>
<tr>
<td>CW</td>
<td>Clark-Wright</td>
</tr>
<tr>
<td>DC</td>
<td>Distribution Center</td>
</tr>
<tr>
<td>EECW</td>
<td>Energy-Efficient Clark-Wright</td>
</tr>
<tr>
<td>EELRP</td>
<td>Energy-Efficient Location-Routing Problem</td>
</tr>
<tr>
<td>EEVRP</td>
<td>Energy-Efficient Vehicle-Routing Problem</td>
</tr>
<tr>
<td>LAP</td>
<td>Location-Allocation Problem</td>
</tr>
<tr>
<td>LRP</td>
<td>Location-Routing Problem</td>
</tr>
<tr>
<td>LRPTD</td>
<td>Location-Routing Problem with Time-Dependent Demand</td>
</tr>
<tr>
<td>LRPTW</td>
<td>Location-Routing Problem with Time Window</td>
</tr>
<tr>
<td>MILP</td>
<td>Mixed-Integer Linear Programming</td>
</tr>
<tr>
<td>MINLP</td>
<td>Mixed-Integer Non-Linear Programming</td>
</tr>
<tr>
<td>TDLRP</td>
<td>Location-Routing Problem with Time-Dependent Travel Time</td>
</tr>
<tr>
<td>TDLRPTW</td>
<td>Location-Routing Problem with Time-Dependent Travel Time with Travel Window</td>
</tr>
<tr>
<td>TDVRP</td>
<td>Vehicle-Routing Problem with Time-Dependent Travel Time</td>
</tr>
<tr>
<td>TR</td>
<td>Train Routing</td>
</tr>
<tr>
<td>TSP</td>
<td>Traveling Salesman Problem</td>
</tr>
<tr>
<td>VRP</td>
<td>Vehicle-Routing Problem</td>
</tr>
<tr>
<td>VRPTD</td>
<td>Vehicle-Routing Problem with Time-Dependent Demand</td>
</tr>
<tr>
<td>VRPTW</td>
<td>Vehicle-Routing Problem with Time Window</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

The objective of a supply chain network is to provide products (services) to customers wherever and whenever they want and with minimum cost. In order to achieve this objective, it is necessary to design a network with facilities and distributors to perform their functions in cooperation with each other and in favor of the system’s ultimate goal. The functions of network elements can be procurement of materials, transformation of these materials into intermediate and finished goods, and distribution of the products to customers. Configuration of these elements has attracted considerable attention among logistics researchers in the context of the supply chain. A supply chain network is the integration of different decisions including production, location-allocation, inventory, and transportation. Supply chain problems are varied, depending on the decisions they address and the characteristics of contributing factors. In this dissertation, two decision problems—location-allocation and transportation (routing)—are addressed simultaneously. These problems are defined separately as follows:

**Location-Allocation Problem (LAP):** Given a set of facilities and a set of customers, locating the set of facilities and allocating the customers to them in such a way that minimizes the cost of transportation.

**Vehicle-Routing Problem (VRP):** According to Laporte (1992), “...the problem of designing optimal delivery or collection routes from one or several distribution centers (DCs) to a number of geographically scattered cities or customers, subject to side constraints.” The VRP is a general form of the traveling salesman problem (TSP). The VRP determines vehicle routes, where a route is a tour that starts at a depot, visits a subset of customers in a defined order, and returns to the same depot. All customers are visited once, and total demands of the customers in the same route should not exceed the vehicle capacity. The objective of the VRP is minimization of the total distribution cost.

The LAP and VRP addressed together is referred to as a location-routing problem (LRP), which deals with selecting a number of DCs among a set of potential DCs, allocating customers to them, and finding a routing plan to serve the customers. According to Guerra et al. (2007), “Integrated location routing models are used to solve the facility location problem and the vehicle-routing problem simultaneously in order to reflect the interactions between these two decisions’ background in supply chain categories and application.”

The LAP and VRP are dependent problems, and separate optimization of them usually concludes a non-optimal decision. The interdependence of the VRP and LAP was not recognized until 1970 (Perl and Daskin, 1985).
Solving the LRP involves two types of sequential methods. The first type solves the LAP and then uses results from the LAP to solve the VRP (Jacobsen and Madsen, 1980; Or and Pierskalla, 1979). In the second type, the allocation, routing, and location problems are solved sequentially. Depending on the problem constraints, either of them can be useful. In addition to the sequential method, many recent papers have focused on integrating these problems and obtaining the results for all problems simultaneously. In this dissertation, an integrated approach for the LRP has been developed.

1.1 Dynamicity

The dynamic version of supply chain problems is the subject of Chapters 2 and 3 in this dissertation. In a dynamic environment, the focus is on finding the location of facilities, allocation of customers, and routing plan over an extended time horizon. Several parameters can be dynamic in a supply chain network. Some examples of dynamic factors are customers’ demands, travel times, production cycle, pricing, etc. In this dissertation, the factors selected to be dynamic are limited to customers’ demands and travel times, which are investigated separately and in conjunction with each other in Chapters 2 and 3. The definition of “dynamic demand” in this dissertation differs from the usual definition found in the literature. In traditional dynamic supply chain problems, demand is defined as a discrete value for each time period. Demand quantities may vary from one period to another. However, during a specific time period, the demand is fixed. In this dissertation, demand is defined to be dynamic within a period. Thus, customers’ demands are continuously changing as a function of time after initiation. In fact, dynamic demand occurs “when customers’ demands do not remain the same after initiation and change with time according to a predefined function.” This function is usually estimated/forecasted from historical data. The proposed optimization model in this dissertation can be used for discrete and continuous demand functions during a time period. An example is the newspaper delivery problem in which the number of newspapers/magazines requested by a store usually decreases over time.

In addition to dynamic customer demand, travel times are also considered dynamic. In Chapter 3, the dynamic travel time is defined as “when travel times change according to a predefined function usually estimated/forecasted from the congestion historical data.”

Based on historical information, the travel time function can be estimated and used for solving the LRP with time-dependent travel time (TDLRP). The number of papers addressing the exact formulation of this problem is very small. All existing formulations of the TDLRP in the literature consider constraints for the arrival time at
each customer in the system. This assumption simplifies the formulation of the problem. However, it allows for waiting times in the system, which is not always realistic. Hence, this disadvantage of the existing formulation has been addressed in this dissertation, and the formulation provided eliminates the waiting times.

The proposed formulations in Chapters 2 and 3 involve mixed-integer non-linear programming (MINLP). Thus, to solve the problems using CPLEX (an optimization software package) or advanced exact heuristics, it is necessary to linearize the model. Hence, the proposed non-linear models are linearized, and different cutting strategies are tested to find the best approach in terms of computation time.

1.2 Sustainability

The main part of a distribution system is transportation. Transportation is a huge business in the United States and around the world. According to the U.S. Department of Transportation, Federal Highway Administration, in the year 2008, more than 2.2 million registered combination trucks (truck-trailers) traveled more than 134.6 billion miles on the nation’s highways and consumed 26.8 billion gallons of diesel fuel. This amount of fuel is equal to 12% to 13% of the total U.S. petroleum usage. In addition, “The number of gallons of fuel burned by commercial trucks increased significantly over the past 28 years. Between 1980 and 2008, the fuel consumed in highway freight transportation increased from 20 billion to nearly 37 billion gallons annually. This is due to a substantial increase in the number of trucks on the road, an increase in the average number of miles traveled per truck, and a doubling of truck miles traveled.”

These statistics show the importance of energy consumption in a transportation/distribution network. Despite this fact, formulation of the location-routing problem has focused traditionally on distance minimization. However, minimizing the distance will not always result in the most energy-efficient network. In a distribution network with delivery/pickup, distance is not the only determining factor that impacts fuel consumption. Parameters such as vehicle weight, road conditions, traffic conditions, and vehicle source of energy and aerodynamic characteristics can have substantial impact on the amount of energy consumed in the supply chain network. Hence, it is important to reformulate the LRP with the objective of minimizing energy and emissions. In the optimization model presented in Chapter 4, an emission cost is associated with consuming each unit of energy; hence, the objective function is to minimize the total network cost incurred from energy, emissions, and establishing DCs. In

---

1 http://ops.fhwa.dot.gov/freight/freight_analysis/nat_freight STATS/docs/10factsfigures/table5_9hist.htm
addition to the exact formulation for the LRP with an energy consideration, a heuristic algorithm is developed for the large-size problem in Chapter 5.

With the increasing price of fuel, railways are an inexpensive alternative method of transportation, which play an important role in supply-chain shipping. People in the railway industry always look for models and algorithms that can incorporate the main cost elements and constraints of railway networks. Considering all cost elements and constraints of a railway network makes the system-configuration problem more complex than a usual vehicle-routing problem. This complexity leads to inefficient computation times. Therefore, in the literature, only a limited number of cost elements are considered for the exact formulation of the problem. Hence, to continue the sustainability portion of this dissertation, the development of a heuristics algorithm for train routing is attempted. The fundamental difference between train routing and the VRP is discussed in detail in Chapter 6. The proposed algorithm is then used for solving two real-life problems. Results obtained from the case studies show the efficiency of the algorithm in terms of cost and computation time.

1.3 Research Objectives and Outlines

This dissertation is organized into seven Chapters. Chapters 2 to 6 each pursue a specific version of a supply chain transportation problem with a different objective. However, their overall objective is to investigate mathematical models and heuristics required to minimize the total cost of a supply chain network.

The main contribution of Chapter 2 is the consideration of dynamic demands in the location-routing problem. To do so, a novel formulation is presented for solving LRP problems with time windows. To illustrate the model, case studies are solved and the results compared with the existing formulation of the problem. The proposed non-linear formulation is then linearized, and different solution strategies are tested.

In Chapter 3, the objective is to formulate the LRP when travel times are dynamic. Also, the mathematical model developed eliminates waiting time at the customer location. In this chapter, it is shown that the formulation presented is also capable of solving the problem with time windows. The formulation is also advanced to the case in which customers’ demands are dynamic. The proposed non-linear formulation is linearized, and a best-solution strategy is found for small-size problems.

In Chapter 4, the energy-efficient LRP (EELRP), which reflects the impacts of energy consumption in the LRP, is formulated. In the calculation of energy, both rolling resistance and aerodynamic drag forces are considered. The formulation presented in this chapter can also handle time-window restrictions, tour-duration
constraints, energy limits, and dynamic demands. This formulation is the first in the literature that can differentiate among vehicles, not only in terms of their capacity but also in terms of other characteristics, such as their source of energy and aerodynamic characteristics.

In Chapter 5, a heuristics algorithm for the VRP is presented, with the objective of energy minimization. Although the quality of exact formulation in proving the solution is better, an exact formulation when the problem size is large has limited capability and cannot overcome the dimension of the problem. Hence, a heuristics algorithm for application in large-size networks is proposed. This heuristic is a revised version of the Clarke-Wright (CW) “savings” algorithm. Several benchmark problems from the literature are selected and customized to test the algorithm.

Chapter 6, which is a continuation of the sustainability context in this research, provides a heuristic algorithm for the rail transportation sector in a supply chain. The objective here is to find the best train schedules that minimize total system cost. In the literature, there are a limited number of cost elements contributing to the total network cost. However, in the proposed algorithm, the majority of real-life cost elements are present. This algorithm is applied to solving two real-life problems that are feasible as well as cost- and time-effective.

Finally, in Chapter 7, the conclusions and insight for future work are presented.

1.4 References


CHAPTER 2

GENERIC FORMULATION OF LOCATION-ROUTING PROBLEM WITH DYNAMIC DEMAND

2.1 Abstract

The vehicle-routing problem with a time window (VRPTW) is a class of vehicle-routing problem in which customers’ demands have a specific function of time. This has been extensively investigated by researchers. According to this function, a vehicle must reach a customer within a time window specified by the customer; otherwise, the customer demand expires. Although the VRPTW has been addressed by researchers, other forms of demand function varying by time have not been investigated. Hence, this paper attempts to formulate the problem when customers’ demands change after the initiation as a function of time. This problem in conjunction with the location-allocation problem (LAP) is called the location-routing problem with time-dependent demand (LRPTD). The objective here is to minimize the total network cost by finding the best network configuration. This includes finding the best strategy with respect to location, allocation, and routing plan. The problem formulation eliminates waiting times and can also handle problems with time windows. The presented mixed-integer non-linear model is then linearized and solved using CPLEX. Both branch-and-bound and cutting approaches are used for solving the model. Results show that the moderate cutting strategy provides results faster than other approaches tested for small-size problems.

Keywords: Location-Routing Problem, Vehicle-Routing Problems with Time Windows, Dynamic Demand

2.2 Introduction

The location-routing problem (LRP) is usually considered a class of supply chain network configuration problems. In fact, four major problems, which encompass both operational and strategic aspects, must be tackled in a supply chain network: production, location-allocation, inventory, and transportation (Arntzen et al., 1995). Among these decisions, location-allocation and location-routing problems are addressed in the LRP. The location-allocation problem is the problem of locating a set of potential facilities and allocating customers to those locations with the objective of minimizing total cost. The vehicle-routing problem (VRP), which originated from the traveling salesman problem (TPS), is defined as the problem of finding a set of routes originating from a set of depots to serve a set of customers with known demands. Each customer must be visited only once, and all vehicles must return to the depot from which they departed. Customer demands in a route should not exceed the vehicle capacity (Larson and Odoni, 1981). The vehicle-routing problem with time window (VRPTW), a class of the VRP
in which each customer must be visited within a specific time window, has been an interesting subject of research in the last three decades.

There are many heuristic/meta-heuristic exact methods developed for solving the VRPTW problem under different conditions and constraints. Braysy and Gendreau present a comprehensive survey on heuristic (2005a) and meta-heuristic (2005b) algorithms developed for solving the VRPTW. Desrochers et al. (1987) provide an early survey on the solution methods of the VRP. Although, there have been efforts on solution approaches to various formulations of this problem, not much effort has gone into developing a formulation that can deal with variations in demand as a function of time. Kallehauge (2008) reviews the formulation and exact algorithm of the VRPTW. He categorizes the formulation and exact methods into four major categories: arc formulation, arc-node formulation, spanning tree formulation, and path formulation. In arc formulation of VRPTW, “each arc of an underlying directed graph is associated with a binary variable.” Dantzig et al. (1954), Kallehauge et al. (2007), and Mak and Ernst (2007) present an arc formulation of the VRPTW. In the arc-node formulation of the problem, binary variables are associated with nodes of the directed graph. This method of formulating the VRPTW can be found in the work of Miller et al. (1960) and Bard et al. (2002). Briefly, the spanning tree formulation is “a method to find lower bounds for the VRPTW, with the help of time and capacity constrained shortest spanning trees and Lagrangian relaxation or Dantzig-Wolfe decomposition” (Held and Karp, 1970 and 1971). Path formulation is “a method to find lower bounds for the VRPTW with the help of time and capacity constrained shortest paths and Lagrangian relaxation or Dantzig-Wolfe decomposition.” In the last two decades, several researchers have focused on solutions to the path formulation approach (Chabrier, 2006; Cook and Rich, 1999; Danna and Pape, 2005; Desrochers et al., 1992; Feillet et al., 2004; Fisher et al., 1997; Halse, 1992; Houck, 1978; Kohl and Madsen, 1997; Kolen et al., 1987; Larsen, 1999, 2004).

In the existing combinatorial formulations of the VRPTW, the order of visits for a vehicle cannot be determined unless the arcs are connected in the right order. The interpretation of the sequence of visits in each route is thus obtained after the solution is obtained and hence cannot be used as an input when the costs incurred depend on parameters affected by sequence of customer visits in a route, e.g., demand. It is felt that the node-based formulation of the problem overcomes this deficiency, and whenever the sequence of visits results in any risks or costs associated with them, then the node-based formulation is a better approach. Previous researchers have used time-dependent demand functions. However, in those formulations, only one type of time-dependent demand
function can be used. Thus, formulations were restricted to a single static type of demand function (equation (2.1)). Thus, the literature contains no formulation where customers’ demand functions are dynamic in a given period, i.e., customers’ demand change after initiation by time.

In the case of dynamic customer demand functions, if all customers’ demand functions monotonously decrease with time (demand functions are similar for all customers), then the problem can be modeled and solved as a shortest-path problem. The model presented in this chapter takes into consideration different functions for customers’ demand. It also considers customers with simultaneous increasing and decreasing demands. If the demand functions are different and some customers’ demands are increasing with time, then the increase in demand may lead to situations in which the demand can no longer be satisfied due to the vehicle capacity constraint. Therefore, it is necessary to prevent the total demand from exceeding the vehicle capacity. Thus, the delay in serving customers with the increasing demand function may not always be cost-effective. The demand functions for each customer are initiated as soon as an order is received. In the examples shown in this chapter, all demands are assumed to be initiated at time zero. Infeasible conditions that may occur when customer demands are dynamic are also discussed.

In the existing arc formulations of the VRPTW (e.g., Desrosiers and Lubbeck, 2005), the arrival time at a customer is calculated based on two conditions: (1) the sum of the arrival time at a customer and the travel time from the current customer to the next customer should be less than the finish time for service at the next customer; and (2) the service start time at each customer should be within the time interval. This may lead to waiting time for trucks. For instance, consider that the truck is at the current customer and is at time 40. The service time window for the next customer is [100, 160]. The travel time to the next customer is 20 minutes. Based on arc formulation conditions, the arrival time at the next customer can be any time between 100 and 160. Thus, in this case, the truck must wait at least 40 minutes before it can serve the customer. In the formulation proposed in this chapter, this deficiency has been addressed as well. The objectives of this chapter are to develop the following:

- A novel formulation of the LRP in which customers’ demands can be any arbitrary function of arrival time and minimizes total cost while maximizing profit.
- A node formulation for the LRP with time-dependent demand.

A model that is capable of solving the VRPTW. Based on the formulation, it is expected that the following questions can be answered:
What is the best strategy for locating distribution centers (DCs)?

How are customers allocated to DCs?

What is the routing plan of DCs to serve customers?

How can infeasible solutions be handled because of dynamic customer demands?

Section 2.3 provides a detailed definition of the problem under investigation. Section 2.4 is devoted to deriving the mathematical formulation of the LRPTD. The proposed formulation is illustrated and also validated by comparing results with the VRPTW in Section 2.5. In section 2.6, additional examples are investigated to illustrate the infeasibility cases, which occur as a result of the dynamic nature of the problem. Section 2.7 provides the linearization approach and computation results. Section 2.8 provides a summary, conclusions, and discussion of future work.

### 2.3 Problem Statement

This section presents notations used in the formulation of the problem. The network for this problem includes set $I$ consisting of $N$ customers, and set $J$ consisting of $M$ potential DCs. The collective set of customers and DCs in the networks is represented by nodes, wherein nodes 1 to $N$ represent customers and nodes $N+1$ to $N+M$ represent potential DCs. The position of a node in the route is the order in which the node is visited by the vehicle. For instance, if node $g$ is in position 2 of vehicle 1’s route, then that node $g$ is the second node visited by vehicle 1. The set of possible positions that a customer can take in a route is $D$. At the most, a vehicle can visit all customers. Thus, the maximum number of possible positions for a customer is equal to $N$. Therefore, decision variables, $X_{mgv} = 1$, implies that node $g$ (a customer at a DC) is the $m$th node visited by vehicle $v$. In other words, node $g$ is the $m$th node in the route assigned to vehicle $v$. Thus, in the following definition of notations, vehicle and route are used alternatively, since they represent the same concept.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>Total number of customers</td>
</tr>
<tr>
<td>$M$</td>
<td>Total number of DCs</td>
</tr>
<tr>
<td>$K$</td>
<td>Total number of vehicles</td>
</tr>
<tr>
<td>$J$</td>
<td>Set of DCs, $J = {1,2,...,M}$</td>
</tr>
<tr>
<td>$I$</td>
<td>Set of customers, $I = {1,2,...,N}$</td>
</tr>
<tr>
<td>$D$</td>
<td>Set of possible positions that a customer can take in a route, $D = {1,2,...,N}$</td>
</tr>
<tr>
<td>$V$</td>
<td>Set of available vehicles, $V = {1,2,...,K}$</td>
</tr>
</tbody>
</table>
\( Y_v \) Capacity of vehicle \( v \)

\( g/h \) Index for all nodes including customers and DCs, \( g/h \in \{I \cup J\} \)

\( T_{gh} \) Travel time between nodes \( g \) and \( h \), \( \forall g, h \in \{I \cup J\} \)

\( S_g \) Service time at node \( g \), \( \forall g \in I \)

\( A_{mv} \) Arrival time at position \( m \) of route \( v \), \( \forall m \in I, \forall v \in V \)

\( X_{mgv} \) 1 if node \( g \) (customer or DC) is in position \( m \) of route \( v \); 0 otherwise, \( \forall g, m \in I, \forall v \in V \)

\( P_{mv} \) 1 if \( m \) is the last taken position of route \( v \); 0 otherwise, \( \forall m \in I, \forall v \in V \)

\( O_g \) 1 if there is any route assigned to node \( g \); 0 otherwise, \( \forall g \in J \)

\( z_{vh} \) 1 if route \( v \) is assigned to DC \( l \); 0 otherwise, \( \forall h \in J, \forall v \in V \)

\( C_{gh} \) Transportation cost between node \( g \) and \( h \), \( \forall g, h \in \{I \cup J\} \)

\( q_g \) Fixed cost for establishing node \( g \), \( \forall g \in J \)

\( d_g \) Initial demand of node \( g \), \( \forall g \in I \)

\( f_g(t) \) Demand function of node \( g \), \( \forall g \in I \)

\( B_g \) Departure cost from DCs, \( \forall g \in J \)

\( E_g \) Customer \( g \) fulfillment level, \( \forall g \in I \)

\( a_{mv} \) 1 if earliest arrival at position \( m \) of route \( v \) is violated; 0 otherwise, \( \forall m \in I, \forall v \in V \)

\( \beta_{mv} \) 1 if latest arrival at position \( m \) of route \( v \) is violated; 0 otherwise, \( \forall m \in I, \forall v \in V \)

\( a_g \) Earliest arrival time at customer \( g \), \( \forall g \in I \)

\( b_g \) Latest arrival time at customer \( g \), \( \forall g \in I \)

\( \Delta a_g \) Maximum deviation permitted from the earliest arrival time at customer \( g \), \( \forall g \in I \)

\( \Delta b_g \) Maximum deviation permitted from the latest arrival time at customer \( g \), \( \forall g \in I \)

\( w^1_g \) 1 if demand has decreased at delivery time to customer \( g \); 0 otherwise, \( \forall g \in I \)

\( w^2_g \) 1 if demand has increased at delivery time to customer \( g \); 0 otherwise, \( \forall g \in I \)

\( \lambda \) Lost-order cost

\( \gamma \) Percentage change in unit price of extra product delivered
Ω Profit obtained from selling a unit of product

ρ_\text{g} Penalty costs associated with lower time limit violation at customer \text{g}, \forall \text{g} \in \text{I}

φ_\text{g} Penalty costs associated with upper time limit violation at customer \text{g}, \forall \text{g} \in \text{I}

It is assumed that vehicles depart the distribution centers fully loaded and that this problem has only a single type of product. Each customer has an initial demand, \text{d}_\text{g}. The demand is expected to change from the time of its initiation, i.e., \text{f(d}_\text{g},\text{t}_\text{g}), \forall \text{g} \in \text{I}. In this special case of the problem, VRPTW, the demand function is defined by equation (2.1):

\[ f(\text{t}_\text{g}) = \begin{cases} 0 & \text{if } \text{t}_\text{g} \leq \text{a}_\text{g} \\ \text{d}_\text{g} & \text{if } \text{a}_\text{g} \leq \text{t}_\text{g} \leq \text{b}_\text{g} \\ 0 & \text{if } \text{t}_\text{g} \geq \text{b}_\text{g} \end{cases} \quad (2.1) \]

where \text{a}_\text{g} is the earliest arrival time at node \text{g}, and \text{b}_\text{g} is the latest arrival time at node \text{g}. Also, based on the basic assumptions of vehicle-routing problems, \text{D}_\text{g} \leq \text{VC}, \forall \text{g} \in \text{I}.

In equations (2.2), (2.3), (2.4), and (2.5), \text{A}_{\text{mv}}, \text{t}_\text{g}, \text{P}_{\text{mv}}, and \text{z}_{\text{ih}}, respectively, are variables defined to simplify the objective function and constraints. Considering the fact that the arrival time at each customer depends on the arrival times at previous customers on the same route, the arrival time at position \text{m} of route \text{v} is calculated by

\[ A_{\text{mv}} = \sum_{\text{m}=1}^{\text{N}} \sum_{\text{h} \in \text{I}} \sum_{\text{g} \in \text{I}} X_{\text{m}-1,\text{h},\text{g}} X_{\text{m},\text{h}} (T_{\text{gh}} + S_{\text{gh}}) \sum_{\text{g} \in \text{I}} X_{\text{mgv}} \quad \forall \text{m} \in \text{D}, \text{v} \in \text{V} \quad (2.2) \]

Thus, the arrival time at customer \text{g} can be calculated using

\[ \text{t}_\text{g} = \sum_{\text{v} \in \text{V}} \sum_{\text{m} \in \text{D}} X_{\text{mgv}} A_{\text{mv}}, \forall \text{g} \in \text{I} \quad (2.3) \]

The variable \text{P}_{\text{mv}}, defined in equation (2.4) to simplify the objective function, indicates whether or not \text{m} is the last taken position on route \text{v}.

\[ \text{P}_{\text{mv}} = \begin{cases} \sum_{\text{g} \in \text{I}} X_{\text{mgv}} & \text{m} = \text{N} \\ \prod_{\text{m} = \text{m}+1}^{\text{N}} (1 - \text{P}_{\text{mv}}) \sum_{\text{g} \in \text{I}} X_{\text{mgv}} & \forall \text{m} \in \{ \text{D} / \{ \text{N} \} \} \end{cases} \quad (2.4) \]

Location 0 of each route is reserved for a DC. However, even if the related binary variable \text{X}_{0,\text{g},v}, \forall \text{g} \in \text{L}, \forall \text{v} \in \text{V} holds a value of 1, this does not mean that node \text{g} is selected to be the assigned DC for
route \( v \). Node \( g \) will be route \( v \)'s DC, only if there is a link between the depot and a customer in the network. Hence, the variable \( z_{vh} \), in equation (2.2), is introduced to specify whether there is a connection between a depot and a customer in the system. This variable acts as a connectivity constraint between DCs and routes, and connects the location decision to the routing decision.

\[
 z_{vh} = \sum_{g \in I} X_{0vh} X_{1gv} \quad \forall h \in J, v \in V 
\]  

(2.5)

2.4 Mathematical Formulation of Problem

2.4.1 Formulation of LRP with Time-Dependent Customer Demands

The objective function of this problem minimizes the total cost of the system, including transportation cost and depot-establishment cost, while maximizing profit. Customers with a decreasing demand function have an associated “lost-order cost,” which is the cost resulting from not meeting a customer’s demand completely or partially, i.e., \( (d_g - f_g(\tau_g)) \lambda \), where \( \lambda \) is the lost-order cost per unit of product. The profit is the product of total quantity delivered to the customer, profit per unit, and customer-fulfillment level, i.e., \( f_g(\tau_g) E_g \Omega \), in which \( E_g \) is the customer-fulfillment level defined by equation (2.3):

\[
 E_g = \frac{f_g(\tau_g)}{d_g} \quad \forall g \in I 
\]  

(2.6)

The value of \( B \) is dynamic and depends on the time of delivery. The value of \( E_g \) is greater than one for customers with a monotonously increasing demand function, and is less than one for customers with a monotonously decreasing demand function, \( E_g \). The unit product price for the additional number of products delivered to customers with increasing demand can be different from the initial price. For instance, the price can be cheaper due to quantity discount. Hence, \( \gamma \) is the constant representing the percentage of decrease or increase in the price.

The mathematical formulation for the LRPTD to minimize the network cost is as follows:

**Objective Function:**

\[
 \begin{align*}
 & \text{Min} \\
 & \sum_{v \in V} \sum_{w \in D} \sum_{h \in [I,J]} \sum_{g \in [I,J]} \sum_{r \in [1,R]} \\
 & \quad C_{gh} \left( X_{m-igr} X_{mhr} + X_{mgr} X_{0hr} P_{mr} \right) + \sum_{v \in V} \sum_{w \in D} \sum_{h \in J} \sum_{g \in I} z_{gh} B_g + \sum_{g \in I} q_g O_g + \\
 & \quad \sum_{g \in I} \left( w_g \lambda (d_g - f_g(\tau_g)) \pm (1 - w_g) \Omega \gamma (f_g(\tau_g) - d_g) - \Omega E_g f_g(\tau_g) \right) 
\end{align*} 
\]  

(2.7)
Subject to:

\[ \sum_{v \in V} \sum_{m \in D} X_{mgv} = 1 \quad \forall g \in I \quad (2.8) \]

\[ \sum_{g \in I} X_{mgv} \leq 1 \quad \forall m \in D, v \in V \quad (2.9) \]

\[ \sum_{g \in I} \sum_{m \in D} \sum_{v \in V} f_g (\tau_g) X_{mgv} \leq Y_v \quad \forall v \in V \quad (2.10) \]

\[ \sum_{v \in V} \sum_{g \in I} \sum_{m \in D} X_{mgv} = 0 \quad (2.11) \]

\[ \sum_{v \in V} \sum_{g \in I} Y_{0gv} = 0 \quad (2.12) \]

\[ \sum_{g \in I \setminus \{1, \ldots, j\}} X_{m-1gv} \geq \sum_{g \in \{1, \ldots, j\}} X_{mgv} \quad \forall m \in D, v \in V \quad (2.13) \]

\[ \sum_{g \notin j} z_{vg} \leq 1 \quad \forall v \in V \quad (2.14) \]

\[ O_g \leq \sum_{v \in F} z_{vg} \leq K_0 \quad \forall g \in L \quad (2.15) \]

\[ 1 \leq \sum_{v \in F} \sum_{g \in I} z_{vg} \leq K \quad (2.16) \]

\[ w_g^1 \left( d_g - f_g (\tau_g) \right) \geq 0 \quad \forall g \in I \quad (2.17) \]

\[ w_g^2 \left( d_g - f_g (\tau_g) \right) < 0 \quad \forall g \in I \quad (2.18) \]

\[ w_g^1 + w_g^2 = 1 \quad \forall g \in I \quad (2.19) \]

Equation (2.4) is the objective function that minimizes the total network cost, including transportation cost, fixed cost, and customer lost-order cost, while maximizing the profit. The first term determines the transportation cost. The second term is the cost of dispatching vehicles from DCs. The third term includes the fixed DC transportation and establishment cost. The fourth, fifth, and sixth terms are lost-order cost, additional-order cost/profit, and sales profit, respectively.
Equation (2.5) ensures that each customer appears in only one route, i.e., only one route is assigned to each customer. Constraint (2.6) enforces that each position of a route is not taken by more than one customer. Constraint (2.7) ensures that the total demand of customers assigned to a route is less than the vehicle capacity. It is assumed that position zero of each route is reserved for DCs. This assumption implies that DCs cannot take any other position in the route and also that customers cannot take position 0 of their assigned route. The former is enforced by constraint (2.8), and the latter is enforced by constraint (2.9). Constraint (2.10) ensures that position \( m+1 \) of a route cannot be taken unless position \( m \) is taken. Constraint (2.11) is used to ensure that a route cannot be assigned to more than one DC. Constraint (2.12) limits the value of the decision variable related to the establishment of a DC between 0 and 1. Constraint (2.13) keeps the total number of routes between one and the number of available vehicles. Constraints (2.14) to (2.16) are used to determine whether a customer’s demand has increased or decreased at delivery time.

### 2.4.2 Formulation of LRP with Time Windows

A vehicle-routing problem in which each customer’s demand must be satisfied within a time interval is called the VRPTW. The time window can be hard or soft. In a formulation with a “hard time window,” each customer has a specified time window during which the demand must be met. The vehicle cannot deliver products to the customer before the start of the time window or after the time window has elapsed, i.e., late or early arrival at a customer is not acceptable. In a VRPTW formulation with a “soft time window,” the customer can be served before and after the preferred time window, i.e., early or late arrival at a customer is acceptable up to a predefined limit. However, with the violation of the time window, there is usually an associated penalty cost, which results in late or early service to the customer. In this section, the formulation proposed in section 2.4.1 is modified to tackle the LRP with hard and soft time windows.

#### 2.4.2.1 Formulation of LRP with Hard Time Windows

Since the LRPTD is a generic form of the VRPTW, the model presented in section 2.4.1 must be able to solve the VRPTW as well. It is also expected that the result obtained from the proposed formulation and the existing models for the VRPTW must be comparable.

In the VRP with time-window constraints, a customer’s demand does not dynamically change with time and is either equal to the initial demand or zero, depending on the arrival time, as shown in equation (2.1). When the time window is hard, no violation from the time intervals is acceptable. One way of solving the VRPTW with the
hard time window is to use the proposed LRPTD formulation. For this purpose, equation (2.1), which represents the customers’ demand function must be substituted in the mathematical model presented in section 2.4.1. The other alternative is to insert constraints in the LRPTD model to enforce the arrival times within the required time windows, as presented in constraint (2.20):

\[
\sum_{g \in I} a_g X_{mgv} \leq A_{mv} \leq \sum_{g \in I} b_g X_{mgv} \quad \forall m \in D, v \in V
\]  

(2.20)

where \(a_g\) and \(b_g\) represent the lower and upper bounds, respectively, of node \(g\)’s time window. In this case, the demand function in constraint (2.10) must be replaced by the initial demand, as shown in constraint (2.21):

\[
\sum_{g \in I} \sum_{m \in D} d_g X_{mgv} \leq Y_v \quad \forall v \in V
\]  

(2.21)

When the LRPTD is used to solve a LRP with hard time windows, the objective function presented in equation (2.7) must be replaced by equation (2.22). This is necessary because in the VRPTW, all customers’ initial demands are met.

\[
Min \sum_{v \in V} \sum_{m \in D} \sum_{h \in \{I,J\}} C_{gh} \left( X_{m-1,v}X_{mbv} + X_{mgv}X_{bvh}P_{mv} \right) + \sum_{v \in V} \sum_{g \in I} z_{vg} B_g + \sum_{g \in I} q_g O_g
\]  

(2.22)

2.4.2.2 Formulation of LRP with Soft Time Windows

In addition to solving the VRP with a hard time window, the model can also handle problems with soft time window constraints. The VRP with a soft time window allows for time-window violations including the assigned penalty cost. Similar to the VRP with a hard time window, customers’ demand does not dynamically change with time and is equal to an initial demand or zero at the time of service. This condition is presented in equation (2.23):

\[
f(\tau_g) = \begin{cases} 
0 & \text{if } \tau_g \leq a_g \\
d_g & \text{if } \Delta a_g + a_g \leq \tau_g \leq b_g + \Delta b_g \quad \forall g \in I \\
0 & \text{if } \tau_g \geq b_g 
\end{cases}
\]  

(2.23)

One way to solve the VRP with a soft time window is to substitute all customers’ demand functions in the mathematical model with functions similar to those in equation (2.23).

Another alternative is to add a set of constraints to impose the arrival time conditions at each customer. These conditions are presented in equation (2.24):
\[
\sum_{g \in D} a_g X_{mgv} - \left( \sum_{g \in D} \Delta a_g X_{mgv} \right) \rho_g \alpha_{mv} \leq A_{mv} \leq \sum_{g \in D} b_g X_{mgv} + \left( \sum_{g \in D} \Delta b_g X_{mgv} \right) \beta_{mv} \quad \forall m \in D, v \in V
\]  

(2.24)

These constraints are used to enforce the arrival time at each customer within the flexible time window. In this case, the demand function in constraint (2.10) must be replaced by initial demand as presented in constraint (2.21).

As opposed to a hard time window, a soft time window allows for a time interval violation with an assigned penalty cost. Therefore, the objective function has an extra term related to the penalty cost associated with the time-window violation. This new objective function term can be defined as

\[
\sum_{v \in V} \sum_{m \in D} \sum_{g \in D} X_{mgv} \left( \rho_g \alpha_{mv} + \varphi_g \beta_{mv} \right)
\]

(2.25)

In the following section, examples illustrate the models presented.

2.5 **Illustrative Examples**

2.5.1 **Example of LRPTD**

As an example to illustrate the proposed mathematical model, Figure 2.1 shows a two-layer network problem consisting of two DCs and four customers. Nodes 1, 2, 3, and 4 represent the customers, and nodes 5 and 6 represent potential DCs. Coordinates of the nodes, their related initial demand, and the demand function are shown in Table 2.1. The demand of each customer is dynamic and starts changing as soon as the customer indicates demand. In this example, it is assumed that all demands are initiated at time zero, and no demand is associated with depots.

Figure 2.1. Two-layer network problem with four customers (nodes 1–4) and two DCs (nodes 5–6)
TABLE 2.1 CUSTOMERS’ DEMAND INFORMATION FOR TWO-LAYER NETWORK PROBLEM
WITH FOUR CUSTOMERS (NODES 1–4) AND TWO DCS (NODES 5–6)

<table>
<thead>
<tr>
<th>Node</th>
<th>Coordinate</th>
<th>Initial Demand</th>
<th>Demand Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(80,45)</td>
<td>20</td>
<td>$20 + 0.2\tau_1$</td>
</tr>
<tr>
<td>2</td>
<td>(70,15)</td>
<td>5</td>
<td>$5 - 0.2\tau_1$</td>
</tr>
<tr>
<td>3</td>
<td>(100,20)</td>
<td>13</td>
<td>$13 + 0.3\tau_1$</td>
</tr>
<tr>
<td>4</td>
<td>(90,10)</td>
<td>12</td>
<td>$12 - 0.5\tau_1$</td>
</tr>
<tr>
<td>5</td>
<td>(75,10)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>(60,25)</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The Euclidean distance formula is used for calculating distance in this problem. The travel cost per unit of distance is assumed to be $0.1, and vehicles are assumed to travel at a constant speed of 40 miles per hour. The cost and time of travel between nodes is shown in Tables 2.2 and 2.3. The two available vehicles are homogenous with a capacity of 30 units each. The cost of vehicle departure from node 5 is $45 and from node 6 is $50. In the planning horizon of this example, the transportation cost from the plant to dc 1 (node 5) is $250 and to dc 2 (node 6) is $200. In addition, fixed costs for establishing dc 1 is $40 and dc 2 is $35. The lost-order cost is $1 per unit of product. Customers with increasing demand must pay 20% more than the original price of $10.

TABLE 2.2 TRANSPORTATION COST BETWEEN NODES ($ THOUSANDS)
FOR TWO-LAYER NETWORK PROBLEM WITH FOUR CUSTOMERS (NODES 1–4)
AND TWO DCS (NODES 5–6)

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>3.16</td>
<td>3.20</td>
<td>3.64</td>
<td>3.54</td>
<td>2.83</td>
</tr>
<tr>
<td>2</td>
<td>3.16</td>
<td>0</td>
<td>3.04</td>
<td>2.06</td>
<td>0.71</td>
<td>1.41</td>
</tr>
<tr>
<td>3</td>
<td>3.20</td>
<td>3.04</td>
<td>0</td>
<td>1.41</td>
<td>2.69</td>
<td>4.03</td>
</tr>
<tr>
<td>4</td>
<td>3.64</td>
<td>2.06</td>
<td>1.41</td>
<td>0</td>
<td>1.5</td>
<td>3.35</td>
</tr>
<tr>
<td>5</td>
<td>3.54</td>
<td>0.71</td>
<td>2.69</td>
<td>1.5</td>
<td>0</td>
<td>2.12</td>
</tr>
<tr>
<td>6</td>
<td>2.83</td>
<td>1.41</td>
<td>4.03</td>
<td>3.35</td>
<td>2.12</td>
<td>0</td>
</tr>
</tbody>
</table>

TABLE 2.3 TRANSPORTATION TIME BETWEEN NODES (HOURS)
FOR TWO-LAYER NETWORK PROBLEM WITH FOUR CUSTOMERS (NODES 1–4)
AND TWO DCS (NODES 5–6)

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
<td>1.0</td>
<td>3.0</td>
<td>3.5</td>
<td>2.0</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>1.0</td>
<td>0.0</td>
<td>2.5</td>
<td>1.5</td>
<td>4.0</td>
<td>2.5</td>
</tr>
<tr>
<td>3</td>
<td>3.0</td>
<td>2.5</td>
<td>0.0</td>
<td>4.5</td>
<td>3.5</td>
<td>2.0</td>
</tr>
<tr>
<td>4</td>
<td>3.5</td>
<td>1.5</td>
<td>4.5</td>
<td>0.0</td>
<td>4.5</td>
<td>1.0</td>
</tr>
<tr>
<td>5</td>
<td>2.0</td>
<td>4.0</td>
<td>3.5</td>
<td>3.5</td>
<td>0.0</td>
<td>1.0</td>
</tr>
<tr>
<td>6</td>
<td>1.0</td>
<td>2.5</td>
<td>2.0</td>
<td>5.0</td>
<td>1.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

This mathematical formulation was solved using LINGO 12.0 optimizer software on a Pentium D CPU 3.2 GHz with 3.25 GB of RAM. The computation time was 197 seconds. Two optimal solutions were obtained:
The objective value: $-176.11$

\[
X_{122} = X_{141} = X_{232} = X_{311} = X_{61} = X_{62} = 1 \quad \text{or} \quad X_{121} = X_{142} = X_{231} = X_{312} = X_{61} = X_{62} = 1
\]

All the other \(X_{mgv}\) variables are equal to zero

\[
z_{16} = z_{26} = 1
\]

The difference between these two solutions is in the assignment of vehicles to routes. However, vehicles 1 and 2 have the same capacity so the solutions are equivalent. Figure 2.2 shows the network configuration solution. Customers 1 and 2 are assigned to one route, and customers 3 and 4 are assigned to another. In this case, DC 2 (node 6) at location (60,25) is selected to be open; thus all customers will be served by this depot. The negative value of the objective function shows the network profit.

![Figure 2.2. Network configuration solution for two-layer network problem with four customers and two DCs](image)

2.5.2 Example of VRP with Hard Time Window

The model presented in this chapter is for the purpose of solving the TDLRP, not the VRPTW. However, to show that the model is formulated correctly and to justify its performance, a VRPTW is solved. The VRPTW is also solved using one of the existing formulations (Desrosiers and Lubbeck, 2005) in the literature. The results from the two models are compared. This case study is similar to the one presented in section 2.5.1. The difference is that there is only one DC (node 5) to serve the customers, as shown in Figure 2.3. Therefore, the case study is formulated without constraints (2.25), (2.26), and (2.27). Deletion of the location-allocation problem is for comparison purposes. In addition, all customers have an assigned time window assigned (Table 2.4). Each customer’s demand is equal to its initial demand, \(D_g\), if it is served within the specified time window; otherwise, it is zero. The result obtained from the presented formulation in section 2.4.1 is compared with the result obtained from the classic arc formulation of the problem (Desaulniers et al., 2005).
TABLE 2.4 TIME INTERVAL ASSIGNED TO EACH CUSTOMER FOR VRP WITH HARD TIME WINDOW

<table>
<thead>
<tr>
<th>Customer Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Interval (Hours)</td>
<td>[5,6]</td>
<td>[4,8]</td>
<td>[8,12]</td>
<td>[3,5]</td>
</tr>
</tbody>
</table>

The notations and node numbering in the arc formulation from the literature differ from the proposed formulation. In the arc formulation, if \( X_{ghv} = 1 \), then there is a link between node \( g \) and \( h \) on route \( v \). Moreover, in each route obtained, the first and last node represent a DC. Thus, nodes 1 and 6 in the arc formulation represent the DC. The problem using the formulations from Desaulniers et al. (2005) was solved using LINGO 12.0 optimizer software. The computation time was 0.001 seconds, and the results obtained are shown below:

- The objective value: \(-489.48\)

All the other \( X \) variables are equal to zero

This problem has two optimal solutions, and the results are the same because there is no difference between the vehicles in terms of capacity. However, if the vehicles are different, then each of these results might have different costs and hence different optimal solutions.

The results obtained from solving the problems using the proposed formulation are shown below; the computation time is 1 second.

![Network configuration solution for VRPTW from proposed and existing models](image)
The objective value: -489.48$

As shown in Figure 2.3, the results obtained from the arc and node formulations are the same; however, the computation time for the proposed formulation is more than the arc formulation. This is because of the large number of constraints and variables in the proposed formulation compared to the existing linear formulation of the VRPTW. It is important to note that the formulation presented in section 2.4.1 is developed to solve the LRPTD in which demand is a dynamic function of time. If demand is static and waiting times in the solution can be ignored, then existing VRPTW formulations are more computationally efficient. The purpose of this section is to solve a VRPTW as a benchmark for validating the proposed node formulation against the existing formulation in the literature.

### 2.5.3 Example of VRP with Soft Time Window

Here, a VRPTW is solved as a benchmark for validating the new formulation against the formulation existing in the literature. The proposed node formulation is compared with the formulation presented by Desrosiers and Lubbeck (2005) for the VRPTW. The example presented in this section is similar to the one presented in section 2.5.2. The difference is that the violation in customers’ time windows is acceptable up to a specified limit. These acceptable violations as well as the penalty cost associated with them are provided in Table 2.5. Since the problem investigated in section 2.5.2 has an optimal feasible solution with a hard time window constraint, solving the same problem with a soft window will conclude with the same result without violating the time windows. Hence, the time windows are tightened in this example to illustrate the soft time window constraint solution.

| TABLE 2.5. TIME INTERVAL ASSIGNED TO EACH CUSTOMER FOR VRP WITH SOFT TIME WINDOW |
|-----------------------------------------------|--------|--------|--------|--------|
| Customer Number               | 1      | 2      | 3      | 4      |
| Time Interval (Hours)         | [6, 6.5] | [4, 7] | [9, 12] | [3, 5] |
| Upper Limit Violation Allowed (Hours) | 1      | 0.5    | 1.2    | 0.4    |
| Lower Limit Violation Allowed (Hours) | 0.5    | 0.75   | 0.5    | 1      |
| Penalty Cost $\rho_g$ ($)     | 2      | 1      | 5      | 3      |
| Penalty Cost $\phi_g$ ($)     | 5      | 7      | 2      | 1      |

This problem was solved with LINGO 12.0 optimizer software, first by using the existing arc formulation of the VRP with a soft time window and then using the proposed formulation. Figure 2.4 shows the results. The computation time for the arc formulation is 0.0012 second. The arc formulation solutions are given below:
Two results were obtained from the proposed formulation. Since the vehicles are homogeneous, both solutions have the same objective function value. The computation time for solving the proposed formulation is 211 seconds. The solutions obtained by the proposed formulation are given below:

When the results from the existing arc formulation of the VRP with a soft time window are compared to the proposed formulation, it is noted that the two models have the same routing plan but different objective function values. The arc formulation provides an objective function value of -$482.48, and the proposed method has an objective function value of -$475.48. As can be seen, the arc formulation provides higher profits. An analysis of the solution shows that the arc formulation results in waiting times. No penalty costs are associated with these waiting times in the arc formulation. In fact, in the arc formulation of the problem, the vehicle waits until the arrival time is current to avoid early arrival at a customer. Since no penalty cost is associated with this delay, the model chooses to wait rather than arriving at a customer earlier and paying penalty costs. Allowing waiting times in the arc formulation leads to a better objective function value. The waiting times are not always realistic or practical, especially when time units are large, such as days, months, etc. In the solution obtained from the arc formulation, the total waiting time is 2.75 hours. Table 2.6 shows the arrival times for the proposed and arc formulations.

### Table 2.6 Actual and Recorded Arrival Times for Proposed and Arc Formulations

<table>
<thead>
<tr>
<th>Customer</th>
<th>Actual Arrival Time for Proposed Formulation (Hours)</th>
<th>Recorded Arrival Time for Arc Formulation (Hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>3.5</td>
<td>4.25</td>
</tr>
</tbody>
</table>

The computation time for the arc formulation is lower than the proposed formulation. Thus, if the demand is static and the waiting times in the solution can be ignored, then existing VRPTW formulations are more
computationally efficient. A comparison of the solution obtained from the LRP with a soft time window to the LRP with a hard time window was also conducted. Although the network configuration obtained for the VRP with a soft time window (Figure 2.4) is similar to the result from the case study for the LRP with a hard time window, the objective function value is worse. This is because, in the case study for soft time windows, the time windows are tightened, and the model uses the flexibility in the soft time window formulation to provide a feasible solution with minimum cost. Thus, these time window penalty costs contribute to the worsening of the objective function value.

Figure 2.4. Network configuration solution for VRP with soft time window using proposed formulation

2.6 Additional Case Studies and Constraint Relaxations for Infeasibility

In this section, additional examples, conditions that might lead to infeasibility, and possible remedies are discussed. When demand is not dynamic, infeasibility can be easily predicted by looking at customer demands and vehicle capacities. However, when demand is dynamic, the real value at the time of delivery is not known beforehand, and this could lead to infeasibility conditions that are not intuitively apparent. When the problem being investigated does not provide a feasible solution, relaxation of constraints may be necessary to arrive at a feasible solution. The examples below are used to illustrate how the relaxation of constraints can help to avoid infeasible solutions. Some of the possible remedies for infeasibility can be relaxation of the following constraints:

- Vehicle capacity
- Number of vehicles
- Requirement for serving all customers
**Relaxation of vehicle capacity constraint:** When the formulation presented in section 2.4.1 cannot provide a feasible solution for the problem, one way to avoid infeasibility is to increase vehicle capacities and solve the problem again. This option is practical when alternative vehicles with higher capacity are available. An example of this case is presented here:

Consider a network problem with two DCs and five customers, as shown in Figure 2.5. Nodes 1, 2, 3, 4, and 5 represent the customers and nodes 6 and 7 represent potential DCs.

![Two-layer network problem with five customers and two DCs](image)

**Figure 2.5.** Two-layer network problem with five customers (nodes 1–5) and two DCs (nodes 6–7)

Node coordinates, initial demand, demand function, and service time are presented in Table 2.7. No demand is associated with the DCs. Distance is calculated using the Euclidean method, and travel cost per unit of distance traveled is $1. It is assumed that the vehicles travel at a constant speed of 40 miles per hour. The cost and time of traveling between nodes is shown in Tables 2.8 and 2.9. Two vehicles with a capacity of 30 units each are available. The vehicle departure cost from node 6 or 7 is $10. The fixed cost for establishing either DC 1 or DC 2 (nodes 6 and 7) is $30. The lost-order cost is $1 per unit of product. Customers with increasing demand pay 20% less than the original price of $10 due to the quantity discount.

<table>
<thead>
<tr>
<th>Node</th>
<th>Coordinate</th>
<th>Initial Demand</th>
<th>Demand Function</th>
<th>Service Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(10,120)</td>
<td>15</td>
<td>15+2.72^{11}</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>(50,140)</td>
<td>16</td>
<td>16+5(τ_1)^{0.5}</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>(30,90)</td>
<td>12</td>
<td>12-2τ_3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>(60,80)</td>
<td>10</td>
<td>10+5τ_4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>(50,10)</td>
<td>8</td>
<td>8+(τ_5)^{0.5}</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>(50,50)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>(80,30)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**TABLE 2.7 CUSTOMERS’ DEMAND INFORMATION FOR TWO-LAYER NETWORK PROBLEM WITH FIVE CUSTOMERS (NODES 1–5) AND TWO DCs (NODES 6–7)**
TABLE 2.8 TRANSPORTATION COST BETWEEN NODES ($ THOUSANDS) FOR TWO-LAYER NETWORK PROBLEM WITH FIVE CUSTOMERS (NODES 1–5) AND TWO DCS (NODES 6–7)

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00</td>
<td>44.72</td>
<td>36.06</td>
<td>64.03</td>
<td>117.05</td>
<td>80.62</td>
<td>114.02</td>
</tr>
<tr>
<td>2</td>
<td>44.72</td>
<td>00.00</td>
<td>53.85</td>
<td>60.83</td>
<td>130.00</td>
<td>90.00</td>
<td>114.02</td>
</tr>
<tr>
<td>3</td>
<td>36.06</td>
<td>53.85</td>
<td>00.00</td>
<td>31.62</td>
<td>82.46</td>
<td>44.72</td>
<td>78.10</td>
</tr>
<tr>
<td>4</td>
<td>64.03</td>
<td>60.83</td>
<td>31.62</td>
<td>00.00</td>
<td>70.71</td>
<td>31.62</td>
<td>53.85</td>
</tr>
<tr>
<td>5</td>
<td>117.05</td>
<td>130.00</td>
<td>82.46</td>
<td>70.71</td>
<td>00.00</td>
<td>40.00</td>
<td>36.06</td>
</tr>
<tr>
<td>6</td>
<td>80.62</td>
<td>90.00</td>
<td>44.72</td>
<td>31.62</td>
<td>40.00</td>
<td>00.00</td>
<td>36.06</td>
</tr>
<tr>
<td>7</td>
<td>114.02</td>
<td>114.02</td>
<td>78.10</td>
<td>53.85</td>
<td>36.06</td>
<td>36.06</td>
<td>00.00</td>
</tr>
</tbody>
</table>

TABLE 2.9 TRANSPORTATION TIME BETWEEN NODES (HOURS) FOR TWO-LAYER NETWORK PROBLEM WITH FIVE CUSTOMERS (NODES 1–5) AND TWO DCS (NODES 6–7)

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00</td>
<td>1.12</td>
<td>0.90</td>
<td>1.60</td>
<td>2.93</td>
<td>2.02</td>
<td>2.85</td>
</tr>
<tr>
<td>2</td>
<td>1.12</td>
<td>0.00</td>
<td>1.35</td>
<td>1.52</td>
<td>3.25</td>
<td>2.25</td>
<td>2.85</td>
</tr>
<tr>
<td>3</td>
<td>0.90</td>
<td>1.35</td>
<td>0.00</td>
<td>0.79</td>
<td>2.06</td>
<td>1.12</td>
<td>1.95</td>
</tr>
<tr>
<td>4</td>
<td>1.60</td>
<td>1.52</td>
<td>0.79</td>
<td>0.00</td>
<td>1.77</td>
<td>0.79</td>
<td>1.35</td>
</tr>
<tr>
<td>5</td>
<td>2.93</td>
<td>3.25</td>
<td>2.06</td>
<td>1.77</td>
<td>0.00</td>
<td>1.00</td>
<td>0.90</td>
</tr>
<tr>
<td>6</td>
<td>2.02</td>
<td>2.25</td>
<td>1.12</td>
<td>0.79</td>
<td>1.00</td>
<td>0.00</td>
<td>0.90</td>
</tr>
<tr>
<td>7</td>
<td>2.85</td>
<td>2.85</td>
<td>1.95</td>
<td>1.35</td>
<td>0.90</td>
<td>0.90</td>
<td>0.00</td>
</tr>
</tbody>
</table>

The first attempt to solve the problem was not successful, and the problem became infeasible. This is because of customer demand increasing exponentially by time. To solve this problem, the company can replace one of the vehicles with a higher-capacity (40 units) rental vehicle. Therefore, the vehicle dispatching cost will increase due to the rental cost. By solving the problem with the new vehicle, a feasible solution was obtained. Figure 2.6 shows the result in which route 1 is 6-4-3-5-6 and route 2 is 6-2-1-6. The objective function value is -357.43 and computation time is 1,597 seconds. The solution is given below:

The objective value: -347.43

\[ X_{122} = X_{141} = X_{212} = X_{231} = X_{351} = X_{61} = X_{62} = 1 \]

All the other \( X_{e,g} \) variables are equal to zero

\[ z_{16} = z_{26} = 1 \]
Relaxation of increasing number of vehicles constraint: If customers’ demand rate is too high, regardless of the vehicle capacity, the problem with a small-size fleet can always become infeasible. If the vehicle capacity cannot be increased, then another way to address the issue would be to increase the number of available vehicles. In this case, the minimum number of vehicles required for a feasible solution can be obtained by the following iterative algorithm:

0- Start
1- \( K = \) number of available vehicle
2- Formulate the problem
3- Solve the problem
4- If the problem is feasible, then go to  6; otherwise, go to 5
5- \( K = K+1 \), go to 1
6- \( K \) is the number of vehicles required
7- End

To explain the above algorithm, the previous problem investigated for infeasibility is considered again. As already mentioned, the problem with two vehicles, each with a capacity of 30 units, was infeasible. By increasing the number of vehicles to three, the reformulated problem resulted in a feasible solution (Figure 2.7). The computation time is 4,004 seconds. Details of the solution are given below:
The objective value: -70.08045

\[X_{111} = X_{123} = X_{142} = X_{231} = X_{253} = X_{061} = X_{062} = X_{063} = 1\]

All the other \(X_{mgv}\) variables are equal to zero

\[z_{16} = z_{26} = z_{36} = 1\]

Figure 2.7. Network configuration solution for LRPTD with three 30-unit vehicles

This type of solution approach is useful when a trucking company is serving multiple supply chain systems and additional vehicles are available.

**Relaxation of requirement for serving all customers constraint:** When increasing the number of vehicles or their capacities is not possible, there is another alternative to avoid infeasibility of the problem. In this case, if serving all customers is not required by contract, then constraint (2.8) can be replaced by constraint (2.26). Here, the problem finds network configuration in a way in which only profitable customers will be served. It is recommended to solve the problem with and without all customer assignment conditions for comparison and making an economical decision.

\[
\sum_{i=1}^{n} \sum_{m \in D} X_{mgv} \leq 1 \quad \forall g \in I
\]  

(2.26)

In the case study represented in section 2.5.2 (hard time window), consider that the profit per unit of product sold is $0.2 (instead of $10), and let all other data remain the same. In this case, when assigning all customers is a requirement, the objective function will be $0.52, which means the system is in loss, and serving all customers is not profitable. Hence, the constraint that required all customers to be served is relaxed and the problem solved. Figure 2.8 shows the results of this relaxation case. The objective function value is $-1.89. Thus, with this
constraint relaxation, the system profits and the route that included customers 1 and 2 is eliminated. Thus, customers 1 and 2 will not be served.

Figure 2.8. Network configuration solution for VRP when $\Omega = \$0.2$ and assignment of all customers is relaxed

2.7 Linearization and Computational Results

For solving a mixed-integer linear programming (MILP) problem, several models have been developed in the literature, including cutting algorithms, Benders decomposition, Lagrangian relaxation, etc. However, the formulation presented in section 2.4.1 involves mixed-integer non-linear programming (MINLP), and it is necessary to transform it to MILP to be able to test other solution approaches. For the formulation presented in this chapter, depending on the type of travel-time function, different approaches can be taken for the linearization. To linearize the formulation, it is assumed that all customers’ demand functions are a linear function, and the generic form is presented in equation (2.27)

$$f_g (\tau_g) = d_g \pm \kappa_g \tau_g \quad (2.27)$$

To reduce the complexity of the model, it is also assumed that the objective function minimizes the total system cost, including transportation, depot establishment, and vehicle departure. With this definition, the objective function of the problem is reduced to term (2.28).

$$\sum_{v \in F} \sum_{m \in D} \sum_{h \in [I \cup J]} \sum \ C_{gh} \left( X_{mhn} X_{m-1gv} + X_{mhn} X_{0gv} P_{mv} \right) + \sum_{v \in F} \sum_{g \in J} z_{vg} B_g + \sum_{g \in J} q_g O_g \quad (2.28)$$

A general polynomial term can be presented by equation (2.29):
where $S \subseteq \{1, 2, ..., n\}$ and $x_j \in \{0, 1\}$, $j \in S$. Since all $x_j$ are binary variables and $y$ is a binary variable as well, the nonlinear term presented in constraint (2.29) can be replaced by two inequalities according to the following theorem:

**Theorem**: Let $s = |S|$, equation (2.29) holds if and only if

\[
\sum_{j \in S} x_j - y \leq s - 1,
\]

(2.30)

\[
-\sum_{j \in S} x_j + sy \leq 0,
\]

(2.31)

\[
x_j \in \{0, 1\}, j \in S, y \in \{0, 1\}
\]

(2.32)

**Proof**: If any $x_j$ is 0, then $y = 0$. In this case, constraint (2.30) is extra and redundant, and constraint (2.31) becomes $y \leq \sum_{j \in S} x_j / s < 1$, which implies $y = 0$ by conditions presented in equation (2.32). If all $x_j$s are equal to 1, then $y = 1$; in this case, $y \geq 1$ implies $y = 1$ by the condition presented in equation (2.32), and thus constraint (2.31) is redundant (Li and Sun, 2006).

By applying this theorem, the nonlinear binary programming presented in section 2.4.1 is linearized with the aforementioned assumptions in this section.

In the objective function, the first nonlinear term $X_{m-1, g, v}X_{m, b, v}$ is linearized with terms (2.33) and (2.34), defined as follows:

\[
Y_{mghv} = X_{m-1, g, v}X_{m, b, v} \quad \forall m \in D; h, g \in \{I \cup J\}
\]

(2.33)

\[
\begin{cases}
X_{m-1, g, v} + X_{m, b, v} - Y_{mghv} \leq 1 \\
-X_{m-1, g, v} - X_{m, b, v} + 2Y_{mghv} \leq 0 \quad \forall m \in D; h, g \in I \\
Y_{mghv} \in \{0, 1\}
\end{cases}
\]

(2.34)

The term $z_{sg}$ is already defined in equation (2.5) to simplify the objective function. By definition of $Y_{m-1, g, h, v}$ defined in 33 it can be modified as presented by equation (2.35).
\[ z_{vg} = \sum_{k \in d} Y_{kgv} \quad \forall g \in J; v \in V \] (2.35)

According to constraint (2.9), \( \sum_{g \in J} X_{mgv} \) shows whether or not there is a customer assigned to position \( m \) of the route. Hence, this summation works like a binary variable. Thus, for simplifying the later relaxation of the problem, this summation is also replaced by a binary variable in equation (2.36).

\[ \sum_{g \in J} X_{mgv} - y_{mv} = 0 \quad \forall m \in D; \forall v \in V \quad y_{mv} \in \{0, 1\} \] (2.36)

As presented in equation (2.4), \( P_{mv} \) has a nonlinear equation when \( m \in \{I / \{N\}\} \) and thus needs to be linearized. For this purpose, the \( P_{mv} \) complementary variable is defined in equation (2.37) and substituted in equation (2.4), which results in equation (2.38).

\[ \bar{P}_{mv} = 1 - P_{mv} \quad \forall m \in D; \forall v \in V \] (2.37)

\[ P'_{mv} = \prod_{n' = m+1}^{N} \bar{P}_{n'v} \sum_{g \in J} X_{mgv} \quad \forall m \in \{D / \{N\}\}, \forall v \in V \] (2.38)

In equation (2.38), the summation term can be substituted by the variable defined in equation (2.39).

\[ P'_{mv} = \prod_{n' = m+1}^{N} \bar{P}_{n'v} y_{mv} \quad \forall m \in \{D / \{N\}\}, \forall v \in V \] (2.39)

where \( P'_{mv} \in \{0, 1\}, s = N - m + 1 \), and

\[
\begin{align*}
\sum_{m' = m+1}^{N} \bar{P}_{m'v} + y_{mv} - P'_{mv} & \leq N - m \\
- \sum_{m' = m+1}^{N} \bar{P}_{m'v} - y_{mv} + (N - m + 1)P'_{mv} & \leq 0
\end{align*}
\]

(2.40)

The second term of the objective function can be linearized by one more step:

\[ \chi_{mghv} = X_{mhv} X_{0gv} P_{mv} \rightarrow s = 3 \quad \forall m \in D; h \in I; g \in J; v \in V \] (2.41)
\[
\begin{align*}
X_{mv} + X_{0v} + P_{mv} - \chi_{m0hv} & \leq 2 \\
-X_{mv} - X_{0v} - P_{mv} + 3\chi_{mhgv} & \leq 0 \quad \forall m \in D; h \in I; g \in J; v \in V \\
\chi_{mhgv} & \in \{0,1\}
\end{align*}
\]  

(2.42)

By replacing the nonlinear terms with the new binary variables defined, the objective function is linearized. However, the vehicle capacity constraint is still nonlinear and needs to be linearized as follows:

\[
\sum_{m} \sum_{g \in I} d_g X_{mgv} + \sum_{g \in I} \kappa_g X_{mgv} \sum_{m'} \sum_{h \in I} \sum_{g' \in [I \cup J]} (T_{gh} + S_g) Y_{m'ghv} \leq Y_v \quad \forall v \in V
\]

(2.43)

\[
\sum_{m} \sum_{g \in I} d_g X_{mgv} + \sum_{g \in I} \kappa_g (T_{gh} + S_g) X_{mgv} Y_{m'ghv} \leq Y_v \quad \forall v \in V
\]

(2.44)

\[
\nabla_{mgv} = X_{mgv} Y_{m'ghv} \forall m' \in \{m, m' \in D | m' \leq m\}; g' \in \{I \cup J\}; h, g \in I; v \in V
\]

(2.45)

Thus, the linear constraint of the vehicle capacity can be presented by equation (2.47).

\[
\sum_{m \in D} \sum_{g \in I} d_g X_{mgv} + \sum_{g \in I} \kappa_g (T_{gh} + S_g) Y_{m'ghv} \leq Y_v \quad \forall v \in V
\]

(2.47)

Finally, the linearized model is

\[
\text{Min} \\
\sum_{v \in V} \sum_{m \in D} \sum_{h \in I} \sum_{g \in [I \cup J]} C_{gh} (Y_{mghv} + \chi_{mghv}) + \sum_{v \in V} \sum_{g \in I} Y_{ghv} B_g + \sum_{g \in J} q_g O_g
\]

(2.48)

Subject to:

\[
\sum_{v \in V} \sum_{m \in D} X_{mgv} = 1 \quad \forall g \in I
\]

(2.49)

\[
v_{mv} \leq 1 \quad \forall m \in D; v \in V
\]

(2.50)

\[
\sum_{g \in [I \cup J]} X_{m-1gv} \geq \sum_{g \in [I \cup J]} X_{mgv} \quad \forall m \in D, v \in V
\]

(2.51)
\[ \sum_{g \in I} \sum_{h \in I} Y_{ghv} \leq 1 \quad \forall v \in V \quad (2.52) \]

\[
\begin{align*}
X_{m-1gv} + X_{mhv} - Y_{mghv} & \leq 1 \\
-X_{m-1gv} - X_{mhv} + 2Y_{mghv} & \leq 0
\end{align*} \quad \forall m \in D; h, g \in I; \forall v \in V \quad (2.53)\]

\[ \sum_{g \in I} X_{mgv} - y_{mv} = 0 \quad \forall m \in D, v \in V \quad (2.54) \]

\[
\begin{cases}
\sum_{m'=m+1}^{N} \bar{P}_{m',v} + y_{mv} - P_{mv} \leq N - m \\
- \sum_{m'=m+1}^{N} \bar{P}_{m',v} - y_{mv} + (N - m + 1)P_{mv} \leq 0
\end{cases} \quad \forall m \in \{D / \{N\}\}; g \in I; \forall v \in V \quad (2.55)\]

\[ y_{Nv} - P_{Nv} = 0 \quad m = N \quad \forall v \in V \quad (2.56) \]

\[
\begin{align*}
X_{mhv} + X_{0gv} + P_{mv} - X_{mhgv} & \leq 2 \\
-X_{mhv} - X_{0gv} - P_{mv} + 3X_{mhgv} & \leq 0
\end{align*} \quad \forall m \in D; h \in I; g \in J; v \in V \quad (2.57)\]

\[
\begin{cases}
X_{mgv} + Y_{mg'hv} - \nabla_{mg'h'}^{mgv} \leq 1 \\
-X_{mgv} - Y_{mg'hv} + 2\nabla_{mg'h'}^{mgv} \leq 0
\end{cases} \quad \forall \{m', m \in D | m' \leq m\}; g' \in \{I \cup J\}; h, g \in I; v \in V \quad (2.58)\]

\[ \sum_{m \in D} \sum_{g \in I} d_{g}X_{mgv} + \sum_{g \in I} \sum_{m \in D} \sum_{m'=1}^{m} \sum_{g \in \{I \cup J\}} \kappa_{g} (T_{gh} + S_{gh}) \nabla_{mg'h'}^{mgv} \leq Y_{v} \quad \forall v \in V \quad (2.59)\]

\[ Q_{g} \leq \sum_{v \in V} \sum_{h \in I} Y_{1ghv} \leq KO_{g} \quad \forall g \in J \quad (2.60)\]

\[ 1 \leq \sum_{v \in V} \sum_{h \in I} \sum_{g \in J} Y_{1ghv} \leq K \quad (2.61)\]

Also,

\[ \sum_{v \in V} \sum_{g \in J} \sum_{m \in D} X_{mgv} = 0 \quad (2.62) \]

\[ \sum_{v \in V} \sum_{h \in I} \sum_{g \in J} Y_{1ghv} = 0 \quad (2.63) \]

\[ \sum_{v \in V} \sum_{h \in I} \sum_{g \in [D \setminus N]} Y_{m+1ghv} = 0 \quad (2.64) \]
\[
\sum_{v \in V} \sum_{k \in I} \sum_{g \in I} \sum_{m \in \{D/N\}} Y_{m+1gkv} = 0 \quad (2.65)
\]
\[
\sum_{v \in V} y_v = 0 \quad (2.66)
\]
\[
y_{mv} \in \{0,1\}, P_{mv} \in \{0,1\}, Y_{mghv} \in \{0,1\}, Y_{mghv} \in \{0,1\}, X_{mghv} \in \{0,1\}, O_g \in \{0,1\}
\]

In addition, the hard time window constraint can be linearized using the definition of variable \( Y_{mghv} \) presented in equations (2.45) and (2.46) as

\[
A_{mv} = \sum_{g \in I} \sum_{m' = 1}^{m} \sum_{h \in I} \sum_{l \in \{J\}} \left( T_{gh} + S_g \right) X_{mghv} Y_{m'ghv} \quad \forall m \in D; v \in V \quad (2.67)
\]
\[
A_{mv} = \sum_{g \in I} \sum_{m' = 1}^{m} \sum_{h \in I} \sum_{l \in \{J\}} \left( T_{gh} + S_g \right) Y_{m'ghv} \quad \forall m \in D; v \in V \quad (2.68)
\]
\[
\sum_{g \in I} a_g X_{m'g} \leq \sum_{g \in I} \sum_{m' = 1}^{m} \sum_{h \in I} \sum_{l \in \{J\}} \left( T_{gh} + S_g \right) Y_{m'ghv} \leq \sum_{g \in I} b_g X_{m'g} \quad \forall m \in D; v \in V \quad (2.69)
\]

For linearizing soft time window variables, \( \Lambda_{m'g}^1 \) and \( \Lambda_{m'g}^2 \) are defined as

\[
\begin{align*}
\Lambda_{m'g}^1 &= \alpha_{mv} X_{m'g} \quad \forall v \in V; \forall m \in D \\
\alpha_{mv} + X_{m'g} - \Lambda_{m'g}^1 &\leq 1 \\
-\alpha_{mv} - X_{m'g} + 2 \Lambda_{m'g}^1 &\leq 0 \quad \forall v \in V; \forall m \in D \\
\Lambda_{m'g}^1 &\in \{0,1\}
\end{align*}
\]

\[
\begin{align*}
\Lambda_{m'g}^2 &= \beta_{mv} X_{m'g} \quad \forall v \in V; \forall m \in D \\
\beta_{mv} + X_{m'g} - \Lambda_{m'g}^2 &\leq 1 \\
-\beta_{mv} - X_{m'g} + 2 \Lambda_{m'g}^2 &\leq 0 \quad \forall v \in V; \forall m \in D \\
\Lambda_{m'g}^2 &\in \{0,1\}
\end{align*}
\]

Hence, the linear constraint is given by

\[
\sum_{g \in I} a_g X_{m'g} - \sum_{g \in I} \Delta a_g \Lambda_{m'g}^1 \leq \sum_{g \in I} \sum_{m' = 1}^{m} \sum_{h \in I} \sum_{l \in \{J\}} \left( T_{gh} + S_g \right) Y_{m'ghv} \leq \sum_{g \in I} b_g X_{m'g} + \sum_{g \in I} \Delta b_g \Lambda_{m'g}^2 \forall m \in D; v \in V 
\]
The linearized model for the example presented in section 2.5.1 is solved using the CPLEX Interactive Optimizer. Tables 2.10 to 2.12 show the objective function values, cutting strategies, number of cuts, and computation times for each solution approach. In the first row of these tables, the value of -1 shows that no cutting strategy is used, 0 indicates the automatic setting of CPLEX for applying cuts, 1 shows where the cuts are used moderately, and 2 indicates the aggressive usage of cuts in generating the results. As reflected in these tables, a moderate application of cutting methods has the smallest computation time among the solution strategies. Therefore, linearization of the model is essential to take advantage of cutting methods and reducing computation time. This is true only when the problem size is not large and the demands are linear. The evaluation of solution approaches for different types of demand functions and problem size is considered future research.

**TABLE 2.10 RESULTS OF SIX-NODE EXAMPLE #1**
**WHEN OBJECTIVE FUNCTION VALUE = $3.4957 \times 10^2**

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Cuts</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>26</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>0.33</td>
<td>0.30</td>
<td>0.30</td>
<td>0.3</td>
</tr>
</tbody>
</table>

**TABLE 2.11 RESULTS OF SEVEN-NODE EXAMPLE #2**
**WHEN OBJECTIVE FUNCTION VALUE = $3.4442 \times 10^2**

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUB Cover Cuts</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Clique Cuts</td>
<td>0</td>
<td>77</td>
<td>54</td>
<td>64</td>
</tr>
<tr>
<td>Cover Cuts</td>
<td>0</td>
<td>156</td>
<td>34</td>
<td>168</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>144</td>
<td>61</td>
<td>0</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>147</td>
<td>137</td>
<td>175</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>2.22</td>
<td>1.98</td>
<td>1.73</td>
<td>1.97</td>
</tr>
</tbody>
</table>

**TABLE 2.12 RESULTS OF EIGHT-NODE EXAMPLE #3**
**WHEN OBJECTIVE FUNCTION VALUE = $3.5055 \times 10^2**

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clique Cuts</td>
<td>0</td>
<td>14</td>
<td>58</td>
<td>21</td>
</tr>
<tr>
<td>Cover Cuts</td>
<td>0</td>
<td>1725</td>
<td>364</td>
<td>1712</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>561</td>
<td>1305</td>
<td>538</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>4</td>
<td>1005</td>
<td>2</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>1465</td>
<td>1</td>
<td>4132</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>107.87</td>
<td>259.37</td>
<td>173.72</td>
<td>271.29</td>
</tr>
</tbody>
</table>
2.8 Conclusions

In this chapter, a novel approach for a location-routing problem and its associated mathematical formulation is presented. In the problem shown, customers’ demands are time-dependent which means that they are dynamic and start changing after initiation according to a predefined function. Demand functions can be extracted from the historical data. In the existing arc formulation of the problem, the customer demand at the arrival time either exists or is zero. This problem is called the VRPTW and is vastly investigated in literature. Although the VRPTW has been thoroughly investigated, it only considers one type of demand function, which is static and does not change by time. Therefore, in this chapter, the problem is investigated with dynamicity in the demand function. The formulation provided is a node-based formulation and is capable of solving the LRP problem when demand is any arbitrary function of time. Since the proposed MINLP is a generic form of the VRPTW, it is also capable of solving the VRPTW. The validity of the model is demonstrated by solving the VRP with both hard time windows and soft time windows. When formulating problems with dynamic demand, infeasibility may occur. These infeasibilities can be solved by relaxing some of the constraints. A set of case studies that allowed constraints to be relaxed when faced with infeasibility has also been used to illustrate the newly developed formulation.

Different methods can be applied for solving the MINLP presented in section 2.4.1 using CPLEX, depending on the type of demand function. By assuming the linear functions for demand, the MINLP can be linearized for obtaining the solutions quickly using CPLEX. For solving the mixed-integer linear programming problem obtained, different solution strategies are applied including branch and bound, moderate and aggressive cut generators, etc. Results show that when customers’ demands are linear functions, the moderate cutting method can provide results faster than other methods. However, as already discussed, depending on the type of demand function, different solution methods can yield different solution times, and this is a subject of future research. In addition, since the problem is NP-hard, heuristic or meta-heuristic algorithms for solving the large problems are required. A closer look at constraints (2.11) and (2.12) reveals that they can be decoupled from the route/vehicle, and hence column generation or Benders decomposition for finding the exact solution of larger-size problems is possible and can be further investigated. The linearized model provided in this chapter can be the starting point for applying the aforementioned methods.

2.9 References


CHAPTER 3
LOCATION-ROUTING PROBLEM WITH TIME-DEPENDENT TRAVEL TIMES

3.1 Abstract

Traffic congestion causes substantial variation in travel time during different hours of a day. This significantly influences traveling decisions. In the context of a supply chain, traveling decisions with varying travel times are triggered in a time-dependent location-routing problem (TDLRP). In this chapter, an exact formulation of the TDLRP is presented in which the time taken to travel between each pair of nodes is a function of time. The problem formulation eliminates waiting times at customer locations and also tackles the problem with different scenarios, such as no time windows, hard and soft time windows, and time-dependent demand. The integer non-linear model presented is linearized and solved using CPLEX. The branch and bound approach and other cutting approaches are used for solving the model. Results show that the pure branch and bound approach provides faster results than cutting approaches for small-size problems.

Keywords: Time-Dependent Location-Routing Problem, Location-Routing Problem, Vehicle-Routing Problem, and Vehicle-Routing Problem with Time Window

3.2 Introduction

The problem under investigation in this chapter is a location-routing problem (LRP). LRP is a combination of two problems: the location-allocation problem (LAP) and the vehicle-routing problem (VRP) (Christofides and Eilon, 1969). LAP is the problem of locating a set of potential facilities and allocating customers to the locations with an objective of cost minimization (Fisher and Jaikumar, 1981). On the other hand, the VRP involves finding a set of routes originating from a set of depots to serve a set of customers with known demands. Each customer must be visited only once, and all vehicles return to the depot from which they departed. Also, cumulative customer demands in a route should not exceed vehicle capacity (Arntzen et al., 1995). Since the location of a distribution center (DC) impacts the routing of vehicles, the LAP and VRP are investigated together in a more comprehensive problem called the LRP.

Four major problems must be confronted in a supply chain network: production, location-allocation, inventory, and transportation (routing). Of the four, location-allocation and routing are usually considered the core problems of supply chain logistics (Larson and Odoni, 1981). These are usually referred to as an LRP. The time-dependent LRP (TDLRP) is a variant of the LRP in which travel times between nodes in the network are not
constant and may change depending on the time at which travel occurs. Even though the LRP has been vastly investigated in the literature, research on the TDLRP is very scarce (Figliozzi, (2009), Ichoua et al., (2003)). Existing research work usually addresses only the VRP with time-dependent travel times and does not approach the location and routing problem simultaneously. Orda and Rom (1990) propose an algorithm for the shortest problem in which an arbitrary function for a link delay is allowed. The objective of their work is to find the shortest path and minimum delay under different waiting constraints. Ahn and Shin (1991) develop a heuristic for the VRP with time window constraints and time-varying congestion. The heuristic is a modification of the savings, insertion, and local improvement algorithms. Hill and Benton (1992) present a method for estimating the time-dependent travel speed and a heuristic to solve the time-dependent vehicle-routing problem (TDVRP). Malandraki and Daskin (1992) present a mixed-integer linear programming (MILP) model for the TDVRP with time window constraints. They develop a heuristic algorithm using the nearest neighborhood heuristic for the VRP without time windows. In addition, a mathematical heuristic for the TDVRP with time windows is also developed. In this work, waiting time is allowed at nodes. The step functions considered for the travel times are symmetric. Ichoua et al. (2003) present a heuristic solution methodology based on the tabu search algorithm. Figliozzi (2009) present a flow-arc formulation for the TDVRP with hard and soft time windows along with heuristic algorithms for solving the problem. Most recent efforts regarding the TDLRP have focused on developing a heuristic/meta-heuristic for solving the problem (Hashimoto et al., (2008); Donati et al., (2008); Zheng-yu et al., (2010)).

All efforts in the literature relative to the formulation of the TDVRP incorporate an assumption of time windows, and to the best knowledge of the author, there is no formulation for the TDVRP without this consideration. Time windows define a period of time during which the customer can be served. The time windows assumption is used to simplify the formulation of the problem and to calculate the arrival time at a node based on two conditions: (1) the sum of the arrival time at a customer and the travel time from the current customer to the next customer should be less than the latest arrival at the next customer; and (2) the service start time at each customer should be within a specified time window. These two conditions may sometimes lead to large and unrealistic waiting times at customer locations. For instance, consider a truck that is at a customer location at time 40. The time window for serving the next customer is [200, 260], and the travel time to the next customer is 20 minutes. Based on the traditional formulation, the arrival time at the next customer can be any time between 200 and 260, which results in a waiting time of at least 140 minutes for the truck before serving the next customer, which
is not practical, especially if the time unit is large, e.g., hours, days, etc. The main weakness of methods in the literature presented for the TDVRP can be categorized into two groups:

1. The travel time function is usually considered a discrete step function. For this reason, waiting times must be permitted at the customer location in order to obtain feasible solutions.

2. The effort toward applying a continuous travel time function is very rare, and the formulations presented are too complicated and intricate to be analytically solved. A continuous travel function will allow improved modeling of the travel times, especially when the time intervals are for a day or shorter. This also allows easier analysis of the travel function, when the time changes from highly congested to less congested, or vice-versa.

The objectives of this chapter are to alleviate these two shortcomings found in the current literature and enhance the TDLRP formulation. Here, the goals are to develop a formulation for the TDLRP when travel times are a function of time (discrete or continuous) and no waiting time is allowed at a customer location. The travel-time function can be developed from historical data of traffic congestion. A TDLRP in which each customer’s demand must be satisfied within a time interval is called a location-routing problem with time-dependent travel times and with time windows (TDLRPTW). In the literature, whenever the TDLRP is investigated, it is in fact a TDLRPTW; it seems that the TDLRP and time windows are not separable. In the TDLRP literature, there is no formulation of the problem in which there is no time window. Formulation of the TDLRP without a time window is necessary for eliminating the waiting times at customers’ location, as described earlier. In this research, TDLRP formulations with hard and soft time windows are also developed. The formulation is further extended to include time-dependent demand. Applying the formulations proposed in this chapter can hopefully determine the best strategies regarding location of DCs, allocation of customers to DCs, and routing plan from DCs to customers.

Section 3.3 provides a detailed definition of the problem under investigation. Section 3.4 is devoted to the development of the mathematical formulation for the TDLRP with different scenarios. In section 3.5, an illustrative example is solved for each of the mathematical models presented in section 3.4. The linearization of the MINLP and the solution approach are investigated in section 3.6. Finally, section 3.7 provides conclusions and suggestions for future research directions.
3.3 Problem Statement

This section presents notations used for the formulation of the problem. There are \( N \) customers and \( M \) depots in the problem. The collective set of DCs and customers in the network are represented by nodes. Nodes 1 to \( N \) represent customers, and nodes \( N + 1 \) to \( N + M \) represent DCs. The decision variables in the formulation provide the assignment of customers to vehicles, vehicles to DCs, and the sequence of visits by each vehicle. When nodes are assigned to a vehicle, a route is formed; thus, a route is formed by a set of nodes. The position of a node in the route is the order in which the node is visited by the vehicle. For instance, node \( g \) in position 2 of vehicle 1’s route implies that node \( g \) is the second node visited by vehicle 1. \( D \) is the set of possible positions that a customer can take in a route. A vehicle may be assigned to visit, at the most, all \( N \) customers. Thus, the maximum number of possible positions for a customer is equal to \( N \). The decision variables, \( X_{mgv} = 1 \), imply that node \( g \) (a customer or a DC) is the \( m^{th} \) node visited by vehicle \( v \). In other words, node \( g \) is the \( m^{th} \) node in the route assigned to vehicle \( v \). Thus, in the definitions of notations, vehicle and route are used alternatively since they represent the same concept:

- \( N \) Total number of customers
- \( M \) Total number of DCs
- \( K \) Total number of vehicles
- \( I \) Set of customers, \( I = \{1, 2, \ldots, N\} \)
- \( J \) Set of DCs, \( J = \{N+1, 2, \ldots, N+M\} \)
- \( D \) Set of possible positions that a customer can take in a route, \( D = \{1, 2, \ldots, N\} \)
- \( V \) Set of vehicles, \( V = \{1, 2, \ldots, K\} \)
- \( Y_v \) Capacity of vehicle \( v \)
- \( E_g \) Customer \( g \) fulfillment level, \( \forall g \in I \)
- \( g/h \) Index used for all nodes

\[
X_{mgv} = \begin{cases} 
1 & \text{if node } g \text{ is in position } m \text{ of the route } v; \\
0 & \text{otherwise}
\end{cases}, \quad \forall g \in \{I \cup J\}; \forall m \in D; \forall v \in V
\]

\[
P_{mv} = \begin{cases} 
1 & \text{if } m \text{ is the last taken position of route } v; \\
0 & \text{otherwise}
\end{cases}, \quad \forall m \in D; \forall v \in V
\]

\[
O_g = \begin{cases} 
1 & \text{if there is any vehicle assigned to node } g; \\
0 & \text{otherwise}
\end{cases}, \quad \forall g \in J
\]
\[ z_{vh} \begin{cases} 1 \text{ if vehicle } v \text{ is assigned to node } h; \\ 0 \text{ otherwise} \end{cases} \quad \forall h \in J; \forall v \in V \]

**Cost per unit of time (labor cost, vehicle cost, etc.)**

**Fixed cost for establishing node** \( g \), \( \forall g \in J \)

**Initial demand of node** \( g \), \( \forall g \in I \)

**Demand function of node** \( g \) **at time** \( t \), \( \forall g \in I \)

**Service time at node** \( g \), \( \forall g \in I \)

**Arrival time at position** \( m \) **of route** \( v \), \( \forall m \in D; \forall v \in V \)

**Cumulative departure time from position** \( m \) **on route** \( v \), \( \forall m \in D; \forall v \in V \)

**Departure time from position** \( m \) **on route** \( v \) **(between 0:00 and 24:00)**, \( \forall m \in D; \forall v \in V \)

**Travel time function between nodes** \( g \) **and** \( h \), \( \forall g, h \in \{I \cup J\} \)

**Departure cost from distribution center** \( g \), \( \forall g \in J \)

\[ a_{mv} \begin{cases} 1 \text{ if earliest arrival at position } m \text{ of route } v \text{ is violated; } \\ 0 \text{ otherwise} \end{cases} \quad \forall m \in D; \forall v \in V \]

\[ \beta_{mv} \begin{cases} 1 \text{ if latest arrival at position } m \text{ of route } v \text{ is violated; } \\ 0 \text{ otherwise} \end{cases} \quad \forall m \in D; \forall v \in V \]

**Earliest arrival time at customer** \( g \), \( \forall g \in I \)

**Latest arrival time at customer** \( g \), \( \forall g \in I \)

**Maximum deviation permitted from the earliest arrival time at customer** \( g \), \( \forall g \in I \)

**Maximum deviation permitted from the latest arrival time at customer** \( g \), \( \forall g \in I \)

\[ w^1_g \begin{cases} 1 \text{ if demand has decreased at the delivery time to customer } g; \\ 0 \text{ otherwise, } \forall g \in I \end{cases} \]

\[ w^2_g \begin{cases} 1 \text{ if demand has increased at the delivery time to customer } g; \\ 0 \text{ otherwise, } \forall g \in I \end{cases} \]

**Lost-order cost**

**Percentage change in unit price of extra product delivered**

**Profit obtained from selling a unit of product**

**Penalty cost associated with violation of lower time limit at customer** \( g \), \( \forall g \in I \)
\( \phi_g \) penalty cost associated with violation of upper time limit at customer \( g \), \( \forall g \in I \)

To ensure that vehicles always depart from a DC, the position zero of each route is reserved for a DC. Hence, only DCs can be assigned to the position 0, and DCs cannot take any other position in a route.

Each customer \( g \in I \) has a demand, \( d_g \), which is less than the vehicle capacity, \( Y \). The travel time between each pair of nodes in the system is a function of time, \( F_{gh}(<t>) \), which is derived from historical data. There are \( K \) heterogeneous vehicles available, and each vehicle departs fully loaded from the DCs. In addition, \( M \) locations are available for the establishment of DCs. If a DC is established, it incurs a cost of \( q_g, g \in J \) in the system. There is only one type of product in the system. The objective of this problem is to find locations for establishing DCs and routing plans for vehicles in order to minimize the system cost.

Decision variables, \( X_{mgv} \), are defined to model the problem. The variable \( X_{mgv} \) is a binary variable for \( \forall g \in \{I \cup J\}, \forall m \in D, \forall v \in V \), which takes a value of 1, if node \( g \) (a customer or a DC) is placed in order (position) \( m \) of vehicle \( v \)'s routing plan; otherwise, it is 0. There are additional terms defined to simplify the representation of the objective function and the constraints in the problem formulation. The term \( P_{mv} \), presented in equation (3.1), is used to determine the final position taken on route \( v \).

\[
P_{mv} = \begin{cases} 
\sum_{g \in I} X_{mgv} & m = N \\
\prod_{m = m+1}^{N} (1 - P_{mv}) \sum_{g \in I} X_{mgv} & \forall m \in \{D \setminus \{N\}\}
\end{cases} \quad v \in V \tag{3.1}
\]

Position 0 of each route is reserved for a DC. If the binary variable \( X_{0gv} = 1, \forall g \in J, \forall v \in V \), then this means that vehicle \( v \) is assigned to distribution center \( g \). However, DC \( g \) is not opened until a customer is also assigned to the vehicle, \( v \). Thus, \( g \) will be vehicle \( v \)'s DC only if there is a link between the depot and a customer in the network. The variable \( z_{vh} \), defined in equation (3.2), ensures that a connection exists between a DC and a customer in the system. The variable \( z_{vh} \) is similar to the connectivity constraint between the LAP and the VRP in the traditional formulation of the LRP and connects the location decision to routing decisions.

\[
z_{vh} = \sum_{g \in I} X_{0vh} X_{1gv} \quad \forall h \in J; v \in V \tag{3.2}
\]
The value of $t_{mv}$ is calculated through a set of recursive equations given by equation (3.3). The value of $t_{mv}$ can be considered to be 0 if the start time is set to zero.

$$
\begin{align*}
  t_{ov} &= 0 \\
  t_{mv} &= \left( \sum_{m=1}^{m} \sum_{he \in I} \sum_{g \in [I,J]} \left( F_{gh} \left( t_{m-1}v \right) + S_{h} \right) X_{m'g'v} X_{m'hv} \right) \left( \sum_{g \in [I,J]} X_{m'g'v} \right) \quad \forall m \in D; \forall v \in V
\end{align*}
$$

(3.3)

If the travel time function repeats itself after $H$ units of time, then equation (3.3) must be replaced by equation (3.4), in order to calculate the $t_{mv}$, $\forall m \in D; \forall v \in V$.

$$
\begin{align*}
  T_{ov} &= t_{ov} = 0 \\
  t_{mv} &= \left( \sum_{m=1}^{m} \sum_{he \in I} \sum_{g \in [I,J]} \left( F_{gh} \left( T_{m-1}v \right) + S_{h} \right) X_{m'g'v} X_{m'hv} \right) \left( \sum_{g \in [I,J]} X_{m'g'v} \right) \quad \forall m \in D; \forall v \in V
\end{align*}
$$

(3.4)

$T_{mv} = \text{Mod}(t_{mv}, H)$

where the Mod function returns the remainder of dividing $t_{mv}$ by $H$. For instance, if the travel-time functions are represented in hour unit and they repeat every day, then the value of $H$ will be 24. The following section presents the mathematical formulation of the problem.

### 3.4 Mathematical Formulation of Problem

As previously discussed, one of the main drawbacks of existing formulations of the TDVRP is the possibility of waiting times at a customer location. The formulation of the problem presented in this section overcomes this deficiency and eliminates waiting times at a customer’s location.

The TDLRP is first formulated without the time window condition in section 3.4.1. The formulation of the TDLRP with time windows is then addressed in sections 3.4.2 and 3.4.3. The formulations in sections 3.4.2 and 3.4.3 ensure that the waiting time at customers’ locations are eliminated. The time window can be hard or soft. In a formulation with a “hard time window,” each customer has an associated time window during which the demand must be met. The vehicle cannot deliver products to the customer before the start of the time window or after the time window has elapsed, i.e., late or early arrival at a customer is not acceptable (section 3.4.2). In a TDLRP with a “soft time window” (section 3.4.3), the customer can be served before and after the preferred time window, i.e., early or late arrival at a customer is acceptable up to a predefined limit. However, when there is a late or early service to the customer, there is a penalty cost associated with the violation of the time window. Section 3.4.4 presents the formulation when both demand and travel times are time dependent.
3.4.1 Formulation of TDLRP without Time Windows

Programming of the MINLP as defined in section 3.3 is presented below:

Min

\[
\sum_{v \in V, m \in D} C \cdot P_{mv} \left( t_{mv} + \sum_{h \in I, g \in J} F_{hg} \cdot T_{mv} \cdot X_{0gv} \cdot X_{mgv} \right) + \sum_{v \in V, g \in J} z_{vg} B_{g} + \sum_{g \in J} q_{g} O_{g}
\]  

(3.5)

Subject to:

\[
\sum_{v \in V, m \in D} X_{mgv} = 1 \quad \forall g \in I
\]  

(3.6)

\[
\sum_{g \in I} X_{mgv} \leq 1 \quad \forall m \in D; v \in V
\]  

(3.7)

\[
\sum_{g \in I} \sum_{m \in D} d_{g} \cdot X_{mgv} \leq Y_{v} \quad \forall v \in V
\]  

(3.8)

\[
\sum_{v \in V, g \in J, m \in D} X_{mgv} = 0
\]  

(3.9)

\[
\sum_{v \in V, g \in J} X_{0gv} = 0
\]  

(3.10)

\[
\sum_{g \in I} X_{m-1gv} \geq \sum_{g \in I} X_{mgv} \quad \forall m \in D; v \in V
\]  

(3.11)

\[
\sum_{g \in J} z_{vg} \leq 1 \quad \forall v \in V
\]  

(3.12)

\[
O_{g} \leq \sum_{v \in V} z_{vg} \leq KO_{g} \quad \forall g \in J
\]  

(3.13)

\[
1 \leq \sum_{v \in V, g \in J} z_{vg} \leq K
\]  

(3.14)

\[
X_{mgv}, z_{vg}, P_{mv}, O_{g} \in \{0, 1\}
\]

Equation (3.5) is the objective function, which minimizes total travel time, cost of establishing DCs, and cost of dispatching vehicles from DCs. Constraint (3.6) ensures that each customer appears in only one route, i.e., only one route is assigned to each customer. Constraint (3.7) enforces that each position of a route is not taken by more than one customer. Constraint (3.8) ensures that the total demand of customers assigned to a route is less than
the vehicle capacity. It is assumed that position zero of each route is reserved for DCs. This assumption implies that DCs cannot take any other position in the routes and also that customers cannot take position 0 of their assigned route. These conditions are imposed by equations (3.9) and (3.10). Constraint (3.11) ensures that position \( m+1 \) of a route cannot be taken unless position \( m \) is taken. Constraint (3.12) guarantees that a vehicle is not assigned to more than one DC. Constraint (3.13) determines whether a DC is open or closed. Constraint (3.14) keeps the total number of vehicles between one and the number of available vehicles.

### 3.4.2 Formulation of TDLRP with Hard Time Window

In this problem, it is assumed that waiting at a customer location is not allowed. The difference between the arrival and departure times at a customer is considered the service time. Therefore, the arrival time at a customer can be determined by equation (3.15).

\[
A_{mv} = \left( t_{mv} - \sum_{m=1}^{\infty} \sum_{h \in I} \sum_{g \in J} S_h \cdot X_{m-1gv} \cdot X_{mhv} \right) \left( \sum_{g \in I} X_{mgv} \right) \quad \forall m \in D; \forall v \in V
\]  

(3.15)

where \( A_{mv} \) calculates the arrival time at position \( m \) of route \( v \). When the service time is zero, equation (3.4) is reduced to equation (3.16).

\[
A_{mv} = t_{mv} \quad \forall m \in D; \forall v \in V
\]

(3.16)

Therefore, by adding the constraint presented in equation (3.17) to the set of constraints (3.6) to (3.14), the model can handle the TDLRP with hard time windows.

\[
\sum_{g \in I} a_g X_{mgv} \leq A_{mv} \leq \sum_{g \in I} b_g X_{mgv} \quad \forall m \in D; v \in V
\]

(3.17)

### 3.4.3 Formulation of TDLRP with Soft Time Window

As opposed to hard time windows, soft time windows allow for a time-interval violation but with an assigned penalty cost. Therefore, the objective function proposed in equation (3.18) has a term related to the penalty cost associated with the time window violation.

\[
\text{Min} \sum_{v \in V} \sum_{m \in D} C \cdot P_{mv} \left( t_{mv} + \sum_{h \in I} \sum_{g \in J} \left( F_{hg} \cdot t_{mv} \cdot X_{0gv} \cdot X_{mhv} \right) \right) + \sum_{v \in V} \sum_{g \in I} z_{vg} B_g + \sum_{g \in I} q_g O_g
\]

(3.18)
The objective function minimizes total travel time, DC-establishment cost, vehicle-dispatching cost, and penalty cost associated with the time window violation.

The time window constraint also must be modified to consider the permitted flexibility. The modified constraint is presented in equation (3.19).

\[
\sum_{g \in I} a_g X_{mgv} - \left( \sum_{g \in I} \Delta a_g X_{mgv} \right) \alpha_{mv} \leq A_{mv} \leq \sum_{g \in I} b_g X_{mgv} + \left( \sum_{g \in I} \Delta b_g X_{mgv} \right) \beta_{mv} \quad \forall m \in D; v \in V, \alpha_{mv}, \beta_{mv} \in \{0,1\}
\]

(3.19)

Constraint 19 is used to ensure that the arrival time at each position of a route is within the related time interval with associated allowed deviation.

Thus, by replacing the objective function presented in equation (3.5) with the term defined in equation (3.18), and adding constraint (3.19) to the set of constraints (3.6) to (3.14), the model is modified to be applied to the TDLRP with soft time windows.

3.4.4 Formulation of TDLRP with Dynamic Travel Time and Demand

In a conventional LRP with a time window (LRPTW), each customer has an initial demand called \( D_g \). The demand may either exist or expire, depending on the arrival time at the customer. This statement is shown by the function presented in equation (3.20) (Mirzaei and Krishnan, 2011):

\[
f(\tau_g) = \begin{cases} 
0 & \text{if } \tau_g \leq a_g \\
d_g & \text{if } a_g \leq \tau_g \leq b_g \\
0 & \text{if } \tau_g \geq b_g 
\end{cases} \quad \forall g \in I
\]

(3.20)

where \( a_g \) is the earliest arrival time at node \( g \), and \( b_g \) is the latest arrival time at node \( g \). Also, based on the basic assumptions of vehicle-routing problems, \( d_g \leq VC, \forall g \in I \).

However, in an LRP with time-dependent demand (LRPTD), the initial demand is expected to change from the time of initiation, i.e., \( f(\tau_g), \forall g \in I \). In fact, the conventional LRPTW is a special case of the time-dependent demand problem in which the demand function is defined by equation (3.20). This section is devoted to the development of a formulation for the TDLRP in which demands and travel time changes are defined by parametric time-dependent functions. The arrival time at customer \( g \) can be calculated as

\[
\tau_g = \sum_{v \in F} \sum_{m \in D} X_{mgv} A_{mv}, \forall g \in I
\]

(3.21)
The objective function of the TDLRP with time-dependent demand minimizes the total cost of the system while maximizing profit. When a customer demand changes with time, the customer demand at delivery time is not the same as the initial demand. Therefore, for customers with a decreasing demand function, there is a “lost-order cost” incurred in the system. This cost is the result of not meeting the demand completely or partially, i.e., \((d_g - f_g(t)) \cdot \lambda\), where \(\lambda\) is the lost-order cost per unit of product. The profit is the product of the total quantity delivered to the customer, the profit per unit, and the customer fulfillment level, i.e., \(\text{profit} = f_g(t) \cdot E_g \cdot \Omega\) in which \(E_g\) is the customer fulfillment level defined by equation (3.22).

\[
E_g = \frac{f_g(t_g)}{d_g} \quad \forall g \in I
\]  

(3.22)

The value of \(E_g\) is dynamic and depends on the time of delivery. The value of \(E_g\) for customers with a monotonously increasing demand function is greater than one, and the value of \(E_g\) for customers with a monotonously decreasing demand function is less than one. The unit product price for the additional number of products delivered to customers with increasing demand can be different from the initial price. For example, the price can be cheaper due to quantity discount. The variable \(\gamma\) is the constant representing the percentage of decrease or increase in the price.

The objective function for the TDLRP with time-dependent demand is given by equation (3.23):

\[
\text{Min}
\sum_{v \in V} \sum_{m \in D} C \cdot P_{mv} \left( t_{mv} + \sum_{h \in H} \sum_{g \in J} \left( F_{hg}(T_{mv}) \cdot X_{0gvm} \cdot X_{mhm} \right) \right) + \sum_{v \in V} \sum_{g \in J} z_{vg} B_g + \sum_{g \in J} q_g O_g
+ \sum_{g \in J} \left( w_g^3 \cdot \lambda \cdot (d_g - f_g(t_g)) \pm w_g^2 \cdot \Omega \cdot \gamma \cdot (f_g(t_g) - d_g) - \Omega E_g \cdot f_g(t_g) \right)
\]  

(3.23)

where the first term determines the total transportation cost, which is the product of the total travel time and unit time cost. The second term is the cost of vehicle dispatching from DCs. The third term includes the fixed DC-establishment cost. The last term includes the lost-order cost, additional order cost/profit, and sales profit, respectively. Two modifications are required for the set of constraints presented in equations (3.6) to (3.14) to handle the TDLRP with time-dependent demand, as opposed to the formulation in section 3.4.1. First, it is necessary to change the vehicle capacity constraint (3.8) to constraint (3.24), in order to consider the demand variability. Second, the set of constraints presented in equation (3.25) should be added to the set of constraints (3.6) to (3.14).
\sum_{g=1}^{G} \sum_{m=0}^{D} f_g(r_g) \cdot X_{m,g} \leq Y_v \quad \forall v \in V \tag{3.24}

w^1_g (d_g - f_g(r_g)) \geq 0 \quad \forall g \in I

w^2_g (d_g - f_g(r_g)) < 0 \quad \forall g \in I

w^1_g + w^2_g = 1 \quad \forall g \in I \tag{3.25}

This set of constraints is used to determine whether a customer's demand at the time of delivery is higher or lower compared to the initial demand.

Since the TDLRP is the generic formulation of the TDLRPTW, it can handle both soft and hard time windows. Interested readers can refer to the work of Mirzaei and Krishnan (2011) for more information.

3.5 Illustrative Examples

3.5.1 Example of TDLRP without Time Window Constraints

A two-layer network problem is used to illustrate the proposed mathematical model. The problem consists of two DCs and four customers. Nodes 1, 2, 3, and 4 represent customers, and nodes 5 and 6 represent potential DCs. Figure 3.1 shows the travel time functions between customers (nodes 1, 2, 3, and 4) and DCs (nodes 5 and 6).

Figure 3.1. Travel time functions between customers and DCs

Figure 3.2 shows the travel time functions between each pair of customers. Although the model can handle such asymmetric functions, in this example, it is assumed that they are symmetric and are repeated every 24 hours. For testing the model, different types of functions are used in this example. The appendix shows a list of travel time functions used in this case study.
Two vehicles with capacities of 50 and 70 units of product are available. The departure costs from node 5 and node 6 are $45 and $50, respectively. In the planning horizon of the problem, the fixed cost of each DC is $250 for node 5 and $200 for node 6. Customers’ demands for customers 1, 2, 3, and 4 are 40, 25, 20, and 10, respectively. Service times at customer locations are all zero. The unit time (hour) cost for service is $3.

The mathematical formulation was solved using LINGO 12.0 optimizer software on a Pentium D CPU 3.2 GHz, with 3.25 GB of RAM. Results, shown in Table 3.1, imply that there are two routes in the network. The first route of 6-3-2-6 is assigned to vehicle 1, which has a capacity of 50 units, and the second route of 6-1-4-6 is assigned to vehicle 2, which has a capacity of 70 units. Both routes are assigned to depot 2 (node 6).

**TABLE 3.1 RESULTS OF TDLRP WITHOUT TIME WINDOW CONSTRAINTS**

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$619.06</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:07:36</td>
</tr>
<tr>
<td>Variables with Value of 1</td>
<td>$X_{12}, X_{13}, X_{23}, X_{25}, X_{65}, X_{66}, z_{16}, z_{26}, P_{21}, P_{22}$</td>
</tr>
</tbody>
</table>
3.5.2 Example of TDLRP with Hard Time Window Constraints

This example is similar to the one presented in section 3.5.1, only each customer is assigned a time window (Table 3.2). Each customer’s demand is equal to its initial demand, \( D_g \), if it is served within the specified time window; otherwise, it is zero.

<table>
<thead>
<tr>
<th>Customer Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Interval (Hours)</td>
<td>[20,27]</td>
<td>[14,16]</td>
<td>[20,25]</td>
<td>[15,20]</td>
</tr>
</tbody>
</table>

The mathematical formulation presented in equation (3.2) was solved using LINGO 12.0 optimizer software on a Pentium D CPU 3.2 GHz, with 3.25 GB of RAM. Results, shown in Table 3.3, imply that two routes are in the network. The first route of 6-2-4-6 is assigned to vehicle 1 with a capacity of 50 units, and the second route of 6-3-1-6 is assigned to vehicle 2 with a capacity of 70 units. Both routes are assigned to depot 2 (node 6).

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$837.53</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:00:40</td>
</tr>
<tr>
<td>Variables with Value of 1</td>
<td>( X_{121}, X_{132}, X_{221}, X_{241}, X_{601}, X_{602}, O_6, z_{16}, z_{26}, P_{21}, P_{22} )</td>
</tr>
</tbody>
</table>

3.5.3 TDLRP with Soft Time Window Constraints

The example in this section is similar to the example in section 3.5.2, except that violations from the time interval are allowed up to a specified limit. The permitted violations and penalty costs associated with them are provided in Table 3.4. Since the problem investigated in section 3.5.2 has an optimal solution with a hard time window constraint, solving the same problem with a soft time window will result in no difference in terms of the final answer. Hence, to illustrate the impact of soft time-window constraints and the formulation on the solution, the time windows are tightened in this example.

<table>
<thead>
<tr>
<th>Customer Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Interval (Hours)</td>
<td>[26,26]</td>
<td>[16,17]</td>
<td>[20,21]</td>
<td>[15,18]</td>
</tr>
<tr>
<td>Upper Limit Violation Allowed (Hours)</td>
<td>1</td>
<td>0.5</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>Lower Limit Violation Allowed (Hours)</td>
<td>0.5</td>
<td>0.75</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>Penalty Cost ( p_g ) ($ )</td>
<td>20</td>
<td>10</td>
<td>50</td>
<td>30</td>
</tr>
<tr>
<td>Penalty Cost ( \phi_g ) ($ )</td>
<td>50</td>
<td>70</td>
<td>20</td>
<td>10</td>
</tr>
</tbody>
</table>
The mathematical formulation presented in section 3.4.3 was solved using LINGO 12.0 optimizer software on a Pentium D CPU 3.2 GHz, with 3.25 GB of RAM. Results are shown in Table 3.5. Although, the route configuration from Table 3.5 is similar to the one obtained in section 3.5.2, the objective function value is larger, as a result of tightening the time windows and allowing the model to violate the time intervals by accepting the associated penalty costs. As shown in Table 3.5, $\alpha_{11}$, $\alpha_{22}$, $\beta_{12}$, and $\beta_{21}$ have values of one, which implies that the lower limit of time windows for customers 1 and 2, and upper limit of time windows for customers 1 and 4 are violated. The consequence of these violations is $60$, which is added to the objective value obtained in Table 3.5. The objective function has a value of 897.53 for the TDLRP with a soft time window.

### TABLE 3.5 RESULTS OF TDLRP WITH SOFT TIME WINDOW

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$897.53</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:01:02</td>
</tr>
<tr>
<td>Variables with Value of 1</td>
<td>$X_{121}, X_{132}, X_{212}, X_{241}, X_{061}, X_{062}, O_6, z_{16}, z_{26}, P_{21}, P_{22}, \alpha_{11}, \alpha_{22}, \beta_{12}, \beta_{21}$</td>
</tr>
</tbody>
</table>

### 3.5.4 TDLRP with Time-Dependent Demand

The example in this section is similar to the one discussed in section 3.5.1, except that the customers’ demands are not static. Each customer has a unique demand function, as presented in Table 3.6.

### TABLE 3.6 CUSTOMERS’ DEMAND INFORMATION FOR TDLRP WITH TIME-DEPENDENT DEMAND

<table>
<thead>
<tr>
<th>Customer</th>
<th>Initial Demand</th>
<th>Demand Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td>$40 + 0.2\tau_1$</td>
</tr>
<tr>
<td>2</td>
<td>25</td>
<td>$25 - 0.2\tau_1$</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>$20 + 0.3\tau_1$</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>$10 - 0.5\tau_1$</td>
</tr>
</tbody>
</table>

The mathematical formulation presented in section 3.4.4 was solved using LINGO 12.0 optimizer software on a Pentium D CPU 3.2 GHz, with 3.25 GB of RAM. Results, shown in Table 3.7, imply that there are two routes in the network. The first route of 6-1-4-6 is assigned to vehicle 1 with a capacity of 50 units, and the second route of 6-3-2-6 is assigned to vehicle 2 with a capacity of 70 units. Both routes are assigned to depot 2 (node 6). The negative value of the objective function indicates a network profit of $386.11.

### TABLE 3.7 RESULTS OF TDLRP WITH TIME-DEPENDENT DEMAND

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-386.11$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:10:38</td>
</tr>
<tr>
<td>Variables with Value of 1</td>
<td>$X_{111}, X_{132}, X_{232}, X_{241}, X_{061}, X_{062}, O_6, z_{16}, z_{26}, P_{21}, P_{22}, w_2, w_4$</td>
</tr>
</tbody>
</table>
LINGO optimization software using the branch-and-bound method was used for solving MINLP presented in this section. Other commercial packages, such as CPLEX, give researchers more options for choosing their problem-solving method in a faster amount of time; however, CPLEX has limitations and cannot solve MINLP directly. In order to be solved by CPLEX, MINLP should first be modified to a MILP or a quadratic formulation. Hence, in the next section of this chapter, the model presented in section 3.4.1 is first linearized, and then different cutting approaches are tested to find the best solution strategy for solving the problem.

3.6 Linearization and Solution Approach

For solving MILP, several models have been developed in the literature, such as cutting algorithms, Benders decomposition, Lagrangian relaxation, etc. However, the formulation presented in section 3.4.1 involves MINLP, and it is necessary to transform it to MILP to be able to test other solution approaches. A closer look at the model reveals that it is, in fact, constrained nonlinear binary programming, and depending on the type of travel time function, different approaches can be taken for the linearization. Since, the most commonly used types of function in the literature are discrete step functions, it is assumed that the travel functions are discrete step functions and the model is linearized accordingly.

A general polynomial term can be presented as

\[ y = \prod_{j \in S} x_j \]  

(3.26)

where \( S \subseteq \{1, 2, \ldots, n\} \) and \( x_j \in \{0, 1\}, j \in S \). Since all \( x_j \) are binary variables and \( y \) is a binary variable as well, the nonlinear term presented in equation (3.26) can be replaced by two inequalities according to the following theorem:

**Theorem:** Let \( s = |S| \), Equation \( y = \prod_{j \in S} x_j \quad x_j \in \{0, 1\} \)

where \( S \subseteq \{1, 2, \ldots, n\} \) and \( x_j \in \{0, 1\}, j \in S \) holds if and only if

\[ \sum_{j \in S} x_j - y \leq s - 1, \]  

(3.27)

\[-\sum_{j \in S} x_j + sy \leq 0, \]  

(3.28)

\[ x_j \in \{0, 1\}, j \in S, y \in \{0, 1\} \]  

(3.29)
**Proof:** If any \( x_j \) is 0, then \( y = 0 \). In this case, constraint (3.28) is extra and redundant, and constraint (3.27) becomes 

\[ y \leq \sum_{j \in S} x_j / s < 1, \]

which implies that \( y = 0 \) by the conditions presented in 3.29. If all \( x_j \)'s are equal to 1, then \( y = 1 \). When \( y \geq 1 \), this implies that \( y = 1 \) by the conditions presented in equation (3.29), and thus, constraint (3.27) is redundant (Li and Sun, 2006).

By applying this theorem, each nonlinear term in the objective function or the constraints can be linearized. For instance, the non-linear term \( X_{m-1,0} X_{m, h} \) existing in the objective function can be replaced by a new variable, \( Y_{mghv} \), by adding two constraints. Therefore,

\[
Y_{mghv} = X_{m-1,0} X_{m, h} \quad \forall m \in D; h, g \in \{I \cup J\} \tag{3.30}
\]

\[
\begin{cases}
X_{m-1,0} + X_{m, h} - Y_{mghv} \leq 1 \\
-Y_{m-1,0} - X_{m, h} + 2Y_{mghv} \leq 0
\end{cases} \quad \forall m \in D; \forall h, g \in I \tag{3.31}
\]

The same concept can be applied to all non-linear terms. The additional list of variables defined for the purpose of linearization is as follows:

\[
U_{mghv} = Y_{mghv} Y_{mv} \quad \forall m \in D; g, h \in I; \forall v \in V \tag{3.32}
\]

\[
P_{mv} = 1 - P_{mv} \quad \forall m \in \{D / \{N\}\}; \forall v \in V \tag{3.33}
\]

\[
E_{mv} = \prod_{m' = m + 1}^{N} P_{m', mv} \rightarrow s = N - m + 1 \quad \forall m \in \{D / \{N\}\}; \forall v \in V \tag{3.34}
\]

\[
\vartheta_{mghv} = E_{mv} U_{mghv} \quad \forall m \in D; \forall m' \in D|m' \leq m\}; \forall h, g \in I \tag{3.35}
\]

\[
\psi_{mghv} = E_{mv} X_{0,0} X_{mghv} \quad \forall m \in D; h \in I; g \in J; \forall v \in V \tag{3.36}
\]

\[\sum_{g \in I} X_{mghv} \] shows whether or not there is a customer assigned to position \( m \) of the route. Hence, it works like a binary variable. Thus, for simplifying the linearization of the problem, \( \sum_{g \in I} X_{mghv} \) is also replaced by a binary variable presented in equation (3.37).

\[
\sum_{g \in I} X_{mghv} - y_{mv} = 0 \quad y_{mv} \in \{0,1\} \tag{3.37}
\]

By using the new variables defined, the linearized model is presented in equations (3.38) to (3.52):
Min

\[
\sum_{v \in V, m \in D} \sum_{g \in I} C(v, m) \left( \sum_{m' = 1}^{m} \sum_{h \in I} \sum_{g \in I} (\frac{F_{gh}}{m} \cdot \nu_{mghv} + \sum_{h \in I} \sum_{g \in I} (\frac{F_{gh}}{m} \cdot \nu_{mghv})) + \right)
\]

\[
\sum_{v \in V, g \in J} \sum_{h \in I} Y_{1ghv} B_g + \sum_{g \in J} q_g O_g
\]

\[
\sum_{v \in V, m \in D} \sum_{g \in I} X_{mgv} = 1 \quad \forall g \in I
\]

(3.39)

\[
\sum_{g \in I} X_{mgv} \leq 1 \quad \forall m \in I; \forall v \in V
\]

(3.40)

\[
\sum_{g \in I} d_g X_{mgv} \leq Y_v \quad \forall v \in V
\]

(3.41)

\[
\sum_{v \in V, g \in I} \sum_{m \in D} X_{mgv} = 0
\]

(3.42)

\[
\sum_{g \in I} \sum_{m \in D} X_{mgv} = 0
\]

(3.43)

\[
\sum_{g \in I} X_{mgv} \geq \sum_{g \in I} X_{mgv} \quad \forall m \in D; \forall v \in V
\]

(3.44)

\[
\sum_{g \in I} \sum_{h \in I} Y_{1ghv} \leq 1 \quad \forall v \in V
\]

(3.45)

\[
O_g \leq \sum_{v \in V, h \in I} Y_{1ghv} \leq K O_g \quad \forall g \in J
\]

(3.46)

\[
1 \leq \sum_{v \in V, g \in I, h \in I} Y_{1ghv} \leq K
\]

(3.47)

\[
\begin{cases}
X_{m-1gv} + X_{mhv} - Y_{mghv} \leq 1 \\
-X_{m-1gv} - X_{mhv} + 2Y_{mghv} \leq 0
\end{cases} \quad \forall m \in D; \forall h, g \in I
\]

(3.48)

\[
\begin{cases}
Y_{mghv} + Y_{mv} - U_{mghv} \leq 1 \\
-Y_{mghv} - Y_{mv} + 2U_{mghv} \leq 0
\end{cases} \quad \forall m \in D; \forall m' \in D \mid m' \leq m; \forall h, g \in I
\]

(3.49)
Here, constraints (3.48) to (3.52) are added to the model for the purpose of linearization. The linearized model is programmed in CPLEX for four different examples, and the data is provided in Table 3.8. To find the fastest method for solving the problem, the examples are then solved with multiple solution strategies, such as branch and bound, clique cuts, GUB cover cuts, implied bound cuts, Gomory fractional cuts, and zero-half cuts.

### TABLE 3.8 INFORMATION REGARDING EXAMPLES SOLVED

<table>
<thead>
<tr>
<th>Example</th>
<th>Number of Customers</th>
<th>Number of DCs</th>
<th>Number of Vehicles</th>
<th>Vehicle Capacities</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>50,70</td>
</tr>
<tr>
<td>#2</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>50,70</td>
</tr>
<tr>
<td>#3</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>50,70</td>
</tr>
<tr>
<td>#4</td>
<td>7</td>
<td>2</td>
<td>3</td>
<td>50,70, 70</td>
</tr>
</tbody>
</table>

Tables 3.9 to 3.12 show the number of cuts and computation times for each solution approach. In the first row of these tables, a value of -1 implies that no cutting strategy is used; 0 implies that the automatic setting of CPLEX for applying cuts is used; 1 implies that cuts are used moderately; and 2 implies that aggressive cuts are used in generating results. As reflected in these tables, the pure branch-and-bound strategy has the smallest computation time. Therefore, in the case where problem sizes are small and computation times are not an issue, model linearization is not needed, and LINGO can be an appropriate interface for the model solution. This conclusion is only true when the problem is not large and travel times are discrete step functions. The evaluation of solution approaches for different types of travel-time functions, demand functions, and problem size is considered as future research.
### TABLE 3.9 RESULTS OF SIX-NODE EXAMPLE #1
WHEN OBJECTIVE FUNCTION VALUE = $5.350 \times 10^2$

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clique Cuts</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>GUB Cover Cuts</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>62</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>0.19</td>
<td>0.34</td>
<td>0.30</td>
<td>0.67</td>
</tr>
</tbody>
</table>

### TABLE 3.10 RESULTS OF SEVEN-NODE EXAMPLE #2
WHEN OBJECTIVE FUNCTION VALUE = $2.675 \times 10^2$

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clique Cuts</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>GUB Cover Cuts</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>6</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>0.34</td>
<td>1.05</td>
<td>0.83</td>
<td>1.42</td>
</tr>
</tbody>
</table>

### TABLE 3.11 RESULTS OF EIGHT-NODE EXAMPLE #3
WHEN OBJECTIVE FUNCTION VALUE = $2.675 \times 10^2$

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clique Cuts</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>GUB Cover Cuts</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>1.97</td>
<td>2.75</td>
<td>2.71</td>
<td>12.47</td>
</tr>
</tbody>
</table>

### TABLE 3.12 RESULTS OF NINE-NODE EXAMPLE #4
WHEN OBJECTIVE FUNCTION VALUE = $2.675 \times 10^2$

<table>
<thead>
<tr>
<th>Cut Setting</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Cuts</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Implied Bound Cuts</td>
<td>0</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Flow Cuts</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Gomory Fractional Cuts</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Zero-Half Cuts</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Solution Time (Seconds)</td>
<td>2.98</td>
<td>4.48</td>
<td>4.51</td>
<td>6.08</td>
</tr>
</tbody>
</table>
3.7 Conclusions

Traffic congestion is a normal phenomenon especially in urban areas. Traffic rush hours in the morning and evening typically result in higher travel times. Thus, traffic congestion influences the time taken to travel. The main drawback of existing formulations of the TDLRP is that the waiting time at customers’ location is used to take care of time window violations. In the formulation presented in this chapter, the assumption of time windows and waiting time at customers’ location are relaxed, and a step-by-step formulation of the TDLRP for several scenarios has been developed. In the initial formulation, time windows were eliminated, and a TDLRP formulation that eliminates the waiting time at customer locations was first developed. The formulation was then extended to include both hard and soft time windows. The model was further enhanced to address the issue of demand variation, i.e., assuming that the demand is a function of time. Each of the formulations is illustrated with the use of an example.

For solving the MINLP presented in section 3.4.1 using CPLEX, depending on the type of travel time function, different methods can be applied. Since the discrete step function is the most common representation for the travel time function in the literature, it is assumed that these functions are discrete step functions. By assuming the discrete step functions to represent travel times, the MINLP can be linearized for a faster solution in CPLEX. For solving the MILP obtained, different solution strategies are applied, including branch and bound, moderate- and aggressive-cut generators, etc. Results show that when travel time is a discrete function, the pure branch-and-bound method can provide results faster than cutting methods. However, as already discussed, depending on the type of travel time function, different solution methods can yield different computation times, and this is a subject for future research. In addition, since the problem is NP-hard, heuristic or meta-heuristic algorithms for solving large problems are required. A closer look at constraints (3.42) and (3.43) reveal that they can be decoupled from the route/vehicle, and hence, column generation or Benders decomposition for finding the exact solution of larger-size problems is possible and can be further investigated.

3.8 References


CHAPTER 4
ENERGY-EFFICIENT LOCATION-ROUTING PROBLEM (EELRP):
GENERIC FORMULATION, HARD AND SOFT TIME WINDOWS, AND TIME-DEPENDENT DEMAND

4.1 Abstract

Sustainability and energy savings have attracted considerable attention in recent years. However, in the traditional location-routing problem (LRP), the objective function has yet to minimize the distance traveled regardless of the amount of energy consumed. Although, distance is one of the major factors determining the energy consumption of a distribution network, it is not the only factor. Therefore, this chapter explains the development of a novel formulation of the LRP that considers energy minimization, which is called the energy-efficient location-routing problem (EELRP). The energy consumed by a vehicle to travel between two nodes in a system depends on many forces. Among those, rolling resistance (RR) and aerodynamic drag are considered in this chapter to be the major contributing forces. The presented mixed-integer non-linear programming (MINLP) finds the best location-allocation routing plan with the objective function of minimizing total costs, including energy, emissions, and depot establishment. The proposed model can also handle the vehicle-selection problem with respect to a vehicles’ capacity, source of energy, and aerodynamic characteristics. The formulation proposed can also solve the problems with hard and soft time window constraints. Also, the model is enhanced to handle the EELRP with dynamic customers’ demands. Some examples are presented to illustrate the formulations presented in this chapter.

Keywords: Mixed-Integer Non-Linear Programming, Vehicle-Routing Problem, Location-Routing Problem, Energy-Efficient Location-Routing Problem with Hard and Soft Time Windows, Location-Routing Problem with Time-Dependent Demand

4.2 Introduction

The problem addressed in this chapter is the energy-efficient location-routing problem (EELRP). The traditional location-routing problem (LRP) is a combination of the location-allocation problem (LAP) and the vehicle-routing problem (VRP) or energy-efficient vehicle-routing problem (EEVRP). The LAP involves finding a set of distribution centers (DCs) among potential DCs and assigning customers to them. The VRP originated from the traveling salesman problem (TPS) and is defined as the problem of finding a set of routes originating from depots to serve customers. Each customer must be visited only once, and all vehicles should return back to the same depot from which they departed. The demand of customers should not exceed the vehicle capacity. The objective of
the VRP is usually minimization of transportation cost based on distance traveled. It has been vastly investigated by researchers (Laporte, 1992; Laporte, et al. 1988; Laporte et al., 2000; Eksioglu et al., 2009).

The increase in fuel costs and other problems caused by fuel consumption, such as emissions, carbon footprint, global warming, etc., increases the importance of an energy-efficient and emission-efficient LRP. Although the number of research and methods developed for the LRP is significant, research is limited relative to the LRP with the objective function of minimizing emissions and energy consumption. Kara et al. (2007) present an energy minimization objective function for the VRP in which all vehicles are identical and have the same capacity, and the problem is separately investigated for symmetric and asymmetric distance matrix. Also, the formulation solves for both the collection problem and the delivery problem. Gusikhin et al. (2010) present a heuristic for solving a mixed-fleet VRP for minimizing fuel consumption and environmental emissions. This heuristic is a variation of the multi-label shortest-path problem, and they did not consider vehicle weight as one of the contributing factors for fuel consumption and emissions. Artmeier et al. (2010) present a generic shortest-path algorithm for battery-powered vehicles in which constraints such as limited cruising range, long recharge times, and energy recovery ability are considered. In that method, the graph theory is applied for formulating and solving the problem.

The increase in energy consumption also results in increased emissions. This is another important aspect of energy consumption that should not be ignored. The amount of emissions differs among vehicles based on their sources of energy. Some sources of energy, such as petroleum-based fuel, are the major source of emissions, while other sources, such as electric power, have very few emissions. Hirashima et al. (2002) present a method for calculating the amount of emissions based on a road gradient factor. They compared the results when the objective is distance minimization and when it is emissions minimization. They show that their method provides a better result in terms of emissions.

An EELRP in which each customer’s demand must be satisfied within a time interval is called an EELRP with a time window. The traditional location-routing problem with a time window (LRPTW) is a class of LRP in which each customer must be visited within a specific time window. It has been an interesting subject of research in the last three decades. Many heuristic and meta-heuristic methods have been developed for solving this problem under different conditions and constraints. Desrochers et al. (1987) provide an early survey on the solution methods of the VRP. The solution methods are classified based on mathematical formulations and models. Braysy and
Gendreau (2005) present a comprehensive survey on the heuristic and meta-heuristic algorithms developed for solving the vehicle-routing problem with time window (VRPTW). Desrochers et al. (1987) provide an early survey of the VRP solution methods, which are developed based on existing mathematical formulations and models. Kallehauge (2008) reviews the formulation and exact algorithm of the VRPTW, and categorized the formulation and exact methods developed for VRPTW into four major categories: arc formulation, arc-node formulation, spanning tree formulation, and path formulation. In the arc formulation of the VRPTW, each arc of an underlying directed graph is associated with a binary variable. Dantzig et al. (1954), Kallehauge et al. (2007), and Mak and Ernst (2007) present an arc formulation of the VRPTW. In the arc-node formulation of the problem, binary variables are also associated with nodes of the directed graph. This method of formulating the VRPTW can be found in the work of Miller et al. (1960) and Bard et al. (2002). The spanning tree formulation, in brief, is “a method to find lower bounds for the VRPTW, with the help of time- and capacity-constrained shortest spanning trees and Lagrangian relaxation or Dantzig-Wolfe decomposition” (Held and Karp, 1970 and 1971). Several researchers have focused on solutions to the path formulation approach in the last two decades (Chabrier, 2006; Cook and Rich, 1999; Danna and Pape, 2005; Desrochers et al., 1992; Feillet et al., 2004; Fisher et al., 1997; Halse, 1992; Houck, 1978; Kohl and Madsen, 1997; Kolen et al., 1987; Larsen, 1999, 2004). Mirzaei and Krishnan (2011) present a node formulation of the LRP in which each node is represented by a set of binary variables. This formulation provides a generic optimization model, which handles the LRP with time-dependent demand (LRPTD) as well as time windows. In this chapter, the formulation presented by the LRP will be extended for application in the EELRP with a time window and with a time-dependent demand.

In the traditional LRP/LRPTW formulation, the result obtained is link-based, i.e., each route is formed by a set of links. The position (order) of customers on each route cannot be determined unless the links are connected in the right order. The interpretation of the sequence of visits in each route is thus obtained after the solution is obtained and hence cannot be used to formulate the problem. On the other hand, in the proposed model, the positions (order) of the customers in each route are presented by a set of binary variables, which can be used for the purpose of formulation and hence is called node formulation. The proposed node-based model can also be used wherever there is any constraint, cost, or risk associated with the sequence of customers in a route. When dealing with EELRP, the weight of the vehicle in each node in the system depends on the sequence of service; hence, node formulation is necessary.
As already mentioned, vehicle energy consumption is considered a function of distance traveled (speed), vehicle weight, coefficient of rolling resistance, and aerodynamic characteristics of the vehicle. The node-based property of the model makes it flexible to involve many parameters in the model. With growing attention toward energy consumption and emission, it is essential to approach the LRP as a trade-off between energy and emission cost and profit. Hence, the proposed formulation in this chapter approaches the LRP with such trade-offs. The main objective of this chapter is summarized as follows:

- Development of a formulation for EELRP that reflects the impacts of energy consumption and emissions in the LRP while taking into consideration the following:
  - The heterogeneous vehicle selection problem in which vehicles have different capacities, aerodynamic characteristics, and source of energy.
  - Both rolling resistance and aerodynamic drag forces.
- Development of a formulation for EELRP that can solve problems with a symmetric or asymmetric distance matrix.
- Development of a formulation for EELRP that can handle time-window restrictions for serving customers and also dynamic demands.

The objective function presented in this chapter is to minimize energy, emissions, and depot-establishment cost, while maximizing profit. The following questions are expected to be answered by solving the model:

- What is the best strategy regarding the location of DCs?
- How are customers allocated to DCs?
- What is the routing plan of DCs to serve customers?
- What vehicle type should be used in each route?

Section 4.3 of this chapter provides a detailed definition of the problem under investigation. A mathematical formulation of the problem and an extension of the problem for solving the EEVRP with a hard time window, a soft time window, and time-dependent demand is presented in section 4.4. Section 4.5 provides illustrative examples. Sections 4.6 and 4.7 provide a summary and conclusions as well as future work under investigation, respectively.

### 4.3 Problem Statement

Notations used to formulate the problem are as follows:
\( N \) \quad \text{Total number of customers} \\
\( M \) \quad \text{Total number of DCs} \\
\( K \) \quad \text{Total number of vehicles} \\
\( I \) \quad \text{Set of customers, } I = \{1, 2, \ldots, N\} \\
\( J \) \quad \text{Set of DCs, } J = \{1, 2, \ldots, M\} \\
\( P \) \quad \text{Set of possible positions that a customer can take in a route, } P = \{1, 2, \ldots, N\} \\
\( V \) \quad \text{Set of vehicles, } V = \{1, 2, \ldots, K\} \\
\( V_v \) \quad \text{Speed of vehicle } v, \forall v \in V \\
\( w_v \) \quad \text{Mass of vehicle } v \text{ when fully loaded (tare mass plus load mass), } \forall v \in V \\
\( \lambda_v \) \quad \text{Emission cost for producing 1 NM energy from vehicle } v, \forall v \in V \\
\( \gamma_v \) \quad \text{Cost of consuming 1 Newton meter (NM) of energy in vehicle } v, \forall v \in V \\
\( A_v \) \quad \text{Frontal area of vehicle } v, \forall v \in V \\
\( C_{d,v} \) \quad \text{Vehicle } v \text{ coefficient of drag, } \forall v \in V \\
\( \psi_{gh} \) \quad \text{Density of air on the road that connect nodes } g \text{ and } h, \forall g, h \in \{I \cup J\} \\
\( Y_v \) \quad \text{Capacity of vehicle } v, \forall v \in V \\
\( g/h \) \quad \text{Index used for all nodes} \\
\( T_{gh} \) \quad \text{Travel time between node } g \text{ and } h, \forall g, h \in \{I \cup L\} \\
\( S_g \) \quad \text{Service time at node } g, \forall g \in I \\
\( A_{m,v} \) \quad \text{Arrival time at position } m \text{ of route } v, \forall m \in I, \forall v \in V \\
\( X_{mgv} \) \quad \begin{cases} 
1 \text{ if node } g \text{ is in position } m \text{ of the route } v; \\
0 \text{ otherwise} 
\end{cases} \quad \forall g \in \{I \cup J\}; \forall m \in D; \forall v \in V \\
\( C_{g,hv}^{cr} \) \quad \text{Coefficient of rolling resistance between vehicle } v \text{ tires and the road that connects node } g \text{ to node } h, \forall v \in V, \forall g, h \in \{I \cup L\} \\
\( L_{m,v} \) \quad \begin{cases} 
1 \text{ if } m \text{ is the last taken position of route } v; \\
0 \text{ otherwise} 
\end{cases} \quad \forall m \in D; \forall v \in V
\[ O_g = \begin{cases} 
1 \text{ if there is any vehicle assigned to node } g; \\
0 \text{ otherwise } 
\end{cases} \quad \forall g \in J \]

\[ z_{vh} = \begin{cases} 
1 \text{ if vehicle } v \text{ is assigned to node } h; \\
0 \text{ otherwise } 
\end{cases} \quad \forall h \in J; \forall v \in V \]

\[ D_{gh} \text{ Distance between node } g \text{ and } h, \quad \forall g, h \in \{I \cup J\} \]

\[ F_g \text{ Fixed cost for establishing node } g, \quad \forall g \in J \]

\[ d_g \text{ Demand of node } g, \quad \forall g \in I \]

\[ f_g(t) \text{ Demand of node } g \text{ at time } t, \quad \forall g \in I \]

\[ S_g \text{ Cost of departure from node } g, \quad \forall g \in J \]

\[ a_{mv} = \begin{cases} 
1 \text{ if earliest arrival at position } m \text{ of route } v \text{ is violated;} \\
0 \text{ otherwise } 
\end{cases} \quad \forall m \in D; \forall v \in V \]

\[ \beta_{mv} = \begin{cases} 
1 \text{ if latest arrival at position } m \text{ of route } v \text{ is violated;} \\
0 \text{ otherwise } 
\end{cases} \quad \forall m \in D; \forall v \in V \]

\[ a_g \text{ Earliest arrival time at customer } g, \quad \forall g \in I \]

\[ b_g \text{ Latest arrival time at customer } g, \quad \forall g \in I \]

\[ \Delta a_g \text{ Maximum deviation permitted from earliest arrival time at customer } g, \quad \forall g \in I \]

\[ \Delta b_g \text{ Maximum deviation permitted from latest arrival time at customer } g, \quad \forall g \in I \]

\[ \rho_g \text{ penalty costs associated to lower time limit violation at customer } g, \quad \forall g \in I \]

\[ \phi_g \text{ Penalty costs associated with upper time limit violation at customer } g, \quad \forall g \in I \]

\[ \lambda \text{ Lost-order cost} \]

\[ \gamma \text{ Percentage change in unit price of extra product delivered} \]

\[ \Omega \text{ Profit obtained from selling a unit of product} \]

\[ u \text{ Unit product mass} \]

\[ g^G \text{ Gravitational acceleration, } 9.81 \text{ m/s}^2 \]

\[ w^I_g = \begin{cases} 
1 \text{ if demand has decreased at the delivery time to customer } g; \\
0 \text{ otherwise } 
\end{cases} \quad \forall g \in I \]

\[ w^I_g = \begin{cases} 
1 \text{ if demand has increased at the delivery time to customer } g; \\
0 \text{ otherwise } 
\end{cases} \quad \forall g \in I \]
The supply chain network for this problem consists of a set of customers, a set of potential distribution centers, a plant, and a set of available vehicles. The vehicles can consume different types of fuel, such as electricity, gasoline, diesel, coal, etc. Two costs are associated with the consumption of unit energy in a vehicle: energy cost and emissions cost. Other costs such as greenhouse gas emissions, etc., also can be simply added to the model. Vehicles depart DCs fully loaded and travel at a constant speed. There is only a single type of product in this problem. Each customer demand must be less than the vehicle capacity, i.e., $d_g \leq VC$, $\forall g \in I$.

The total transportation cost includes the cost of departure from the DCs ($S_g, \forall g \in L$), energy cost, and emissions cost. The energy cost is equal to the total energy consumed by a vehicle times its energy unit cost, $\gamma_v$. The amount of energy used (work) to travel between each pair of nodes in the network is presented by equation (4.1).

$$W = Force \times Acceleration \times Distance$$  \hspace{1cm} (4.1)

Under a steady state of driving on flat ground at a constant speed, equation (4.1) can be rewritten as

$$Work = Force \times Distance$$  \hspace{1cm} (4.2)

where force is the steady-state force required to overcome friction and aerodynamic drag. The friction force is the rolling resistance, which is calculated using equation (4.3):

$$Rolling\ Resistance\ (RR) = C_{rrghv} \times Mass \times g^G$$  \hspace{1cm} (4.3)

where $C_{rrghv}$ is the coefficient of rolling resistance and depends on the vehicle tire and road surface. The gravitational acceleration is almost the same and equal to 9.81 m/s$^2$ for all objects. The aerodynamic drag force in the steady state is given by equation (4.4):

$$F_A = \frac{1}{2} \times Cd_v \times A_v \times \psi_{gh} \times V_v^2$$  \hspace{1cm} (4.4)

where $Cd_v$ and $A_v$ depend on the characteristics of the vehicle $v$. Hence, the EELRP is not only sensitive to vehicle capacity but also influenced by characteristics of vehicles with respect to energy consumption. The variable $Cd_v$ describes the smoothness of the vehicle shape. During the vehicle design stage, improving the drag coefficient is a high priority. In addition, the frontal area ($A_v$) is just as important. The variable $\psi_{gh}$ is the density of air in the road between nodes $g$ and $h$, and is usually about 1.3 kg/m$^3$ but can vary with temperature and barometric pressure.

By substituting equations (4.3) and (4.4) into equation (4.2), work can be presented by equation (4.5):

$$Work = (C_{rrghv} \times Mass \times g^G + \frac{1}{2} \times Cd_v \times A_v \times \psi_{gh} \times V_v^2) \times Distance$$  \hspace{1cm} (4.5)
From equation (4.5), it is important to calculate the weight of vehicles at each node. The vehicle mass at each node depends on the previous customers’ demands on the same route; hence, the vehicle mass after delivering load $d_g$ at position $m$ on route $v$ is calculated by equation (4.6):

$$
\Gamma_{mv} = w_v - \sum_{m'=1}^{m} \sum_{g \in J} u_g d_{m'} X_{m'g} \sum_{g \in J} X_{mgv} \quad \forall m \in P, v \in V
$$

Variables $L_{mv}$ in equation (4.7) and $z_{vh}$ in equation (4.8) are defined to simplify the objective function and constraints. A binary variable, $L_{mv}$, is used to identify the last customer on a route and is defined by the recursive equations presented in equation (4.7).

$$
L_{mv} = \begin{cases} 
\sum_{g \in J} X_{mgv} & \quad m = N \\
\prod_{m'=m+1}^{N} (1 - L_{m'}) \sum_{g \in J} X_{mgv} & \quad \forall m \in \{P / \{N\}\}, v \in V
\end{cases}
$$

Location 0 of each route is reserved for a DC. However, even if the related binary variable, $X_{0gv}, \forall g \in L, \forall v \in V$, holds a value of 1, this does not mean that node $g$ is selected to be the assigned DC for route $v$. Node $g$ will not be route $v$’s DC unless there is a link between the depot and a customer in the network. Hence, $z_{vh}$ is introduced to specify whether there is a connection between a depot and a customer in the system. The variable $z_{vh}$ works like a connectivity constraint between DCs and routes, and connects the location decision to the routing decision.

$$
z_{vh} = \sum_{g \in J} X_{hvg} X_{jgv} \quad \forall h \in J, v \in V
$$

The following section presents the mathematical formulation of the problem.

### 4.4 Mathematical Formulation of the Problem

#### 4.4.1 Formulation of EELRP

The objective function of the problem is to minimize the total cost of the system while maximizing profit. The mathematical formulation for EELRP is as follows:
Objective Function:

\[
\text{Min} \quad E_{\text{net}} = \sum_{v \in V} \sum_{m \in P} \sum_{h \in \{1,\ldots,J\}} \sum_{g \in \{1,\ldots,J\}} D_{gh} \left( X_{m-1gv} X_{mv} + X_{mgv} X_{ghv} L_{m} \right) \left( g^G C_{ghv} G_{mv} + \frac{1}{2} c_{d,v} A_{gh} V^2 \right) (\gamma_v + \lambda_v) \\
+ \sum_{g \in J} \sum_{v \in V} z_{vg} S_g + \sum_{g \in J} F_g O_g
\]  

(4.9)

Subject to:

\[
\sum_{v \in V} \sum_{g \in J} X_{mgv} = 1 \quad \forall g \in I 
\] 

(4.10)

\[
\sum_{g \in I} X_{mgv} \leq 1 \quad \forall m \in P, v \in V 
\] 

(4.11)

\[
\sum_{g \in J} \sum_{m \in P} d_{g} X_{mgv} \leq V C_v \quad \forall v \in V 
\] 

(4.12)

\[
\sum_{v \in V} \sum_{g \in J} \sum_{m \in P} X_{mgv} = 0
\] 

(4.13)

\[
\sum_{v \in V} \sum_{g \in J} X_{ogv} = 0
\] 

(4.14)

\[
\sum_{g \in \{1,\ldots,J\}} X_{m-1gv} \geq \sum_{g \in J} X_{mgv} \quad \forall m \in P, v \in V
\] 

(4.15)

\[
\sum_{g \in J} z_{vg} \leq 1 \quad \forall v \in V
\] 

(4.16)

\[
O_g \leq \sum_{v \in V} z_{vg} \leq K O_g \quad \forall g \in J
\] 

(4.17)

\[
1 \leq \sum_{v \in V} \sum_{g \in J} z_{vg} \leq K
\] 

(4.18)

Equation (4.9) is the objective function, which minimizes the total cost of the network while maximizing profit. The first term in the objective function calculates energy and emission cost. The second term is the cost of dispatching vehicles from DCs. The third term is the DC’s fixed costs in case they are open.
Equations (4.10) to (4.18) are constraints for this problem. Constraint (4.10) ensures that each customer appears in only one route. Constraint (4.11) guarantees that each position on a route is not taken by more than one customer. Constraint (4.12) ensures that the total demand of customers assigned to a route is less than the vehicle capacity. It is assumed that position zero of each route is reserved for DCs. This assumption implies that DCs cannot take any other position in routes and also that customers cannot take position 0 of their assigned route. The former is enforced by constraint (4.13), and the latter is enforced by constraint (4.14). Constraint (4.15) ensures that position \( m+1 \) on route \( v \) cannot be taken unless position \( m \) is taken. Constraint (4.16) guarantees that a route cannot be assigned to more than one DC. Constraint (4.17) determines if a DC is open or is closed. Constraint (4.18) keeps the total number vehicles between one and the number of available vehicles.

If there is any energy restriction for vehicle \( v \), constraint (4.19) can be added to the model. For instance, for an electric vehicle \( v \) with power restriction, it might be of interest to enforce the vehicle to return to the depot from which it originated before its battery becomes discharged.

\[
\sum_{m \in P} \sum_{h \in f} \sum_{g \in \{J, J\}} D_{gh} X_{mhv} \left( X_{m-1gv} + X_{0gv} L_{mv} \right) \left( g^c C_{gh} \Gamma_{mv} + \frac{1}{2} c d_v A_{vy} g V^2 \right) \leq \pi_v \quad \forall v \in V
\]  

(4.19)

where the left side of the inequality is the energy consumed by vehicle \( v \) to travel its assigned route, and the right side of the inequality is the energy limit of vehicle \( v \).

It is also possible to enforce the tour duration constraint for all or some of the vehicles. Considering the fact that the arrival time at each customer depends on the arrival times at previous customers on the same route, the arrival time at position \( m \) on route \( v \) is calculated using equation (4.20):

\[
A_{mv} = \sum_{n=m+1}^{N} \sum_{h \in f} \sum_{g \in \{I, J\}} X_{n-1gv} X_{mgh} \left( S_g + T_{gh} \right) \sum_{e=d} X_{mve} \quad \forall m \in P, v \in V
\]  

(4.20)

where \( T_{gh} \) is the travel time between nodes \( g \) and \( h \) \( \forall g, h \in \{I, J\} \), and \( S_g \) represents the service time at node \( g \), \( \forall g \in I \). By using the definition of \( L_{mv} \) presented in equation (4.7), the tour completion constraint will be

\[
\sum_{m \in P} \left( A_{mv} + \sum_{h \in f} \sum_{g \in L} T_{gh} X_{0gv} X_{mhv} \right) \leq \chi_v \quad \forall v \in V
\]  

(4.21)

where, the first term on the left side of the inequality is the time elapsed to meet the last customer on route \( v \), and the second term on the left side of the inequality is the travel time from the last customer on route \( v \) to the originated DC. The summation of these two terms is the total tour duration imposed to be less than the tour duration, \( \chi_v \).
Extension of the model to a multi-product case is straightforward. The following section presents an example to illustrate this problem.

4.4.2 Formulation of EELRP with Time Window

An EELRP in which each customer’s demand must be satisfied within a time interval is called an EELRP with a time window. The time window can be hard or soft. In a formulation with a “hard time window,” each customer has an associated time window during which the demand must be met. The vehicle cannot deliver products to the customer before the start of the time window or after the time window has elapsed, i.e., late or early arrival at a customer is not acceptable. In a VRPTW formulation with a “soft time window,” the customer can be served before and after the preferred time window, i.e., early or late arrival at a customer is acceptable up to a predefined limit. However, there is usually a penalty cost associated with the violation of the time window, which results in late or early service to the customer. In this section, the formulation proposed in section 4.1.1 is modified to tackle the LRP with hard and soft time windows.

4.4.2.1 Formulation of EELRP with Hard Time Window

In the LRP with a hard window constraint, a customer’s demand does not dynamically change with time and is either equal to the initial demand or zero, depending on the arrival time. When the time window is hard, no violation from the time intervals is acceptable. Therefore, the time window can be easily represented in the model with a constraint that enforces the arrival time within the related time intervals.

\[
\sum_{g \in I} a_g X_{m, g, v} \leq A_{m, v} \leq \sum_{g \in I} b_g X_{m, g, v} \quad \forall m \in P, v \in V
\]  

Constraint (4.22) must be added to the formulation presented in Section 4.4.1 to enforce the arrival time at customer to be within the related time interval.

4.4.2.2 Formulation of EELRP with Soft Time Window

In addition to solving the EELRP with a hard time window, the model can handle EELRP with soft time window constraints, which allows the time interval violation for serving customers with related penalty costs. The modification will be adding a constraint to enforce the arrival time:

\[
\sum_{g \in I} a_g X_{m, g, v} + \left( \sum_{g \in I} \Delta a_g X_{m, g, v} \right) t_{m, v} \leq A_{m, v} - \sum_{g \in I} b_g X_{m, g, v} + \left( \sum_{g \in I} \Delta b_g X_{m, g, v} \right) \beta_{m, v} \quad \forall m \in P, v \in V
\]  

(4.23)
Constraint (4.23) is used to enforce the arrival time at each spot of a route to be within the related time interval with associated allowed deviations.

The objective function also needs to be modified by adding a term related to the penalty costs associated with the time window violation. This term is presented in equation (4.24).

\[
\sum_{g \in I} \sum_{v \in V} \sum_{m \in N} X_{gmv} \left( \rho_g \alpha_{mv} + \phi_g \beta_{mv} \right) \quad (4.24)
\]

**4.4.3 Formulation of EELRP with Time-Dependent Customer Demand**

The proposed model can also solve the EELRP with a time-dependent demand. In the EELRP with a time window, a customer’s demand does not dynamically change with time and is either equal to the initial demand or zero, depending on the arrival time. The node-based property of the proposed EELRP enables it to solve the problem when customers’ demands are any arbitrary function of time (Mirzaei and Krishnan, 2011). The objective function of the problem is to minimize the total cost of the system, including energy and emission cost, while maximizing profit. Each customer has an initial demand, \( d_g \), which will dynamically change with time after initiating, i.e., \( d'_g = f(d_g, \tau_g) \), \( \forall g \in I \). The special case of this problem is the EELRP with a time window, in which the demand function is defined by equation (4.25) (Mirzaei and Krishnan, 2011):

\[
f_g(\tau_g) = \begin{cases} 
0 & \text{if } \tau_g \leq a_g \\
d_g & \text{if } a_g \leq \tau_g \leq b_g \\
0 & \text{if } \tau_g \geq b_g 
\end{cases} \quad (4.25)
\]

where \( a_g \) is the earliest arrival time at node \( g \), and \( b_g \) is the latest arrival time at node \( g \). Based on the basic assumptions of vehicle-routing problems, \( d'_g \leq VC \), \( \forall g \in I \). If instead of the function presented in equation (4.25) any other function is defined for a customer’s demand, the problem is no longer an LRP with a time window but rather an LRP with a time-dependent demand.

To formulate the problem for the EELRP with the time-dependent demand, it is necessary to calculate arrival times at each customer \( g \) using equation (4.26):

\[
\tau_g = \sum_{v \in V} \sum_{m \in N} X_{gmv} A_{T_{mv}}, \forall g \in I
\]
For customers with a decreasing demand function, there is a “lost-order cost,” which is the cost resulting from not meeting a customer demand completely or partially, i.e., \( D_g - f_g(\tau_g) \lambda \), where \( \lambda \) is the lost-order cost per unit of product. The profit is the product of the total quantity delivered to the customer, the profit per unit of product, and the customer-fulfillment level, i.e., \( D' \Omega B \) in which \( B \) is the customer fulfillment level defined by equation (4.27):

\[
E_g = \frac{f_g(\tau_g)}{d_g} \quad \forall g \in I
\]  

(4.27)

The value of \( E_g \) is dynamic and depends on the time of delivery. The value of \( E_g \) for customers with a monotonously increasing demand function is greater than one, and the value of \( E_g \) value for customers with a monotonously decreasing demand function is less than one. The unit product price for the additional number of products delivered to customers with increasing demand can be different from the initial price. For example, the price can be cheaper due to a quantity discount. The constant \( \gamma \) represents the percentage of decrease or increase in price.

The objective function for the TDLRP with a time-dependent demand is given by

**Objective Function:**

\[
\text{Min } \sum_{v \in V} \sum_{n \in N} \sum_{k \in K} \sum_{g \in \{1, ..., G\}} D_{gh} X_{nvh} (X_{n-1gv} + X_{0gv} L_{mv}) \left( \sum_{g \in \{1, ..., G\}} \left( g^G C^G_{ghv} \Gamma_{mv} + \frac{1}{2} c d_c A \psi_{ghv} V_v^2 \right) (\gamma_v + \lambda_v) \right)
\]

(4.28)

In equation (4.28), the first term determines the energy and emission cost. The second term is the cost of dispatching vehicles from DCs. The third term is the DCs’ fixed establishment cost. The last term includes the lost-order cost, cost/profit of additional items requested by customers, and sales profit, respectively.

Two modifications are required for the set of constraints presented in equations (4.10) to (4.18) in order to handle the TDLRP with a time-dependent demand, as opposed to the formulation in section 4.4.1. First, it is necessary to change the vehicle-capacity constraint of equation (4.12) by equation (4.29) to consider the demand variability.

\[
\sum_{v \in V} \sum_{n \in N} \sum_{k \in K} \sum_{g \in \{1, ..., G\}} f_g(\tau_g) X_{nvg} \leq Y_v \quad \forall v \in V
\]  

(4.29)

Second, the set of constraints presented in equation (4.30) should be added to the set of constraints:
\begin{align}
    w_g^1 (d_g - f_g (r_g)) & \geq 0 \quad \forall g \in I \\
    w_g^2 (d_g - f_g (r_g)) & < 0 \quad \forall g \in I \\
    w_g^1 + w_g^2 & = 1 \quad \forall g \in I
\end{align}

(4.30)

This set of constraints is used to determine whether a customer’s demand at the time of delivery is higher or lower compared to the initial demand.

4.5 Illustrative Examples

4.5.1 Example of EELRP

Figure 4.1 shows a three-layer network problem used to illustrate the proposed mathematical model. The problem consists of one plant, four customers, and two DCs. Nodes 1, 2, 3, and 4 represent the customers, and nodes 5 and 6 represent potential DCs. Coordinates of the nodes and their associated demand are presented in Table 4.1. No demand is associated with depots.

![Three-layer network problem with one plant (P), four customers (nodes 1–4), and two DCs (nodes 5–6)](image)

Figure 4.1. Three-layer network problem with one plant (P), four customers (nodes 1–4), and two DCs (nodes 5–6)

<table>
<thead>
<tr>
<th>Node</th>
<th>Coordinate</th>
<th>Demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(80,45)</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>(70,15)</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>(100,20)</td>
<td>65</td>
</tr>
<tr>
<td>4</td>
<td>(90,10)</td>
<td>60</td>
</tr>
<tr>
<td>5</td>
<td>(75,10)</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>(60,25)</td>
<td>-</td>
</tr>
</tbody>
</table>

TABLE 4.1 NODE COORDINATES AND ASSOCIATED DEMANDS

Four vehicles are available. Information regarding the vehicle types, capacity, initial mass, drag coefficient, frontal area, emissions cost, and energy cost are provided in Table 4.2. The coefficient of the rolling resistance for each of the four vehicles on each road is shown in Table 4.3, Table 4.4, and Table 4.5. Air density is considered to
be the same and equal to 1.3 kg/m³ for all roads. All vehicles travel at a constant speed of 40 kilometer per hour. The travel time between nodes is then calculated accordingly in Table 4.6. Distances are calculated as Euclidean and presented in Table 4.7.

### TABLE 4.2 VEHICLE INFORMATION

<table>
<thead>
<tr>
<th>Vehicle Number</th>
<th>Type</th>
<th>VC (Unit of Product)</th>
<th>Initial Mass (kg) (Tare Mass + Load Mass)</th>
<th>Drag Coefficient</th>
<th>Frontal Area (m²)</th>
<th>Emissions Cost ($)</th>
<th>Energy Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Gasoline</td>
<td>100</td>
<td>6000 = 4000+20×100</td>
<td>0.75</td>
<td>4.0</td>
<td>0.1×10⁻⁴</td>
<td>0.2×10⁻⁴</td>
</tr>
<tr>
<td>2</td>
<td>Electric</td>
<td>70</td>
<td>4200 = 2800+20×70</td>
<td>0.6</td>
<td>3.5</td>
<td>0</td>
<td>0.2×10⁻⁴</td>
</tr>
<tr>
<td>3</td>
<td>Hybrid</td>
<td>70</td>
<td>4200 = 2800+20×70</td>
<td>0.6</td>
<td>3.5</td>
<td>0.5×10⁻⁴</td>
<td>2.7×10⁻⁵</td>
</tr>
<tr>
<td>4</td>
<td>Diesel</td>
<td>115</td>
<td>9600 = 7100+20×125</td>
<td>0.89</td>
<td>4.5</td>
<td>0.2×10⁻⁴</td>
<td>0.1×10⁻⁴</td>
</tr>
</tbody>
</table>

### TABLE 4.3 COEFFICIENT OF ROLLING RESISTANCE FOR VEHICLE 1

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.012</td>
<td>0.013</td>
<td>0.010</td>
<td>0.012</td>
<td>0.013</td>
</tr>
<tr>
<td>2</td>
<td>0.012</td>
<td></td>
<td>0.013</td>
<td>0.013</td>
<td>0.010</td>
<td>0.015</td>
</tr>
<tr>
<td>3</td>
<td>0.013</td>
<td>0.013</td>
<td></td>
<td>0.015</td>
<td>0.030</td>
<td>0.025</td>
</tr>
<tr>
<td>4</td>
<td>0.010</td>
<td>0.013</td>
<td>0.015</td>
<td></td>
<td>0.030</td>
<td>0.025</td>
</tr>
<tr>
<td>5</td>
<td>0.012</td>
<td>0.010</td>
<td>0.030</td>
<td>0.030</td>
<td></td>
<td>0.030</td>
</tr>
<tr>
<td>6</td>
<td>0.013</td>
<td>0.015</td>
<td>0.025</td>
<td>0.025</td>
<td>0.030</td>
<td></td>
</tr>
</tbody>
</table>

### TABLE 4.4 COEFFICIENT OF ROLLING RESISTANCE FOR VEHICLES 2 AND 3

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.010</td>
<td>0.025</td>
<td>0.015</td>
<td>0.017</td>
<td>0.018</td>
</tr>
<tr>
<td>2</td>
<td>0.010</td>
<td></td>
<td>0.010</td>
<td>0.015</td>
<td>0.080</td>
<td>0.010</td>
</tr>
<tr>
<td>3</td>
<td>0.025</td>
<td>0.010</td>
<td></td>
<td>0.012</td>
<td>0.010</td>
<td>0.035</td>
</tr>
<tr>
<td>4</td>
<td>0.015</td>
<td>0.015</td>
<td>0.012</td>
<td></td>
<td>0.020</td>
<td>0.015</td>
</tr>
<tr>
<td>5</td>
<td>0.017</td>
<td>0.080</td>
<td>0.010</td>
<td>0.020</td>
<td></td>
<td>0.010</td>
</tr>
<tr>
<td>6</td>
<td>0.018</td>
<td>0.010</td>
<td>0.035</td>
<td>0.015</td>
<td>0.010</td>
<td></td>
</tr>
</tbody>
</table>

### TABLE 4.5 COEFFICIENT OF ROLLING RESISTANCE FOR VEHICLE 4

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.011</td>
<td>0.022</td>
<td>0.016</td>
<td>0.014</td>
<td>0.010</td>
</tr>
<tr>
<td>2</td>
<td>0.011</td>
<td></td>
<td>0.011</td>
<td>0.010</td>
<td>0.012</td>
<td>0.010</td>
</tr>
<tr>
<td>3</td>
<td>0.022</td>
<td>0.011</td>
<td></td>
<td>0.013</td>
<td>0.022</td>
<td>0.030</td>
</tr>
<tr>
<td>4</td>
<td>0.016</td>
<td>0.010</td>
<td>0.013</td>
<td></td>
<td>0.020</td>
<td>0.019</td>
</tr>
<tr>
<td>5</td>
<td>0.014</td>
<td>0.012</td>
<td>0.022</td>
<td>0.020</td>
<td></td>
<td>0.011</td>
</tr>
<tr>
<td>6</td>
<td>0.010</td>
<td>0.010</td>
<td>0.030</td>
<td>0.019</td>
<td>0.011</td>
<td></td>
</tr>
</tbody>
</table>
TABLE 4.6 TRANSPORTATION TIME BETWEEN NODES (HOURS)

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00</td>
<td>0.53</td>
<td>0.53</td>
<td>0.61</td>
<td>0.59</td>
<td>0.47</td>
</tr>
<tr>
<td>2</td>
<td>0.53</td>
<td>0.00</td>
<td>0.51</td>
<td>0.34</td>
<td>0.12</td>
<td>0.24</td>
</tr>
<tr>
<td>3</td>
<td>0.53</td>
<td>0.51</td>
<td>0.00</td>
<td>0.24</td>
<td>0.03</td>
<td>0.67</td>
</tr>
<tr>
<td>4</td>
<td>0.61</td>
<td>0.34</td>
<td>0.24</td>
<td>0.00</td>
<td>0.25</td>
<td>0.56</td>
</tr>
<tr>
<td>5</td>
<td>0.59</td>
<td>0.12</td>
<td>0.45</td>
<td>0.25</td>
<td>0.00</td>
<td>0.35</td>
</tr>
<tr>
<td>6</td>
<td>0.47</td>
<td>0.24</td>
<td>0.67</td>
<td>0.56</td>
<td>0.35</td>
<td>0.00</td>
</tr>
</tbody>
</table>

TABLE 4.7 DISTANCE BETWEEN NODES (KM)

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>31.6</td>
<td>32</td>
<td>36.4</td>
<td>35.4</td>
<td>28.3</td>
</tr>
<tr>
<td>2</td>
<td>31.6</td>
<td>0</td>
<td>30.4</td>
<td>20.6</td>
<td>7.1</td>
<td>14.1</td>
</tr>
<tr>
<td>3</td>
<td>32</td>
<td>30.4</td>
<td>0</td>
<td>14.1</td>
<td>2</td>
<td>40.3</td>
</tr>
<tr>
<td>4</td>
<td>36.4</td>
<td>20.6</td>
<td>14.1</td>
<td>0</td>
<td>15</td>
<td>33.5</td>
</tr>
<tr>
<td>5</td>
<td>35.4</td>
<td>7.1</td>
<td>26.9</td>
<td>15</td>
<td>0</td>
<td>21.2</td>
</tr>
<tr>
<td>6</td>
<td>28.3</td>
<td>14.1</td>
<td>40.3</td>
<td>33.5</td>
<td>21.2</td>
<td>0</td>
</tr>
</tbody>
</table>

The cost of vehicle departure from node 5 is $45 and from node 6 is $50. The transportation cost (energy and emission cost) from the plant to the DCs is fixed in the planning horizon of the problem and is equal to $250 for node 5 and $200 for node 6. In addition, the fixed cost for establishing DC 1 is $40 and for DC 2 is $35. The profit obtained from selling each unit of product is $100. This problem is a single product with unit product mass of 20 kg.

The mathematical formulation was solved using LINGO optimizer software, and results of the EELRP are shown in Table 4.8. Figure 4.2 shows the network configuration solution for the EELRP. Customers 1 and 4 are assigned to vehicles 1 and 3, respectively, and customers 2 and 3 are assigned to vehicle 4. In this case, both DCs are selected to be open. Thus, customer 1 will be served through DC 1, and the other customers will be served through DC 2. The negative value of the objective function shows the network profit, which is $13,942.

TABLE 4.8 RESULTS OF EELRP

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-13941.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:04:08</td>
</tr>
</tbody>
</table>

Binary Variables with Value of 1: $X_{11}, X_{13}, X_{23}, X_{24}, X_{25}, X_{35}, X_{43}, X_{45}, O_{5}, O_{6}, z_{1}, z_{3}, z_{4}, z_{5}, LS_{11}, LS_{13}, LS_{24}$
Figure 4.2. Network configuration solution for EELRP

The same problem is solved by traditional formulation of the LRP (Mirzaei and Krishnan, 2011). Results of the LRP are presented in Table 4.9. In this case, the transportation cost is assumed to be $1/km. Figure 4.3 shows the network configuration solution for the LRP in which the objective function is the trade-off between distance and profit. By comparing the distance and energy consumption between the two results, presented in Table 4.10, the distance traveled is increased by 7.9%. However, the energy and emission cost has decreased more than 36%.

TABLE 4.9 RESULTS OF LRP

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-24458.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:0:53</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>X_{111}, X_{122}, X_{144}, X_{234}, X_{051}, X_{052}, X_{054}, O_3, z_{15}, z_{25}, z_{46}, LS_{11}, LS_{12}, LS_{24}</td>
</tr>
</tbody>
</table>

Figure 4.3. Network configuration solution for LRP

TABLE 4.10 COMPARISON OF EELRP AND LRP RESULTS

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Distance Traveled (KM)</th>
<th>Energy and Emission Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EELRP</td>
<td>126.10</td>
<td>1553.32</td>
</tr>
<tr>
<td>LRP</td>
<td>116.10</td>
<td>2429.37</td>
</tr>
</tbody>
</table>
4.5.2 Example of EELRP with Hard Time Window

This section illustrates the EELRP with a hard time window. The example is similar to the one presented in section 4.5.1, except that all customers have a time window assigned to them, as shown in Table 4.11. The time interval implies that each customer’s demand is $d_g$ if it is served within the specified time window; otherwise, it is zero.

**TABLE 4.11 TIME INTERVAL ASSIGNED TO EACH CUSTOMER FOR EELRP WITH HARD TIME WINDOW**

<table>
<thead>
<tr>
<th>Customer Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Interval (Hours)</td>
<td>[35,40]</td>
<td>[50,58]</td>
<td>[20,30]</td>
<td>[14,16]</td>
</tr>
</tbody>
</table>

The mathematical formulation was solved using LINGO optimizer software, and results for the EELRP with a hard time window are presented in Table 4.12. Figure 4.4 shows the network configuration solution for the EELRP with hard time window. Customers 1 and 4 are assigned to vehicle 1 and 3, respectively, and customers 3 and 2 are assigned to vehicle 4. In this case, only depot 1 is selected to serve the customers. The negative value of objective function shows a network profit of $7,037.6.

**TABLE 4.12 RESULTS OF EELRP WITH HARD TIME WINDOW**

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-7037.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:00:23</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>$X_{111}, X_{134}, X_{143}, X_{224}, X_{051}, X_{053}, X_{054}, O_5, z_{15}, z_{35}, z_{45}, LS_{11}, LS_{13}, LS_{24}$</td>
</tr>
</tbody>
</table>

![Figure 4.4. Network configuration solution for EELRP with hard time window](image)

By solving the same problem with a traditional LRP with a hard time window, the same network configuration is obtained, except that vehicle 2, instead of vehicle 3, is assigned to serve customer 4. Results for this
configuration are shown in Table 4.13 and Figure 4.5. This happens because the traditional formulation of LRP does not distinguish between the vehicles in terms of energy and emissions, only with respect to their capacities. Thus, there is no difference between vehicles 2 or 3 when using the traditional LRP because they have the same capacity.

### TABLE 4.13 RESULTS OF LRP WITH HARD TIME WINDOW

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-24409.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:00:02</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>$X_{111}, X_{134}, X_{142}, X_{224}, X_{051}, X_{052}, X_{053}, X_{054}, X_{063}, O_5, z_{15}, z_{25}, z_{45}, LS_{11}, LS_{12}, LS_{24}$</td>
</tr>
</tbody>
</table>

Figure 4.5. Network configuration solution for LRP with hard time window

Table 4.14 shows the comparison between results obtained from the LRP and the EELRP with hard time windows. From this table it can be concluded that the EELRP formulation results in more than 18% savings in energy and emission costs while the distance traveled remains the same.

### TABLE 4.14 COMPARISON OF RESULTS OF EELRP AND LRP WITH HARD TIME WINDOWS

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Distance Traveled (KM)</th>
<th>Energy and Emission Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EELRP</td>
<td>165.2</td>
<td>$1717.15</td>
</tr>
<tr>
<td>LRP</td>
<td>165.2</td>
<td>$2099.25</td>
</tr>
</tbody>
</table>

In the next section, the formulation and an example of EELRP with a soft time window is presented.

#### 4.5.3 Example of EELRP with Soft Time Window

The example in this section is similar to the one presented in section 4.5.2 except for time intervals. In this example, violation from the time intervals is allowed up to a specified limit (Table 4.15). A penalty cost, associated with violating a lower or upper limit of a time interval, is also provided in this table.
TABLE 4.15 TIME INTERVAL, VIOLATION, AND PENTALTY COST ASSIGNED TO EACH CUSTOMER FOR EELRP WITH SOFT TIME WINDOW

<table>
<thead>
<tr>
<th>Customer Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Interval (Hours)</td>
<td>[36,40]</td>
<td>[50,55]</td>
<td>[25,30]</td>
<td>[14,14.5]</td>
</tr>
<tr>
<td>Lower Limit Violation Allowed (Hours)</td>
<td>1.5</td>
<td>0.5</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Upper Limit Violation Allowed (Hours)</td>
<td>0.5</td>
<td>2.3</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>Penalty Cost ρg ($)</td>
<td>20</td>
<td>10</td>
<td>50</td>
<td>30</td>
</tr>
<tr>
<td>Penalty Cost φg ($)</td>
<td>50</td>
<td>70</td>
<td>20</td>
<td>10</td>
</tr>
</tbody>
</table>

The mathematical formulation was solved using LINGO optimizer software, and results for the EELRP with a soft time window are shown in Table 4.16. Figure 4.6 shows the network configuration solution. Customers 1 and 4 are assigned to vehicle 1 and 2, respectively, and customers 3 and 2 are assigned to vehicle 4. In this case, only one depot (node 5) is selected to serve the customers. The negative value of the objective function shows the network profit of $6,938.

TABLE 4.16 RESULTS OF EELRP WITH SOFT TIME WINDOW

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-6937.65</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:1:38</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>X_{14}, X_{14}, X_{24}, X_{24}, X_{65}, X_{65}, X_{65}, X_{65}, \beta_{13}, \beta_{13}, \beta_{24}</td>
</tr>
</tbody>
</table>

By solving the same problem with the traditional LRP with a soft time window, the same network configuration is obtained except for a difference in vehicle assignments. Results are shown in Table 4.17 and Figure 4.7. As previously mentioned, this happens because the traditional formulation of the LRP does not differentiate among vehicles in terms of energy and emissions, only with respect to their capacities.
TABLE 4.17 RESULTS OF LRP WITH SOFT TIME WINDOW

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>-24309.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:5:06</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>$X_{14}, X_{131}, X_{142}, X_{221}, X_{051}, X_{052}, O_5, z_{15}, z_{25}, z_{45}, LS_{12}, LS_{14}, LS_{21}, a_{14}, a_{31}, a_{34}, \beta_{12}, \beta_{13}, \beta_{21}, \beta_{31}, \beta_{34}$</td>
</tr>
</tbody>
</table>

![Network configuration solution for LRP with soft time window](image)

Figure 4.7. Network configuration solution for LRP with soft time window

By comparing the results, as shown in Table 4.18, it can be concluded that the proposed formulation will result in more than 26% savings in energy and emission cost, while the distance traveled remains the same as in the traditional formulation.

TABLE 4.18 COMPARISON OF RESULTS OF EELRP AND LRP WITH SOFT TIME WINDOW

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Distance Traveled (KM)</th>
<th>Energy and Emission Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EELRP</td>
<td>165.2</td>
<td>1717.15</td>
</tr>
<tr>
<td>LRP</td>
<td>165.2</td>
<td>2320.61</td>
</tr>
</tbody>
</table>

The following section presents a generic formulation of EELRP that can handle the dynamicity of customers’ demands.

4.5.4 Example of EELRP with Time-Dependent Demand

The example investigated in this section is similar to the example presented in section 4.5.1 However, in this case, it is assumed that each customer’s demand varies with time after initiation according to a function provided in Table 4.19. The mathematical formulation was solved using LINGO optimizer software, and results of the EELRP with a time-dependent demand are shown in Table 4.20. Figure 4.8 shows the network configuration solution. Customers 1 and 4 are assigned to vehicle 4 and 3 respectively, and customers 2 and 3 are assigned to
vehicle 1. In this case, both depots are selected to serve the customers. The negative value of the objective function shows the network profit of $11,791.

**Table 4.19** Customers’ Demand Information for EELRP with Time-Dependent Demand

<table>
<thead>
<tr>
<th>Node</th>
<th>Coordinate</th>
<th>Initial Demand</th>
<th>Demand Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(80,45)</td>
<td>100</td>
<td>100-0.2τ₁</td>
</tr>
<tr>
<td>2</td>
<td>(70,15)</td>
<td>25</td>
<td>25-0.2τ₁</td>
</tr>
<tr>
<td>3</td>
<td>(100,20)</td>
<td>65</td>
<td>65-0.3τ₁</td>
</tr>
<tr>
<td>4</td>
<td>(90,10)</td>
<td>60</td>
<td>60-0.5τ₁</td>
</tr>
<tr>
<td>5</td>
<td>(75,10)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>(60,25)</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**Table 4.20** Results of EELRP with Time-Dependent Demand

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-11791.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:04:14</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>X₁₁₄, X₁₂₁, X₁₄₂, X₂₃₁, X₀₅₁, X₀₅₂, X₀₆₃, O₅, O₆, z₁₅, z₅₁, z₄₅, LS₁₅, LS₁₄, LS₂₁, Γ₁, Γ₄</td>
</tr>
</tbody>
</table>

Figure 4.8. Network configuration solution for EELRP with time-dependent demand

The same problem is solved using the traditional formulation of the LRP. Results are presented in Table 4.21, and Figure 4.9 shows the network configuration solution.

**Table 4.21** Results of LRP with Time-Dependent Demand

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>$-24469.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computation Time</td>
<td>00:01:15</td>
</tr>
<tr>
<td>Binary Variables with Value of 1</td>
<td>X₁₁₄, X₁₂₁, X₁₄₂, X₂₃₁, X₀₅₁, X₀₅₂, X₀₆₃, O₅, O₆, z₁₅, z₅₁, z₄₅, LS₁₅, LS₁₄, LS₂₁, Γ₁, Γ₄</td>
</tr>
</tbody>
</table>
The distance traveled along with energy and emission cost obtained from the EELRP and LRP with a time-dependent demand are compared in Table 4.22. It can be concluded that distance traveled to serve the customers from depots is reduced more than 10% (traveled distance does not include the distance from the plant to depots, because it is already considered a fixed cost of establishing the depots). In addition, the energy and emission cost has been reduced by more than 37% in the routes obtained by the EELRP with the time-dependent demand.

**TABLE 4.22 COMPARISON OF RESULTS OF EELRP AND LRP WITH TIME-DEPENDENT DEMAND**

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Distance Traveled in Tours Serving Customer (KM)</th>
<th>Energy and Emission Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EELRP with Time-Dependent Demand</td>
<td>126.10</td>
<td>1447.01</td>
</tr>
<tr>
<td>LRP with Time-Dependent Demand</td>
<td>140.30</td>
<td>2306.18</td>
</tr>
</tbody>
</table>

**4.6 Conclusions**

In this chapter, the traditional LRP is approached with a novel perspective that considers energy and emission cost of a logistics problem. In the presented mathematical formulation, the objective function is the trade-off between energy and emission cost and profit. In addition, the model deals with the problem of vehicle selection when the available vehicles have different sources of energy, different aerodynamic structures, and consequently different costs of fuel and emission, while the traditional LRP is sensitive only to vehicle capacity. The node-based property of the presented model enhances the flexibility of the model. To illustrate the flexibility of the formulation, other possible scenarios of the EELRP, such as the EELRP with a soft or hard time window and the EELRP with a time-dependent demand are formulated. The mathematical formulations are illustrated with examples. The
comparison of results between the EELRP and LRP shows that the model provides more economical solutions in terms of energy and emission cost.

Since the problem is NP-hard, for networks with a high number of nodes, heuristic or meta-heuristic algorithms for solving the problem are required. A closer look at constraints (4.13) and (4.14) reveals that they can be decoupled from the route/vehicle, and hence, applying column generation or Benders decomposition for finding the exact solution of larger-size problems may be possible. Adopting heuristic methods to tackle large-size problems is currently under investigation. The node-based formulation is also critical for configuring supply networks under risk associated with the sequence of customer visits. This aspect of the problem also is currently under investigation.

4.7 References


CHAPTER 5
HEURISTIC ALGORITHM FOR ENERGY-EFFICIENT VEHICLE-ROUTING PROBLEM

5.1 Abstract

Sustainability and energy consumption has attracted considerable attention in recent years. In the vehicle-routing problem (VRP), the objective traditionally is to minimize the distance traveled, regardless of the amount of energy consumed. The energy consumed by a vehicle to travel between two nodes depends on many parameters. Among those, vehicle weight and distance traveled are considered in this chapter as the major contributing factors. The presented heuristic method is a type of “savings” algorithms, which are well-known heuristics developed for the VRP because of their simplicity and computation speed in providing results with near-optimal solutions. The presented algorithm finds the best routing plan with the objective of minimizing total energy cost. Several benchmark problems are presented to evaluate and compare the performance of the proposed model with the traditional savings method.

Keywords: Vehicle Routing, Energy Efficiency, Heuristics

5.2 Introduction

The problem addressed in this chapter is the energy-efficient vehicle-routing problem (EEVRP). The vehicle-routing problem (VRP) originated from the traveling salesman problem (TPS) and is defined as the problem of finding a set of routes originating from depots to serve customers. Each customer must be visited only once, and all vehicles should return to the same depot from which they departed. Customer demand should not exceed vehicle capacity. The objective of the VRP is usually minimization of the distance traveled. In the context of energy minimization, minimization of distance traveled does not guarantee minimum energy consumption in the system. This problem has been extensively investigated by researchers (Eksioglu et al., 2009; Laporte, 1992; Laporte et al., 2000).

The increase in fuel cost and other problems caused by fuel consumption, such as emission, carbon footprint, global warming, etc., increases the importance of and energy- and emission-efficient location-routing problem (LRP). Although the number of research and methods developed for the LRP is significant, research specifically studying LRP with the objective function of minimizing emission and energy consumption is limited.

With the increased attention toward energy consumption and the skyrocketing price of fuel, it is necessary to approach the VRP with an objective of minimizing energy consumption. Although, distance is one factor that
determines the energy consumption of a vehicle, it is not the only parameter. Vehicle weight is another factor. Most of the literature has focused on minimizing the distance/time of travel in the network. In the context of energy consumption, the shortest path/route is not necessarily the most energy-efficient route; sometimes serving a customer at a farther distance with a heavier demand can be a more energy efficient than serving customers who are closer with a lighter demand. Little attention has been paid to addressing energy-efficient vehicle-routing problems. Kara et al. (2007) presents an energy minimization objective function for the VRP. In their model, they assume that all vehicles are identical and have the same capacity. The problem is separately investigated for symmetric and asymmetric distance matrix for both collection and delivery problems. Shokri et al. (2009) present a method to find the most fuel-efficient traveling path based on traffic congestion information in real-time problems. In their work, the factor affecting fuel efficiency is travel time, which is considered a function of traffic level. Hence, the effect of different traffic levels on travel time and consequently fuel consumption is investigated. Gusikhin et al. (2010) present a heuristic for solving the mixed-fleet VRP for minimizing fuel consumption and environmental emissions. This heuristic is a variation of the multi-label shortest-path problem, and does not consider vehicle weight as one of the contributing factors for fuel consumption and emissions. Armeier et al. (2010) present a generic shortest-path algorithm for battery-powered vehicles in which constraints such as limited cruising range, long recharge times, and energy recovery ability are considered. Mirzaei and Krishnan (2011) present a mixed-integer nonlinear programming (MINLP) for the LRP in which the objective function minimizes the energy-consumption cost. The formulation encompasses both the location-allocation and vehicle-routing problems. However, the problem formulated is NP-hard, and when the number of nodes is increased, the formulation cannot yield an exact solution in a reasonable time. Hence, this chapter proposes the use of a heuristic method to solve large-size problems. The proposed heuristic developed in this chapter is the energy-efficient version of the Clark-Wright (CW) algorithm and hence is named the energy-efficient Clark-Wright (EECW) algorithm.

As previously mentioned, vehicle energy consumption is considered a function of distance and vehicle weights in this chapter. In physics, work is the amount of energy transferred by a force that acts through a distance, as presented in equation (5.1).

\[ W = \tilde{F} \cdot \tilde{d} \]  

(5.1)

where \( \tilde{F} \) and \( \tilde{d} \) are force and distance vectors, respectively. Under a steady state of driving on flat ground at a constant speed, the force in equation (5.1) is the force required to overcome friction and aerodynamic drag. If the
aerodynamic drag of a vehicle is ignored, then in the steady state, the friction force is the rolling resistance (RR), which is calculated using equation (5.2):

\[ \vec{F}_{RR} = C_{RR} m g^G \]  

(5.2)

where \( C_{RR} \) is the coefficient of rolling resistance and depends on the vehicle tire and road surface, and \( m g^G \) is the vehicle weight. By substituting equation (5.2) into equation (5.1), work can be presented by equation (5.3):

\[ W = \vec{F}_{RR} \vec{d} \]  

(5.3)

The factor \( C_{RR} \) can be assumed to be the same for a given type of route and tire, and hence can be eliminated from the calculations. However, inclusion of this factor in the calculation is straightforward. With this definition, the proposed EECW algorithm includes the impacts of vehicle weight and distance traveled on transportation cost, which can be significant in most real-case scenarios. The objective of the EECW algorithm is to minimize the energy cost (rolling resistance) by finding the proper routing plan in which the \( \vec{F}_{RR} \vec{d} \) is minimized. In fact, fuel economy, which is usually represented by miles per gallon (MPG) for a gasoline-powered vehicle, is a function of its weight and usually decreases as the vehicle weight increases; as a result, the transportation and emission cost increases.

Section 5.3 of this chapter provides a detailed survey of savings methods and its enhancement in the literature. The proposed algorithm is presented in section 5.4. In section 5.5, several benchmark problems are solved, and the computation results are compared with the CW algorithm. Section 5.6 provides a summary and conclusions as well as future work under investigation.

### 5.3 Clark-Wright Savings Algorithm

Saving algorithms are very popular heuristic algorithms developed for the VRP. They are commonly used in commercial packages because of their simplicity and speed in providing a near-optimal solution. Savings methods are more time and cost-effective than meta-heuristic algorithms. The results obtained from savings methods are sometimes used as the initial solution for launching meta-heuristic methods.

This chapter is based on the savings method developed by Clark-Wright. In this algorithm, it is initially assumed that each customer will be served by a vehicle. Then, the amount of savings in distance traveled will be calculated by merger of each pair of customers using equation (5.4).
\[ S_{ij} = (C_{0i} + C_{i0} + C_{0j} + C_{j0}) - (C_{0i} + C_{ij} + C_{j0}) = C_{i0} + C_{0j} - C_{ij} \] (5.4)

Figure 5.1 shows the merging routes in the savings method. Here, node 0 is a depot, and nodes \( i \) and \( j \) are customers. The amount of savings will be calculated for each possible pair of customers. Therefore, if \( N \) customers exist in the network, then \( \binom{N}{2} \) savings must be calculated. Since the distance matrix is considered symmetric, there is no difference between \( S_{ij} \) and \( S_{ji} \), and the result obtained from the CW algorithm is insensitive to the direction of routes. After calculating the savings using this savings algorithm, routes are determined. Details of the algorithm can be found in the work of Larson and Odoni (1981).

Features have been developed to enhance the accuracy of the savings method. Gaskell (1967) and Yellow (1970) discussed the basis of the savings method and presented a modified version for calculating the savings, as shown in equation (5.5):

\[ S_{ij} = C_{i0} + C_{0j} - \lambda C_{ij} \] (5.5)

Based on this modified version, \( \lambda \) is the “shape parameter,” which reflects the emphasis on distance between nodes over distance between depot and nodes. This modification boosts the savings performance when the distance between \( i \) and \( j \) is smaller relative to the depot. It also revises the original CW method by preventing the formation of circle-shaped routes.

Paessens (1988) presented another approach to enhance the CW savings method by adding another parametric term to equation (5.6). This new term signifies the difference in the distance between customers \( i \) and \( j \) from the depots.

\[ S_{ij} = C_{i0} + C_{0j} - \lambda C_{ij} + \mu |C_{0i} - C_{j0}| \] (5.6)

Altinel and Oncan (2005) defined another parametric term to be added to equation (5.6) for the purpose of increasing the vehicles capacity utilization. Equation (5.7) shows this extra term:
This extra term will maximize the utilization of vehicle capacity by assigning the customers for a route. The impact of this extra term can be comprehended by comparing the saving values obtained from equations (5.6) and (5.7). For instance, when two pairs of nodes have the same savings values based on equation (5.6), equation (5.7) ensures that the pair with greater customer demand is selected first.

Parametric terms added to the original savings method can improve CW performance if they are properly defined. Battarra et al. (2007) developed a two-stage method using a genetic algorithm and a local search to determine the proper set of parameters for equation (5.7), including $\lambda$, $\mu$, and $v$.

All of these methods and efforts are developed to minimize the total distance traveled and bring the solution closer to the optimum. In this chapter, the original CW savings method is modified to solve the energy-efficient VRP.

5.4 Energy-Efficient Clark-Wright Savings Algorithm

The following notations are used to explain the savings method for the EECW savings algorithm:

$I$ Set of customers

$NA$ Set of customers not assigned to a depot

$A$ Set of customers assigned to a depot

$w_0$ Vehicle initial weight

$w_i$ Weight of customer $i$'s demand

$w_T$ Vehicle tare weight

$L_0$ Weight of vehicle initial load

$d_{ij}$ Distance between node $i$ and $j$

$D_i$ Demand of customer $i$

$D_{ip}$ Demand of customer $i$ for product $p$

$u$ Unit product weight

$u_p$ Product $p$ unit weight

$P$ Set of products

$S_{ij}$ Savings resulting from merging routes of customers $i$ and $j$
Vehicle capacity in terms of number of products contained therein

Vehicle capacity in terms of weight

5.4.1 Calculation of Savings Values for EECW Algorithm

It is assumed that the fleet is homogenous and vehicles depart the depot with full capacity. The merger process in the new savings algorithm will be the same. However, the amount of savings will be calculated based on energy consumed. To calculate the amount of energy consumed in each link, the initial weight of the vehicle must be known. This is usually the summation of the vehicle tare weight and vehicle load. Equation (5.8) represents the initial weight of the vehicle. In real-life scenarios, a vehicle capacity is represented in terms of kg or lb load that it can carry.

\[ w_0 = w_T + L_0 \] (5.8)

When there is only one type of product to be delivered to customers, each customer’s demand weight will be

\[ w_i = uD_i \quad \forall i \in I \] (5.9)

If the problem involves a multi-product, then each customer’s demand can be calculated by

\[ w_i = \sum_{p \in P} u_p D_{ip} \quad \forall i \in I \] (5.10)

Then the amount of savings will be calculated by

\[
S_{ij}^E = (w_i D_{ij} + w_j D_{ij}) - (w_i D_{ij} + w_j D_{ij} + (w_i - w_j) D_{ij}) - (w_i D_{ij} + (w_i - w_j) D_{ij}) - (w_i D_{ij} + w_j D_{ij}) \quad \forall i, j \in I, i \neq j
\] (5.11)

where the upper index of \( E \) represents the savings value related to energy consumption.

5.4.2 Modified Clark-Wright Savings Algorithm

The CW algorithm is insensitive to route direction. However, the EECW algorithm is sensitive to the direction of travel on a route. In other words, if nodes \( i \) and \( j \) are linked, then the amount of savings for route 0-i-j-0 and route 0-j-i-0 in the traditional CW algorithm is the same. However, in the EECW algorithm, the savings could be different for each route. Hence, in the EECW algorithm, the savings are not symmetric, and the CW algorithm must be revised to become direction sensitive. The EECW algorithm follows:
0- Start.

1- Calculate equation (5.11) for each pair of nodes.

2- Sort the obtained value in descending order and create a savings list.

3- Select the highest savings in the list if no constraint will be violated for any $S_{ij}$.

4- If neither $i$ nor $j$ is already assigned to a route, then start a new route with these nodes; otherwise, go to 5.

5- If $i$ is already assigned and is the last node in its route, then insert $j$ after $i$ in the same route; otherwise, go to 6.

6- If $j$ is already assigned and is the first node on its route, then insert $i$ before $j$ in the same route; otherwise, go to 7.

7- If both $i$ and $j$ are already assigned to different routes and $i$ is the last node in one route and $j$ is the first node in the other route, then merge the two routes; otherwise, go to 8.

8- Eliminate $S_{ij}$ from the savings list, and if the list is not yet exhausted, then go to 3; otherwise, stop.

Consider the set of routes formed during steps 4 to 7 to be the answer. Make a list of customers which are not assigned, put them in a new route, and add the route to the set of previously found routes.

In the above algorithm, nodes refer to customers. In the following section, the computation results applying the original CW and EECW algorithms are presented, and these are compared in terms of distance and energy consumption.

5.5 Computation Results

Two sets of benchmark problems from the literature were used to test the performance of the new algorithm. For the purpose of comparison and to include the impact of energy usage, these problems were modified to include the vehicle tare weight and product unit weight. The results obtained from the CW and EECW algorithms are compared in terms of distance traveled and the amount of energy saved. The distances are considered symmetric in the benchmark problems. Also, in the original CW method, the direction of routes in an obtained solution is not important with respect to the distance traveled. For a CW algorithm solution that has $n$ routes, there are $2^n$ combinations of routes (direct and reverse), which result in the same distance traveled. However, the EECW algorithm is sensitive to the direction of routes, and each route should be exactly implemented in the order provided by the solution. For the purpose of comparison, only one set of routes obtained from the CW method is considered.
First, a set of medium-size benchmarks was selected from the work of Augerat et al. (1995) (A, B benchmark problems), Christofides and Eilon (1969) (E benchmark problems), and Fisher (F benchmark problems). These instances can be downloaded from the website [http://www.branchandcut.org/](http://www.branchandcut.org/), and the problems can be tested for any vehicle tare weight and product unit weight. In this set of problems, it is assumed that the vehicle capacity and tare weight of each vehicle are both 5,000 kg, while all other data used are the same as in the benchmark problem. To keep the vehicle capacity similar to that in the original benchmark problems in terms of units that it can contain, the product unit weight is calculated according to equation (5.9).

\[ u = \frac{V_u}{V} \quad (5.12) \]

Using equation (5.12), the vehicle capacity in terms of the number of products that it can contain will remain the same as in the original benchmark problem, thus making the comparison possible. Table 5.1 shows results from solving the benchmark problems. From Table 5.1, it can be seen that the EECW algorithm for most of the benchmark problems outperforms the original CW method in terms of energy consumed. In five of the 27 “A” benchmark problems, the EECW algorithm resulted in a better solution in terms of energy consumption. A closer look at these benchmarks shows that in 7 out of 37 cases, the EECW algorithm could not outperform the CW algorithm in terms of energy consumption. The maximum energy savings using the EECW algorithm is 8.23%. The EECW algorithm also outperformed the CW algorithm in terms of distance traveled in more than 43% of the problems solved. There was no benchmark problem for which the distance traveled is the same for the two algorithms. This implies that the difference in energy consumptions is the result of selecting a different (improved) routing plan and not a different direction of travel in the routing plan.

In Table 5.1, the two methods are compared in terms of computation time. The times recorded are the sum of the times used for implementing each algorithm, times for calculating the distance traveled, and times to calculate the energy consumed. The case studies were run on an Intel Core Quad CPU, Q8400 @ 2.66 GHz, with 4 GB RAM and a 32-bit operating system. As shown in Table 5.1, the computation time for all problems is very small. On average, the computation time is 0.358 second using the CW algorithm and 0.354 second using the EECW algorithm, which indicates that, on average, the EECW algorithm is 0.004 second faster than the original CW algorithm. This occurs because of the sensitivity of the EECW algorithm with respect to the direction of the routes.
TABLE 5.1 RESULTS OF BENCHMARK PROBLEMS (PROBLEM SETS A, B, E, AND F)

<table>
<thead>
<tr>
<th>No.</th>
<th>Problem</th>
<th>CW</th>
<th>EECW</th>
<th>Extra Distance Traveled by Applying EECW (%)</th>
<th>Energy Saved by Applying EECW (%)</th>
<th>Computation Time of CW (seconds)</th>
<th>Computation Time of EECW (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A-n32-k5</td>
<td>844</td>
<td>647344449</td>
<td>801</td>
<td>62923185</td>
<td>-5.016</td>
<td>2.798</td>
</tr>
<tr>
<td>2</td>
<td>A-n33-k5</td>
<td>694</td>
<td>52417623</td>
<td>697</td>
<td>53583459</td>
<td>0.513</td>
<td>-2.224</td>
</tr>
<tr>
<td>3</td>
<td>A-n33-k6</td>
<td>776</td>
<td>62496367</td>
<td>775</td>
<td>57261209</td>
<td>-0.200</td>
<td>8.293</td>
</tr>
<tr>
<td>4</td>
<td>A-n34-k5</td>
<td>807</td>
<td>61937565</td>
<td>799</td>
<td>60279083</td>
<td>-1.049</td>
<td>2.677</td>
</tr>
<tr>
<td>5</td>
<td>A-n36-k5</td>
<td>831</td>
<td>64603932</td>
<td>838</td>
<td>62296017</td>
<td>0.811</td>
<td>3.572</td>
</tr>
<tr>
<td>6</td>
<td>A-n37-k5</td>
<td>706</td>
<td>56694645</td>
<td>720</td>
<td>53734660</td>
<td>2.165</td>
<td>4.206</td>
</tr>
<tr>
<td>7</td>
<td>A-n37-k6</td>
<td>980</td>
<td>74632359</td>
<td>979</td>
<td>74617043</td>
<td>-0.022</td>
<td>0.021</td>
</tr>
<tr>
<td>8</td>
<td>A-n38-k5</td>
<td>795</td>
<td>62960861</td>
<td>770</td>
<td>57781241</td>
<td>-3.166</td>
<td>8.227</td>
</tr>
<tr>
<td>9</td>
<td>A-n39-k5</td>
<td>902</td>
<td>69506721</td>
<td>880</td>
<td>65845261</td>
<td>-2.421</td>
<td>5.268</td>
</tr>
<tr>
<td>10</td>
<td>A-n39-k6</td>
<td>883</td>
<td>66945672</td>
<td>873</td>
<td>65019035</td>
<td>-1.055</td>
<td>2.878</td>
</tr>
<tr>
<td>11</td>
<td>A-n4-k7</td>
<td>1021</td>
<td>77698933</td>
<td>986</td>
<td>72141634</td>
<td>-3.419</td>
<td>7.152</td>
</tr>
<tr>
<td>12</td>
<td>A-n4-k6</td>
<td>1013</td>
<td>79621012</td>
<td>1015</td>
<td>73358168</td>
<td>0.164</td>
<td>7.866</td>
</tr>
<tr>
<td>13</td>
<td>A-n4-k7</td>
<td>1200</td>
<td>90187043</td>
<td>1184</td>
<td>90200539</td>
<td>-1.314</td>
<td>-0.015</td>
</tr>
<tr>
<td>14</td>
<td>A-n4-k7</td>
<td>940</td>
<td>71719200</td>
<td>941</td>
<td>68316239</td>
<td>0.103</td>
<td>4.745</td>
</tr>
<tr>
<td>15</td>
<td>A-n4-k7</td>
<td>1115</td>
<td>87453575</td>
<td>1114</td>
<td>83992772</td>
<td>-0.083</td>
<td>3.957</td>
</tr>
<tr>
<td>16</td>
<td>A-n5-k7</td>
<td>1093</td>
<td>83912212</td>
<td>1098</td>
<td>83050640</td>
<td>0.481</td>
<td>1.027</td>
</tr>
<tr>
<td>17</td>
<td>A-n5-k7</td>
<td>1202</td>
<td>91939271</td>
<td>1215</td>
<td>91142860</td>
<td>1.154</td>
<td>0.866</td>
</tr>
<tr>
<td>18</td>
<td>A-n5-k9</td>
<td>1111</td>
<td>85974668</td>
<td>1117</td>
<td>84382393</td>
<td>0.617</td>
<td>1.853</td>
</tr>
<tr>
<td>19</td>
<td>A-n6-k9</td>
<td>1377</td>
<td>106551382</td>
<td>1414</td>
<td>106064857</td>
<td>2.638</td>
<td>0.457</td>
</tr>
<tr>
<td>20</td>
<td>A-n6-k9</td>
<td>1103</td>
<td>86160883</td>
<td>1065</td>
<td>80980891</td>
<td>-3.449</td>
<td>6.012</td>
</tr>
<tr>
<td>21</td>
<td>A-n6-k8</td>
<td>1353</td>
<td>102138456</td>
<td>1377</td>
<td>102736651</td>
<td>1.755</td>
<td>-0.586</td>
</tr>
<tr>
<td>22</td>
<td>A-n6-k9</td>
<td>1691</td>
<td>127646413</td>
<td>1691</td>
<td>126143993</td>
<td>0.024</td>
<td>1.176</td>
</tr>
<tr>
<td>23</td>
<td>A-n6-k10</td>
<td>1359</td>
<td>103647386</td>
<td>1371</td>
<td>102973507</td>
<td>0.915</td>
<td>0.666</td>
</tr>
<tr>
<td>24</td>
<td>A-n6-k9</td>
<td>1482</td>
<td>112580339</td>
<td>1472</td>
<td>108916746</td>
<td>-0.664</td>
<td>3.254</td>
</tr>
<tr>
<td>25</td>
<td>A-n6-k9</td>
<td>1243</td>
<td>93806636</td>
<td>1281</td>
<td>96803299</td>
<td>3.041</td>
<td>-3.195</td>
</tr>
<tr>
<td>26</td>
<td>A-n69-k9</td>
<td>1209</td>
<td>91379350</td>
<td>1212</td>
<td>92060381</td>
<td>0.234</td>
<td>-0.745</td>
</tr>
<tr>
<td>27</td>
<td>A-n80-k10</td>
<td>1837</td>
<td>138307147</td>
<td>1822</td>
<td>13625641</td>
<td>-0.834</td>
<td>1.483</td>
</tr>
<tr>
<td>28</td>
<td>B-n57-k9</td>
<td>1656</td>
<td>127845698</td>
<td>1617</td>
<td>122815274</td>
<td>-2.378</td>
<td>3.934</td>
</tr>
<tr>
<td>29</td>
<td>B-n63-k10</td>
<td>1598</td>
<td>124880591</td>
<td>1565</td>
<td>117422748</td>
<td>-2.218</td>
<td>5.972</td>
</tr>
<tr>
<td>30</td>
<td>B-n64-k9</td>
<td>922</td>
<td>72030474</td>
<td>924</td>
<td>70978113</td>
<td>0.173</td>
<td>1.461</td>
</tr>
<tr>
<td>31</td>
<td>E-n76-k7</td>
<td>752</td>
<td>57858112</td>
<td>737</td>
<td>57538957</td>
<td>-1.959</td>
<td>0.552</td>
</tr>
<tr>
<td>32</td>
<td>E-n76-k10</td>
<td>909</td>
<td>69550717</td>
<td>915</td>
<td>68556681</td>
<td>0.670</td>
<td>1.428</td>
</tr>
<tr>
<td>33</td>
<td>E-n76-k14</td>
<td>1055</td>
<td>80864479</td>
<td>1100</td>
<td>83407569</td>
<td>4.311</td>
<td>-3.145</td>
</tr>
<tr>
<td>34</td>
<td>F-n4-k4</td>
<td>730</td>
<td>58226555</td>
<td>762</td>
<td>54251715</td>
<td>4.363</td>
<td>6.827</td>
</tr>
<tr>
<td>35</td>
<td>F-n72-k4</td>
<td>258</td>
<td>20781912</td>
<td>263</td>
<td>21340481</td>
<td>1.696</td>
<td>-2.688</td>
</tr>
<tr>
<td>36</td>
<td>F-n135-k7</td>
<td>1232</td>
<td>93214207</td>
<td>1215</td>
<td>91861982</td>
<td>-1.435</td>
<td>1.451</td>
</tr>
</tbody>
</table>

To show that the model also can handle large problems, a large set of customers (200–483), proposed by Golden et al. (1998), and very large-size benchmark problems (520–720 customers), introduced by Li et al. (2005), are selected for testing. The benchmark problems are solved by the two algorithms. All of these instances can be downloaded from the website http://users.ntua.gr/ezach. The results and computation times are presented in Table 5.2.
TABLE 5.2 RESULTS OF BENCHMARK PROBLEMS FROM LI ET AL.

<table>
<thead>
<tr>
<th>No.</th>
<th>n</th>
<th>V</th>
<th>CW</th>
<th>EECW</th>
<th>Extra Distance Traveled by Applying ECW (%)</th>
<th>Energy Saved by Applying ECW (%)</th>
<th>Computation Time of CW (minutes)</th>
<th>Computation Time of EECW (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>400</td>
<td>900</td>
<td>12762</td>
<td>3075936000</td>
<td>-2.00</td>
<td>2.52</td>
<td>17.46</td>
<td>16.97</td>
</tr>
<tr>
<td>2</td>
<td>480</td>
<td>1000</td>
<td>17028</td>
<td>4075108520</td>
<td>-5.40</td>
<td>5.94</td>
<td>34.98</td>
<td>35.09</td>
</tr>
<tr>
<td>3</td>
<td>280</td>
<td>900</td>
<td>10876</td>
<td>2659850370</td>
<td>-8.88</td>
<td>5.71</td>
<td>11.16</td>
<td>11.09</td>
</tr>
<tr>
<td>4</td>
<td>360</td>
<td>900</td>
<td>11976</td>
<td>2910572010</td>
<td>-4.76</td>
<td>5.47</td>
<td>25.83</td>
<td>26.71</td>
</tr>
<tr>
<td>5</td>
<td>255</td>
<td>1000</td>
<td>930</td>
<td>130666020</td>
<td>-4.88</td>
<td>0.85</td>
<td>2.58</td>
<td>2.57</td>
</tr>
<tr>
<td>6</td>
<td>323</td>
<td>1000</td>
<td>814</td>
<td>196050880</td>
<td>-1.41</td>
<td>-1.84</td>
<td>6.95</td>
<td>6.93</td>
</tr>
<tr>
<td>7</td>
<td>399</td>
<td>1000</td>
<td>1020</td>
<td>245200280</td>
<td>-1.72</td>
<td>1.35</td>
<td>17.95</td>
<td>17.90</td>
</tr>
<tr>
<td>8</td>
<td>483</td>
<td>1000</td>
<td>1239</td>
<td>298222320</td>
<td>0.11</td>
<td>-0.47</td>
<td>38.03</td>
<td>37.73</td>
</tr>
<tr>
<td>9</td>
<td>252</td>
<td>1000</td>
<td>948</td>
<td>230450790</td>
<td>-2.57</td>
<td>2.28</td>
<td>2.11</td>
<td>2.10</td>
</tr>
<tr>
<td>10</td>
<td>300</td>
<td>200</td>
<td>1077</td>
<td>260497440</td>
<td>0.09</td>
<td>1.14</td>
<td>5.04</td>
<td>4.88</td>
</tr>
<tr>
<td>11</td>
<td>360</td>
<td>200</td>
<td>1460</td>
<td>352737820</td>
<td>0.01</td>
<td>1.44</td>
<td>11.77</td>
<td>10.82</td>
</tr>
<tr>
<td>12</td>
<td>420</td>
<td>200</td>
<td>1958</td>
<td>469567660</td>
<td>-0.06</td>
<td>0.89</td>
<td>21.56</td>
<td>22.13</td>
</tr>
<tr>
<td>13</td>
<td>560</td>
<td>1200</td>
<td>19192</td>
<td>4600314520</td>
<td>-0.01</td>
<td>0.47</td>
<td>66.02</td>
<td>65.15</td>
</tr>
<tr>
<td>14</td>
<td>640</td>
<td>1400</td>
<td>23899</td>
<td>5732457420</td>
<td>-3.32</td>
<td>1.89</td>
<td>109.28</td>
<td>109.27</td>
</tr>
<tr>
<td>15</td>
<td>720</td>
<td>1500</td>
<td>27536</td>
<td>6607599480</td>
<td>-5.48</td>
<td>6.23</td>
<td>178.61</td>
<td>178.12</td>
</tr>
</tbody>
</table>

Among these ten case studies, two cases occur where the CW algorithm outperforms the EECW algorithm in terms of energy consumption. When the number of nodes increases, the computation times of the two methods also increase, as expected. Figure 5.2 shows the fitted line over the 36 benchmark problems computation time using the CW and ECW algorithms. As can be seen, the difference between computation times for the two algorithms when the number of nodes is less than 100 is not significant. However, for problems with a higher number of nodes, the computation time is faster. For instance, when the number of nodes is 262, the ECW algorithm is about 6% faster than the CW algorithm, while the quality of the result is almost the same. Thus, the ECW algorithm can expedite the process of solving a VRP problem.

![Figure 5.2: Comparison of CW and ECW computation times](image)
5.6 Conclusions

This chapter presents a new heuristic method (EECW) based on the traditional CW savings method, which approaches the LRP from an energy-efficiency perspective. Many factors contribute to the amount of energy used by a vehicle. Among those, this chapter considers vehicle weight and distance traveled as the major contributing factors. Here, the CW savings calculation is revised to take into consideration vehicles’ and customers’ demands weights. The CW algorithm is insensitive to route direction. However, the EECW algorithm is sensitive to direction, and hence, its algorithm is designed in such a way as to consider route direction. The proposed method is tested by several benchmark problems. By evaluating the case studies, it is perceived that applying the EECW algorithm results in better solutions in terms of energy and, in some cases, in terms of distance traveled.

As discussed in section 5.3, some parametric terms are added to the CW algorithm to boost its ability. Investigation of these parametric terms and adoption of them to increase the ability of the ECW algorithm is one of the future works under investigation. Also, many heuristics and meta-heuristics are developed for the LRP/VRP, with the objective of distance minimization in which energy minimization is a new window through which these heuristics/meta-heuristics can be approached differently.

5.7 References


CHAPTER 6
HEURISTIC ALGORITHM FOR BLOCK-TO-TRAIN ASSIGNMENT
AND TRAIN-ROUTING PROBLEMS

6.1 Abstract

The block-to-train assignment (BTA) and train-routing (TR) problem, together called the block-to-train-rout-ning problem (BTRP), involves finding the number of trains and their routes to transfer a set of blocks from their origins to their destinations. The objective of this problem is to minimize the total railway network cost. The BTRP is a very large-scale problem with millions of decision variables. Its size and mathematical difficulty preclude solving it using exact formulation techniques. In this chapter, a heuristic algorithm capable of solving the BTRP cost-effectively with a low computation time is developed. This algorithm is a two-phase heuristic. In phase I, the route construction phase, an initial solution is obtained. In phase II, a route improvement heuristic is applied to advance the solution to a feasible cost-effective solution. The algorithm solves problems with a variety of practical and business constraints. This heuristic has been applied to two sets of data provided by railroad industries through the INFORMS Railway Applications Section. Computation results show that the model can consider all cost elements and constraints imposed. It also provides results that are economical in terms of cost, especially relative to computation time.

6.2 Introduction

Railroads have been in existence for the past two centuries. The increasing cost of fuel has pushed industry to consider trains as an inexpensive alternative for fulfilling the transportation needs of their supply chain systems. With the growing use of trains in supply chain systems, train formation, which includes the allocation of railcars to trains and the efficient routing of trains, is becoming more important.

In rail transportation, “block” is referred to a set of railcars that are collected and moved as one group. The railcars that are assigned to a block usually have a compatible attribute, such as same origin or destination, but the railcars grouped in a block do not necessarily have the same origin and destination. The railcars can be grouped according to other shared contributes. “Block swap” occurs when a block is detached from one train and attached to another train in an intermediate point of its route, i.e., block swap allows a block to be assigned to more than one train while traveling on its path. Another concept in railroad problems is “crew segment.” Railroad staff who usually travel within a specific district or region are called a “crew segment” or “crew district.” A crew district is a path on
which a crew can operate a train. A train cannot start in the middle of a train crew segment. Train routing has
considerable flexibility since blocks can be dispatched or attached at any rail yard. These characteristics increase
the complexity of train scheduling in a supply chain system and make it a highly combinatorial problem. Several
attempts have been made by researchers in the past to find solutions for special cases of the problem. These
researchers have attempted to solve the block-to-train assignment (BTA) and train-routing (TR) problem together,
which is called the block-to-train-routing problem (BTRP).

A railroad network is a set of nodes and links in which nodes represent the physical locations of rail yards,
and links are rail tracks that connect nodes in the network. Upon receiving orders from customers, railcars are
grouped into blocks, and a trip plan is generated for each block. Four main issues must be addressed for generating
a trip plan:

- What cars should be grouped in a block?
- How many trains are required?
- What is the best block-to-train assignment?
- What is the best routing plan for each train (BTRP)?

The first question is referred to in the literature as the “blocking problem.” In this research, it is assumed
that blocks are formed in advance; hence, this chapter investigates the last three questions. Much work in the
literature addresses different aspects of the trip plan problem. Assad (1980) provides a literature survey on the
existing models for rail transportation. He categorized modeling efforts in rail transportation and investigated the
relationships between rail-related models and other transportation models. Keaton (1989 and 1992) provides mixed-
integer programming (MIP) for the rail freight transportation problem. In this MIP, the objective is to minimize
total cost, which is the summation of train-startup cost, car-time cost, and yard-classification cost. The problem
constraints are the train capacity and yard volume. The solution of the MIP provides service frequency, assignment
of railcars to predefined train routes, and grouping of railcars. Keaton suggests a Lagrangian relaxation algorithm
for the MIP when the train capacity constraint is relaxed. However, for solving the problem with a train size
limitation, he recommends using heuristics. Crainic and Rosseau (1986) present a network optimization model for a
multi-mode freight transportation problem which is solved by applying decomposition and column generation
methods. The model is a shortest-path algorithm in which capacity constraints are added to the objective function
using a penalty term. By using this model, they determine the best demand flow for a given train schedule. Then, a
new train schedule is obtained by using that demand flow information. Haghani (1987) provides a survey for the optimization models applied to a block-to-train assignment and train-routing problems. He also introduces areas for potential improvement in the rail transportation literature. Haghani (1989) considers empty railcars as one of his cost elements, which have been ignored by other researchers. He presents a formulation and a heuristic decomposition technique to solve the problem. As discussed by Gorman (1998), Keaton (1989 and 1992), Carinic and Rosseau (1986), and Haghani (1887 and 1989) obtain only the trains’ frequency of departure for a day, which is a not a sufficient basis for creating a train schedule. Newman and Yano (2001) propose an integer programming approach for intermodal containers. Intermodal containers can be carried either by trains or trucks. The objective for their problem is to minimize operating costs, including fixed train cost, variable transportation cost, each container handling cost, and yard inventory cost. The main constraint of the problem is the on-time delivery requirement. They present a heuristic decomposition procedure to find cost-effective solutions. Carpara et al. (2001) propose a graph theoretic formulation for the train time-tabling problem, which determines the departure time from the origin and arrival time at the destination as well as intermediate stops for each train. They used Lagrangian relaxation procedure to solve the linear integer programming presented. The relaxation is then embedded within a heuristic algorithm. Dorfman and Medanic (2004) provide a discrete event model scheduling for a train time-tabling problem. They enhanced the model using a capacity check algorithm to prevent deadlock. Jha et al. (2008) provide both an arc-based formulation and a path-based formulation for the block-to-train assignment problem. They propose exact and heuristic algorithms based on the path-based formulation. This heuristic applies a greedy construction algorithm to find cost-effective solutions.

A train-outing problem has several cost elements several of which are considered in this chapter and listed below:

- Train startup cost: Product of total number of trains dispatched and each train startup cost.
- Train travel cost: Product of total distance traveled by trains and train travel cost per unit of distance.
- Car travel cost: Product of total distance traveled by cars and car travel cost per unit of distance.
- Work event cost: Product of total number of work events and work event cost.
- Block swap cost: Summation of swap costs for blocks that are swapped.
- Crew imbalance cost: Product of number of crew imbalances in all crew segments and crew imbalance cost.
• Train imbalance cost: Product of number of train imbalances in each node and train imbalance cost.

• Missed block cost: Product of total number of blocks that are not carried to their destinations and missed block cost.

None of the papers in the literature considers all of these costs simultaneously. For example, Keaton (1989 and 1992) considers only train-startup cost, car-time cost, and yard-classification cost. Newman and Yano (2001) consider trains’ fixed cost, variable transportation cost, handling cost for each container, and yard-inventory cost, and they do not consider important elements such as block-swap cost, work event cost, train imbalance cost, and crew imbalance cost. Thus, a heuristic that is more comprehensive in the consideration of the cost elements is needed. In addition, since these existing models do not consider all cost elements, any associated constraints also are not considered in the formulation. When considering all cost elements and their associated constraint terms, the problem becomes more complex in terms of the computation time necessary for solving. Hence, the objective of this chapter is to develop a heuristic approach to minimize the summation of all cost elements explained above while imposing the following constraints:

• Number of blocks assigned to a train: Each train has a restriction on the number of blocks that it can carry. Assigning too many blocks to a train will cause more intermediate stops, which is not efficient.

• Number of block swaps: A block is limited to a few numbers of swaps. Each swap requires additional time and resources. Therefore, a greater number of block swaps may result in network inefficiency.

• Number of intermediate stops (work events) per train: When a train stops at an intermediate point, it is usually for dropping off or picking up a block. Each intermediate stop is called a work event and incurs cost to the system. Therefore, work events are restricted to a few in order to avoid unpractical solutions. It is also important to note that picking up or dropping off a block at the origin or the destination of a train is not considered a work event.

• Train length and weight on each link: Each link in the system has a restriction on the maximum length and weight of the trains that it can handle.

• Number of trains on each link: Each link cannot handle more than a limited number of trains in each direction. This restriction is imposed to avoid congestion on the links.

• Crew starting and terminating yards: In a railway network, a crew can travel only on the predefined crew segment. Each train must be assigned to a crew on each crew segment. Thus, a train might pass many
crew segments during its trip. However, the train must begin at the starting point of a crew segment and finish at the ending point of a crew segment, even if there is no block to be carried to the end of the crew segment.

At first glance, the BTRP may seem similar to the vehicle-routing problem (VPR). But there are fundamental differences between the two:

- In the BTRP, the origin and destination of trains is not an input to the problem.
- Each block has a different origin and destination.
- The number of times a train can visit a node is not limited.

These fundamental differences make the BTRP problem more complex than a regular VRP. However, similar to the VRP, the heuristics for a BTRP can be designed for two purposes: route construction and route improvement. In route construction methods, the heuristic is used to construct an initial solution, and the results can be further improved to near optimality by applying a route improvement algorithm(s). In fact, a route improvement algorithm requires an initial solution, which can be obtained using a route construction method. Thus, these two methods work in series (Figure 6.1).

Figure 6.1. Sequence of route construction and route improvement steps to provide near-optimal results

The formulation of the problem while considering all of the cost elements and constraints is tedious. The dimensions that contribute to the complexity of the formulation and the computation time are as follows: number of nodes, number of blocks, number of block swaps, number of links, and number of crew segments. Even if an exact formulation is attempted, the computation time for obtaining the solution will be extremely large. All previous researchers who attempted the BTRP formulation have solved it using heuristics, especially when the problem is large. The reason for choosing a heuristic approach instead of an exact formulation is to reduce the computation effort required for obtaining a solution.

6.3 Heuristic Algorithm for Block-to-Train Assignment and Train-Routing Problem

The following notations are used in the explanation of the heuristic algorithm:

- \( N \) Number of nodes in the system
- \( I \) Set of nodes in the system
The heuristic for solving the BTRP has two phases. The first phase is a route-construction phase during which time an initial solution set is generated with some relaxed constraints. In the second, route-improvement phase, the result obtained from the first phase is consolidated to provide a feasible and cost-effective alternative.

In Phase I, the heuristic initiates a train for each block in the system. Therefore, there are \( B \) trains in the initial solution. Each train starts from the origin of the block to which the train is assigned and terminates at the block destination. Thus, in Phase I, to find the route for each train, the most cost-minimizing route is the shortest path from the start of each block to the destination of each block. In the initial solution, there are as many trains as there are number of blocks. Hence, to ensure the shortest route for each block, the constraint on the number of trains passing through a link is relaxed. Thus, the solution generated in this phase may be infeasible, but the infeasibility is repaired in the second phase (route-improvement heuristic).

The train assigned to a block starts from the block origin and travels to the block destination through the shortest path between the two nodes. For the calculation of the shortest path, the Dijkstra algorithm is used according to the following steps (Larson and Odoni, 1981):

1- Assign a temporary distance to each node (zeros for the starting node and infinity for the rest of the nodes).

2- Mark the initial node as current and all other nodes as unvisited nodes, and put them in the list of unvisited nodes.

3- Calculate the distance from the current node to all of its neighbor nodes. If the distance is less than the last reordered distance, then update the distance. If a node is examined but not updated to current, keep it in the list of unvisited nodes.
4. When all neighbors of the current node are considered, mark the node as visited and delete it from the unvisited set. Once a node is visited, it will not be examined again and the recorded distance is finalized and minimum.

5. Mark the next current node as the node with the minimum distance in the unvisited set.

6. If the unvisited set is exhausted, consider the algorithm complete; otherwise, consider the unvisited node marked with the smallest distance as the next current node, and repeat steps 3 through 6.

To implement the first step of the Dijkstra algorithm, it is necessary to calculate an adjacency matrix, called $A$. Matrix $A$ is an $N \times N$ matrix in which $a_{ij} = 1$, if there is a direct link between rail yards $i$ and $j$; otherwise, $a_{ij} = 0$. This matrix will also be used in the route-improvement heuristic.

As stated earlier, the initial solution obtained may not be feasible; however, the infeasibility is removed during the second step. The route-improvement heuristic is based on the similarity between each pair of routes in the initial solution obtained from phase I. The similarity between each pair of routes is calculated in terms of the links that the two routes share in the direct or reverse directions. For this purpose, two similarity measures are introduced: direct similarity and reverse similarity. The formulas for the calculation of direct similarity between train $i$'s and train $j$'s routes is presented in equations (6.1).

$$s_{ij} = \frac{\text{Number of links in route } i \text{ that exist in route } j}{\text{Total number of links in route } i}, \quad \forall i, j, i \neq j \quad (6.1)$$

The similarity index, $s_{ij}$, is a measure of similarity between the routes of trains $i$ and $j$. Since initially there are $B$ blocks in the system, and each is served by a train, the number of trains in the system is $B$. Therefore, the calculations presented in equation (6.2) can be presented in a $B \times B$ matrix, called $S$. It is necessary to mention that $S$ is not a symmetric matrix. This can be clarified by an example: if train $k$’s route is 1-2-4-5-7 and train $z$’s route is 9-1-2-4-5-7, then $s_{kz} = 1$, while $s_{zk} = 0.8$.

The reverse similarity index $r_{kj}$ is defined by equation (6.2):

$$r_{ij} = \frac{\text{Number of links in route } i \text{ that exist in the reverse of route } j}{\text{Total number of links in route } i}, \quad \forall i, j, i \neq j \quad (6.2)$$

The term $r_{kj}$ shows the similarity between route $k$ and the reverse of route $z$. For instance if $r_{kz} = 1$ and $r_{zk} = 1$, then $k$ is the reverse of $z$. Similar to $S$, $r_{kj}$ can be presented in a $B \times B$ matrix, called $R$, which is also not a symmetric matrix. Reverse similarity plays an important role in avoiding train imbalance and crew imbalance cost. Train imbalance
cost is the difference between the number of trains entering to and exiting from a node, and crew imbalance is the difference between the number of trains traveling through links \( i-j \) and \( j-i \).

After calculating the similarity matrices, each pair of routes is examined to determine whether they can be fully or partially combined. The merger can happen due to direct or reverse similarity of the two routes, which will be explained in detail in the steps of the proposed heuristic. The merger will continue until no further simplification or merger is possible without violating the existing constraints in the system.

During the merging process, the number of trains passing through a link decreases and moves toward a feasible status while all other constraints are enforced. The constraints considered in this problem are the number of blocks assigned to a train, number of block swaps, work event per train, train length and tonnage, number of trains passing through a link, and crew segments, as explained in the introduction.

As previously mentioned, the criteria for combining a pair of routes is their similarity. The conditions that two routes can have with respect to each other can be expressed using their similarity indices. Table 6.1 shows the interpretation of route \( i \) and \( j \) conditions with respect to each other in terms of the similarity indices as well as an example for each condition.

<table>
<thead>
<tr>
<th>Flag</th>
<th>Condition</th>
<th>Description</th>
<th>Figure Example</th>
<th>Route Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( s_{ij} = s_{ji} = 1 )</td>
<td>Route ( i ) is equal to route ( j )</td>
<td></td>
<td>Route 1: 1-5-7-8-9 Route 2: 1-5-7-8-9</td>
</tr>
<tr>
<td>2</td>
<td>( s_{ij} = 1, s_{ji} \neq 1 )</td>
<td>Route ( i ) is sub-route of ( j )</td>
<td></td>
<td>Route 1: 1-5-7-8-9 Route 2: 1-5-7-8-9</td>
</tr>
<tr>
<td>3</td>
<td>( r_{ij} = r_{ji} = 1 )</td>
<td>Route ( i ) is reverse of route ( j )</td>
<td></td>
<td>Route 1: 1-5-7-8-9 Route 2: 1-5-7-8-9</td>
</tr>
<tr>
<td>4</td>
<td>( r_{ij} = 1, r_{ji} \neq 1 )</td>
<td>Route ( i ) is reverse of sub-route of ( j )</td>
<td></td>
<td>Route 1: 1-5-7-8-9 Route 2: 1-5-7-8-9</td>
</tr>
<tr>
<td>5</td>
<td>( 0 &lt; s_{ij}, s_{ji} &lt; 1 )</td>
<td>Sub-route of route ( i ) is sub-route of ( j )</td>
<td></td>
<td>Route 1: 1-5-7-8-9 Route 2: 1-5-7-8-9</td>
</tr>
<tr>
<td>6</td>
<td>( 0 &lt; r_{ij}, r_{ji} &lt; 1 )</td>
<td>Sub-route of route ( i ) is reverse of sub-route of ( j )</td>
<td></td>
<td>Route 1: 1-5-7-8-9 Route 2: 1-5-7-8-9</td>
</tr>
</tbody>
</table>

These conditions and matrices are used in the structure of the following heuristic algorithm:

0- Start.
1- Standardize nodes: Rename nodes from \((1,2,…,N)\), and rename links, blocks’ origin, and destination accordingly, if their names are linguistics or not cardinals.
2- Calculate shortest path and its related distance for each block using Dijkstra algorithm.
3- Check summation of all blocks’ weights and lengths on each link, to see whether there is a link in which the summation of the blocks’ weight is greater than $T_{ij} \times N_{ij}$, or the summation of blocks’ length is greater than $L_{ij} \times N_{ij}$.

- If yes, then find blocks passing through that link, and among those, find block that has shortest path. Make $A_{ij} = 0$, and recalculate shortest path for that route (block). Repeat step 3.
- If no, then go to next step.

4- Calculate $R$ and $S$.

5- Find all routes $i$ and $j$ with condition 1; check constraints.

- If constraints are not violated, then delete route $j$, put route $j$ in the list of deleted routes, and update $S$ as $s_{ji} = 0, \forall i$ and $s_{ij} = 0$. Repeat step until no route pair has condition 1.
- If at least one constraint is violated for a route pair, then do not delete any routes, and update $S$ as $s_{ij} = 0$ and $s_{ji} = 0$. Repeat step until no route pair has condition 1.

6- Find all routes $i$ and $j$ with condition 2. Sort these route pairs in descending order. Starting with the highest $s_{ji}$, check constraints.

- If constraints are not violated, then delete route $i$. Put route $i$ in the list of deleted routes, and update $S$ as $s_{ij} = 0, \forall j$ and $s_{ji} = 0$. Repeat step until no route pair has condition 2.
- If at least one constraint is violated, then update $S$ as $s_{ij} = 0$ and $s_{ji} = 0$. Repeat step until no route pair has condition 2.

7- Find all routes $i$ and $j$ with condition 3, and check constraints.

- If constraints are not violated, then add route $j$ to route $i$, and delete route $j$. Put route $j$ in list of deleted routes, and update $R$ as $r_{ji} = 0, \forall i$ and $r_{ij} = 0$. Repeat step until no pair has condition 3.
- If at least one constraint is violated, then update $R$ as $r_{ij} = 0$ and $r_{ji} = 0$. Repeat step until no pair has condition 3.

8- Find all routes $i$ and $j$ with condition 4. Sort these route pairs in descending order. Starting with highest $s_{ji}$, check constraints.
If constraints are not violated, then add route \( i \) to route \( j \), and delete route \( i \). Put route \( i \) in list of deleted routes, and update \( r \) as \( r_{ij} = 0, \forall j \) and \( r_{ji} = 0 \). Repeat step until no route pair has condition 4.

If at least one constraint is violated, then update \( R \) as \( r_{ij} = 0 \) and \( r_{ji} = 0 \). Repeat step until no route pair has condition 4.

9- Find all routes \( i \) and \( j \) with condition 5. Sort these route pairs in descending order. Starting with route pair with highest \( s_{ji} \) and \( s_{ij} \), check constraints.

- If constraints are not violated, then delete the part of route \( j \) that is similar to route \( i \). Update route \( j \). Add swap cost for nodes at beginning and end of shared route, if they are not origin and destination of route \( i \). Update \( r \) as \( s_{ji} = 0, \forall i \) and \( s_{ij} = 0 \). Repeat step until no route pair has condition 5.

- If at least one constraint is violated, then update \( S \) as \( s_{ij} = 0 \) and \( s_{ji} = 0 \). Repeat step until no route pair has condition 5.

10- Find all route pairs with condition 6. Sort these route pairs in descending order. Starting with route pair with highest \( r_{ji} \) and \( r_{ij} \), check constraints.

- If constraints are not violated, then delete part of route \( j \) that is similar to route \( i \). Update route \( j \). Add swap cost for nodes at beginning and end of shared route, if they are not origin and destination of route \( i \). Update \( R \) as \( r_{ji} = 0, \forall i \) and \( r_{ij} = 0 \). Repeat step until no route pair has condition 6.

- If at least one constraint is violated, then update \( R \) as \( r_{ij} = 0 \) and \( r_{ji} = 0 \). Repeat step until no route pair has condition 6.

11- Revise nodes that start from middle of crew segment.

12- All routes are finalized in requested format and heuristic stops.

Steps 1 to 3 represent the route-construction phase of the heuristic. Steps 4 to 11 represent the route-improvement phase of the heuristic. It is important to mention that each of the conditions in steps 8 and 9 have two other sub-conditions which are also considered in the algorithm. If route \( i \) is a sub-route of \( j \), then \( i \) and \( j \) can have the following conditions with respect to each other:

1. \( i \) and \( j \) have the same origins
   a. destination of \( i \) and \( j \) are directly connected
2. i and j have the same destinations
   a. destination of i and j are directly connected
   b. destination of i and j are not directly connected

In case 1a, route i can be eliminated, and train j can finish its route at route i’s destination since the destinations of i and j are connected. Similarly, in case 2a, route i can be eliminated, and train j can start its route from the origin of route i, because i’s and j’s origins are directly connected, and i is a sub-route of j.

It must be noted that reordering steps 5 to 10 can provide a different result. Since the computation time is small, it is possible to run the model for all possible sequences and choose the one that provides the minimum cost for the system.

This algorithm is coded in MATLAB 7.10.0(R2010a) and is implemented on an Intel® Core™ 2 Quad CPU with 32-bit operating system. The algorithm is tested on two sets of data provided by industry. A summary of input values for the first and second case studies are presented in Tables 6.2 and 6.3. More information regarding the case studies can be obtained from the INFORMS Railway Application Section Website: [http://www.informs.org/Community/RAS/Problem-Solving-Competition/2011-RAS-Problem-Solving-Competition](http://www.informs.org/Community/RAS/Problem-Solving-Competition/2011-RAS-Problem-Solving-Competition).

Tables 6.3 shows the output summaries from the first set of data obtained from the aforementioned source. The computation time for the first data set is 7.82 seconds. For this problem, the number of required trains is 81. This number can be further reduced by using the newly formed routes as input to the improvement phase of the heuristic. After three iterations of this step, the number of routes is reduced to 69. Table 6.4 shows the output summaries from the first second set of data. For the second data set, the computation time is 26.83 seconds, and the number of trains required is 100. The details of train scheduling and block-to-train assignments are available for interested researchers upon request. The total cost of the system is $2,133,673 for the first data set and $2,908,105 for the second data set.
### TABLE 6.2 INPUT SUMMARY of FIRST CASE STUDY

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>94</td>
</tr>
<tr>
<td>Number of Links</td>
<td>134</td>
</tr>
<tr>
<td>Number of Blocks</td>
<td>239</td>
</tr>
<tr>
<td>Maximum Number of Trains per Link</td>
<td>6</td>
</tr>
<tr>
<td>Crew Imbalance Penalty per Imbalance ($)</td>
<td>600</td>
</tr>
<tr>
<td>Train Imbalance Penalty per Imbalance ($)</td>
<td>1000</td>
</tr>
<tr>
<td>Train Travel Cost per Mile</td>
<td>10</td>
</tr>
<tr>
<td>Car Travel Cost per Mile ($)</td>
<td>0.75</td>
</tr>
<tr>
<td>Cost per Work Event ($)</td>
<td>350</td>
</tr>
<tr>
<td>Maximum Blocks per Train</td>
<td>8</td>
</tr>
<tr>
<td>Maximum Block Swaps per Block</td>
<td>3</td>
</tr>
<tr>
<td>Maximum Intermediate Work Events per Train</td>
<td>4</td>
</tr>
<tr>
<td>Train Start Cost ($)</td>
<td>400</td>
</tr>
<tr>
<td>Missed Cost per Railcar ($)</td>
<td>5000</td>
</tr>
</tbody>
</table>

### TABLE 6.3 INPUT SUMMARY of SECOND CASE STUDY

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>221</td>
</tr>
<tr>
<td>Number of Links</td>
<td>154</td>
</tr>
<tr>
<td>Number of Blocks</td>
<td>369</td>
</tr>
<tr>
<td>Maximum Number of Trains per Link</td>
<td>10</td>
</tr>
<tr>
<td>Crew Imbalance Penalty per Imbalance ($)</td>
<td>600</td>
</tr>
<tr>
<td>Train Imbalance Penalty per Imbalance ($)</td>
<td>1000</td>
</tr>
<tr>
<td>Train Travel Cost per Mile</td>
<td>10</td>
</tr>
<tr>
<td>Car Travel Cost per Mile ($)</td>
<td>0.75</td>
</tr>
<tr>
<td>Cost per Work Event ($)</td>
<td>350</td>
</tr>
<tr>
<td>Maximum Blocks per Train</td>
<td>8</td>
</tr>
<tr>
<td>Maximum Block Swaps per Block</td>
<td>3</td>
</tr>
<tr>
<td>Maximum Intermediate Work Events per Train</td>
<td>4</td>
</tr>
<tr>
<td>Train Start Cost ($)</td>
<td>400</td>
</tr>
<tr>
<td>Missed Cost per Railcar ($)</td>
<td>5000</td>
</tr>
</tbody>
</table>

### TABLE 6.4 OUTPUT SUMMARY of FIRST CASE STUDY

<table>
<thead>
<tr>
<th>Objective Function Component</th>
<th>Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train Start Cost</td>
<td>32,400.00</td>
</tr>
<tr>
<td>Total Train Travel Cost</td>
<td>373,254.00</td>
</tr>
<tr>
<td>Total Car Travel Cost</td>
<td>1,556,658.98</td>
</tr>
<tr>
<td>Work Event Cost</td>
<td>46,200.00</td>
</tr>
<tr>
<td>Block Swap Cost</td>
<td>560.00</td>
</tr>
<tr>
<td>Crew Imbalance Cost</td>
<td>54,600.00</td>
</tr>
<tr>
<td>Train Imbalance Cost</td>
<td>70,000.00</td>
</tr>
<tr>
<td>Missed Cars Cost</td>
<td>0</td>
</tr>
<tr>
<td>Total Cost</td>
<td>2,133,672.98</td>
</tr>
</tbody>
</table>
### TABLE 6.5 OUTPUT SUMMARY OF SECOND CASE STUDY

<table>
<thead>
<tr>
<th>Objective Function Component</th>
<th>Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train Start Cost</td>
<td>40,000</td>
</tr>
<tr>
<td>Total Train Travel Cost</td>
<td>496,012</td>
</tr>
<tr>
<td>Total Car Travel Cost</td>
<td>2,184,243.3</td>
</tr>
<tr>
<td>Work Event Cost</td>
<td>79,100</td>
</tr>
<tr>
<td>Block Swap Cost</td>
<td>1,150</td>
</tr>
<tr>
<td>Crew Imbalance Cost</td>
<td>33,600</td>
</tr>
<tr>
<td>Train Imbalance Cost</td>
<td>74,000</td>
</tr>
<tr>
<td>Missed Cars Cost</td>
<td>0</td>
</tr>
<tr>
<td>Total Cost</td>
<td>2,908,105</td>
</tr>
</tbody>
</table>

### 6.4 Conclusions

In this chapter, a two-phase heuristic method for the BTRP is presented. In the first phase, a route construction heuristic is proposed using the Dijkstra algorithm to generate an initial solution. The initial solution may be infeasible, because in phase I, the constraints are relaxed. In the second phase, the solution is iteratively improved. In the second phase, all constraints are imposed to ensure feasibility. The improvement phase merges train routes created in the first phase by using the similarity that exists between routes. In this process, the initial solution becomes feasible and cost-effective. The heuristic is computationally efficient and considers many business constraints simultaneously to provide the result. Two real-life case studies were investigated. The result for the first case study, which has 94 nodes and 239 blocks, is obtained in 7.82 seconds. The computation time for the second case study, in which there are 221 nodes and 369 blocks, is 26.83 seconds. These small computation times justifies the time efficiency of the proposed algorithm.

Combining a blocking algorithm with the heuristics proposed in this work for the BTRP can result in a more economical solution. However, it would impact the computation time adversely. Thus, investigating the impact of the integration between the proposed BTRP and a blocking problem on solution quality and time can be of great value.

### 6.5 References


CHAPTER 7

SUMMARY AND CONCLUSIONS

7.1 Summary and Conclusions

In this chapter, the important findings of this dissertation are summarized and possible future works are discussed. Chapter 2 presented a formulation for LRP with dynamic demand, which minimizes the total network cost by finding the best network configuration. The network configuration includes finding the best strategy with respect to location, allocation, and routing plan. The proposed formulation considers the dynamic demand and eliminates the waiting times in providing effective solutions. Since the VRPTW is a special case of the TDLRP, the proposed model for the TDLRP was validated by solving VRPTW examples. Also, remedies for the infeasibility, which may occur as the result of dynamic demands were discussed. The presented mixed-integer non-linear formulation was linearized, and different cutting approaches were applied to solve the linear model. Results showed that when customers’ demands are linear, the moderate cutting method strategy provides results faster.

Chapter 3 presented an exact formulation for the LRP in which the time taken to travel between each pair of nodes is considered dynamic. The problem formulation eliminates the waiting times at customer locations and also solves the problem of different scenarios, such as no time windows, hard and soft time windows, and time-dependent demand. The presented mixed-integer non-linear model was linearized and solved using CPLEX. The branch-and-bound approach and other cutting methods were used for solving the model. Results indicated that the pure branch-and-bound algorithm provides results faster than cutting approaches for small-size problems.

Chapter 4 presented an exact formulation for the LRP in which the objective function is to minimize the energy, emission, and depot-establishment cost while imposing common LRP constraints. For calculating energy consumption, it was assumed that the vehicles travel at a constant speed on flat ground. With this condition, forces that contribute to the energy consumption of vehicles are rolling resistance and aerodynamic drag. The proposed formulation is the first formulation of the LRP in the literature that contributes these forces together in the objective function of the LRP. It is also the first formulation that differentiates among vehicles, not only in term of capacity but also in terms of sources of energy and aerodynamic characteristics. The proposed model can also handle hard time window, soft time window, and dynamic demands. The examples solved showed that a significant amount of savings can be obtained by applying the proposed formulation.
The computational complexity of the formulation presented in Chapter 4, especially when the problem size is large, has forced the development of heuristics for solving large-size problems. Hence, Chapter 5 presents a heuristics algorithm for the VRP in which the objective function is energy minimization. The presented heuristic, which is a modified version of the Clark-Wright savings algorithm, was tested by customized benchmark problems from the literature. Analysis of the results obtained from traditional and proposed savings showed that the proposed energy-efficient Clark-Wright algorithm can result in a significant amount of energy savings.

Chapter 6 presents a heuristic algorithm that is capable of solving the block-to-train assignment and train-routing problems. This heuristic considers many business cost elements and constraints, which have not been investigated simultaneously in an integrated model or algorithm in the literature. The heuristic is a two-phase algorithm that generates an initial solution in phase I and improves the solution in phase II. The algorithm proposed was used to solve two real-life problems obtained from industry. Results showed that the proposed heuristics is capable of proving the solutions, which are time and cost-effective.

7.2 Future Work

Chapters 2 and 3 of this dissertation were devoted to developing formulations for the LRP with the existence of dynamic parameters—demand and travel times. To linearize the formulations presented in these chapters, demand and travel time functions were assumed to be linear and discrete step functions, respectively. Different types of demand or travel time functions can yield different linearization approaches. Thus, investigating other types of demand/travel time functions and studying their impact on solution quality and time could be interesting topics for future research. In addition, since the majority of constraints in the model can be separated in terms of vehicles, practicing exact heuristics, such as column generation or Benders decomposition (row generation), can enable the models to solve large-size problems. In addition, developing exact heuristics, heuristics, or meta-heuristics for solving the formulations presented in Chapters 2, 3, and 4 can be of great value.

It is expected that the heuristic algorithm developed in Chapter 5 for the energy efficient VRP could be further enhanced by adding some parametric terms described in that same chapter. Further work to test the use of these parametric terms and study their impact on the result could be of a great value.

To study the accuracy of the model presented in Chapter 6 by developing other heuristics or exact formulations for the BTRP is another interesting subject of research. Also, combining a blocking algorithm with the heuristic proposed for the BTRP can result in a solution that is more economical. However, it would impact the
computation time adversely. Thus, investigating the impact of the integration between the proposed BTRP and a blocking problem on solution quality and time can be of great value.
APPENDIX
APPENDIX
TRAVEL-TIME FUNCTIONS

\[ F_{12}(t) = \begin{cases} 
(t-7)^3 + 8 & 0 < t < 12 \\
-t + 10 & 12 \leq t < 24
\end{cases} \]

\[ F_{13}(t) = \begin{cases} 
t^2 + 10 & 0 < t < 8.5 \\
-t^2 + 11.5 & 8.5 \leq t < 24
\end{cases} \]

\[ F_{14}(t) = \sin\left(\frac{t}{2}\right) + 2 \quad 0 < t < 24 \]

\[ F_{21}(t) = e^{10-t} + 10 \quad 0 < t < 24 \]

\[ F_{23}(t) = t^2 - \left(\frac{t}{3}\right)^3 + 10 \]

\[ F_{51}(t) = \cos\left(\frac{x}{3} - 50\right) + 5 \quad 0 < t < 24 \]

\[ F_{52}(t) = 1 \quad 0 < t < 24 \]

\[ F_{53}(t) = \left(\frac{t-5}{10}\right)^2 - \left(\frac{x}{10}\right)^4 + 35 \quad 0 < t < 24 \]

\[ F_{54}(t) = \sin\left(\frac{t}{2} - 6\right) + 5 \quad 0 < t < 24 \]