Optimization Models for Retail Reverse Supply Chains

Optimization Models For Retail Reverse Supply Chains

By Mehmet Erdem Coskun, M.A.Sc., B.Sc.

A Thesis Submitted to the School of Graduate Studies in the Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy

McMaster University, Hamilton, Ontario
Doctor of Philosophy (2022)
(Computational Science and Engineering)

TITLE: Optimization Models For Retail Reverse Supply Chains
AUTHOR: Mehmet Erdem Coskun, M.A.Sc., B.Sc. (McMaster University)
SUPERVISOR: Dr. Elkafi Hassini
NUMBER OF PAGES: xvii, 110

Lay Abstract

This thesis deals with Retail Reverse Supply Chain (RRSC) management. We consider an independent retail company's and its franchise stores' ineffective inventory which may be constituted of unsold, under-selling, slow-moving, customer-returned, end-of-life, end-of-use, damaged, and faulty products within their inventory. We take into account the retailer's reverse supply chain structure and investigate the following problems: 1) How to manage a store's product returns under a given budgetary limitation for financial planning and taxation reasons, due to lost income from returned items, 2) Inventory optimization by taking into account the reverse supply chain structure of the retailer, and 3) Providing insight to the retailer on how it can best re-negotiate its vendor (buy-back) contracts for its product returns. The thesis covers decision making in all three levels: day-to-day operational decisions such as which products to be returned and where to allocate them within its reverse supply chain options, mid-term tactical decisions such as which Return Centers (RC) to be activated for the Reverse Logistics (RL) activities, and long-term strategic decisions such as what should be the optimal contract terms to re-negotiate with the vendors in order to cut future return costs.

This thesis is dedicated to my family, Ismet - Adile - Emrah - Derya COSKUN, and my friends who supported me all the way through my entire life with their love and patience.

Abstract

Unlike most of the existing literature on reverse supply chains, that focuses on product recovery or waste management, in this thesis we consider reverse supply chain operations for an independent retailer. The latter have forward and reverse supply chains that are independent of the manufacturers. We study three major problems related to Retail Reverse Supply Chains (RRSC) for independent retailers. In RRSCs, each retail store holds some products that are not selling (and/or under-selling) and wishes to salvage them optimally. We refer to these products as Ineffective Inventory. Salvage can be in many forms and take place by relocating a product within the reverse supply chain (RSC), such as sending the product from a franchise store back to a Distribution/Return Center (RC) and then forward to another franchise store, or returning it to a vendor, liquidation, etc. The RRSC network may includes system members such as stores (retailer owned and/or franchise), RCs, warehouses, vendors and liquidators. Each of the stores carries some inventory that is underselling, and it is important to reduce the inventory of such products in order to refill the space with inventory that is more likely to sell.

In the first problem, we consider a basic RRSC with retail stores, vendors and a warehouse. The retail company allocates a budget for its RRSC activities. We refer to this budget as a Profit-Loss budget, due to lost income from the items that will be removed from the stores that was a part of the gains resulting from the previous year tax calculations. The objective is to use this Profit-Loss budgetary limitation as effectively as possible with the most suitable products to relocate products within the supply chain and/or return them back to their vendor. A heuristic algorithm is developed to solve this problem, by making use of the problem structure, and results are compared with the solutions of an exact state-of-the-art commercial solver.

In the second problem, we consider a network optimization model with inventory decisions. The goal is to optimize ineffective inventory levels in stores and the disposition of their returns. We model a comprehensive RRSC network with multiple stores that could be Company-Owned or Franchise Stores, multiple warehouses, multiple RCs, multiple vendors, and liquidators. The objective of the retailer is to minimize costs for relocating some of this ineffective inventory within the network or scrapping. However, individual franchise stores have their own goals of how their excessive inventory should be handled. The franchisee goals may be conflicting with those of the franchisor in terms of how much inventory should be chosen from each store to be relocated. In return, this conflict may lead to a conflict among franchise stores. This issue is addressed and resolved through

inventory transparency among all the supply chain members. The tactical decision making process of which RC should be used for handling returns is incorporated into the model. In order to overcome the complexities of the large size problem, a multi-stage heuristic is developed to solve this problem within reasonable times. The results are then compared with the solutions of state-of-the-art commercial solver.

In the third problem, we focus on the strategic decision of developing optimal vendor contract parameters for the retailer, using optimization models. Specifically, we identify optimal return penalties and associated return thresholds, between an independent retailer and its vendors. This model will support the retailer in their contract re-negotiation for its RSC activities. Vendors use a multi-layered penalty structure that assigns higher penalties to higher returns. The objective is to find the optimal penalties and/or optimal return thresholds that should be negotiated with the vendors in order to pay a lower penalty in the upcoming return cycles compared to existing penalty structures. We first design a Mixed Integer Non-Linear Program (MINLP) where the model makes the decision of vendor penalty fees and return thresholds simultaneously for each vendor. We generate small size to large size problems and solve them via MINLP solvers such as DICOPT and ANTIGONE. In order to gain insights to the inner workings of the MINLP, the decision variables, vendor penalty fees and return thresholds, are considered as parameters and hence, two models are designed to find the optimal penalty structure and optimal return thresholds, respectively. Useful insights from both of the models' solutions are derived in order to generate rule-of-thumb methodologies to find approximate solutions close to optimal penalty percentages and return thresholds via identifying all possible scenarios that can exist in the problem structure.

Acknowledgements

With lots of unknowns within the academic research world, and also in life, it is always easy to get lost. However, with the right direction from an amazing mentor, it is very easy to focus and get back on track to move forward. Therefore, the existence of this doctoral thesis would not be possible without the help, guidance, suggestions and support of my professor and mentor Dr. Elkafi Hassini. I am deeply indebted for the time and efforts he spent on my academic and personal growth as well as supporting my every step throughout the journey, with patience, to make this research a reality. I have always felt his sincere and continuous support and guidance which encouraged me to see the light at the end of the tunnel, especially when I needed it the most. I take this opportunity to show my sincere appreciation for encouraging me to accomplish my goals and thank him for giving me the opportunity to be his student. I had the upmost gratification to work with him and hope to collaborate again in the future.

I am very appreciative of my committee members Dr. Kai Huang and Dr. Vladimir Mahalec for the suggestions and direction they have provided throughout my research which helped me to focus on the important areas in my thesis. I would like to thank them for their valuable feedback and guidance which made me realize what matters in my research.

I also thank my parents, brother, wife and friends for their ongoing love and support.

Contents

List of Figures

List of Tables

Notation and Abbreviations

Sets:

P	set of Ineffective Products in the stores
S	set of all Stores ($CS \cup FS = S$)
CS	set of Company-owned Stores ($CS \subset S$)
FS	set of Franchisee-owned Stores ($FS \subset S$)
ST	set of Store-Types (ST = $\{CS,\ FS\}$)
V	set of all Vendors
W	sset of all Warehouses that accept returned inventory
D	sset of all the Distribution/Return Centers (RC) that participate in a reverse logistics (RL) activity
SC	sset of all Scenarios that can exist when certain RCs are active.

Indices:

p	products in the stores ($p = 1, 2, ..., \|P\|$)
s	stores ($s = 1, 2, ..., \|S\|$)
st	store-types ($st \in ST$)
v	vendors ($v = 1, 2, ..., \|V\|$)
w	warehouses that accept returned inventory ($w = 1, 2, ..., \|W\|$)
d	RCs that participate in a reverse logistics (RL) activity ($d = 1, 2, ..., \|D\|$)
sc	scenarios ($sc = 1, 2, ..., \|SC\|$)

Parameters, subsets and Abbreviations:

P_v	Set of Ineffective Products in the stores supplied by vendor v, $P = \cup_{v=1}^{\lvert V\rvert} P_v$ and $P_v \cap P_{v'} = \emptyset \ \ \forall\, v \neq v'$
R_p	Store refund rebate rate of product p,
COG_p	Cost of Good of product p
LQR_p	Liquidation Rebate Rate of product p
LC_p	Landed Cost of product p
SP_p	Store Purchase Price of product p from the retail company ($SP_p > LC_p$)
DV_p	Deposit Value of product $p, DV_p < COG_p$
VOL_p	Volume of product p (in cm^3)
WE_p	Weight of product p (in g)
RHF_d	Per unit Receiving & Handling Fee at RC d
$E_{s,d}$	1 if store s returns products to RC d, 0 otherwise
$E_{d,d'}$	1 if RC d forwards products to RC d', 0 otherwise
$E_{d,w}$	1 if warehouse w receives products from RC d, 0 otherwise
$E_{d,sc}$	1 if RC d is active in scenario sc, 0 otherwise
$E_{s,d,sc}$	1 if store s returns products to RC d under scenario sc, 0 otherwise
$E_{d,w,sc}$	1 if (active) RC d forwards products to warehouse w under scenario sc, 0 otherwise
$E_{d,d',sc}$	1 if (active) RC d forwards products to (active) RC d' under scenario sc, 0 otherwise
$TCD_{s,d}$	Per unit Transportation Cost to RC d from a store s
$TCT_{d,d'}$	Per unit Transportation Cost to RC d' from a RC d
$TCV_{d,v}$	Per unit Transportation Cost to vendor v from a RC d
$TCW_{d,w}$	Per unit Transportation Cost to warehouse w from a RC d
TCL_d	Per unit Transportation Cost to liquidation at RC d
$Q_{s,p}$	Number of items (quantity) of product p in store s
DAC_d	Activation Cost of RC d
WRA_w	Maximum Return Allowance (in terms of value, $) to a warehouse w
$WDC_{p,w}$	Warehouse Demand Capacity of product p (in terms of # of units)
$RT1_v$	Return Threshold 1 for the available vendor funds of vendor v
$RT2_v$	Return Threshold 2 for the available vendor funds of vendor v
VPF_v	Vendor v Penalty Fee (%)
$VPF1_v$	Vendor v Penalty Fee (%) for returns up to $RT1_v$ (for the current contract term)

$VPF2_v$	Vendor v Penalty Fee (%) for returns between $RT1_v$ and $RT2_v$ (for the current contract term)
$VPF3_v$	Vendor v Penalty Fee (%) for returns over $RT2_v$ (for the current contract term)
VF_v	Vendor v available Funds
PLB	Profit-Loss Budget for the retail company
N	a slack for the profit-loss budget constraint ($),
M	a large number
EI_s	Effective Inventory value of store s ($)
IEI_s	Ineffective Inventory value of store s ($)
EI_{st}	Total Effective Inventory value of store-type st ($)
IEI_{st}	Total Ineffective Inventory value of a store-type st ($)
$NIER$	National Ineffective Ratio target for the retail company's owned stores
$FRSA$	Franchise Stores' Total Reverse Supply Chain Amount ($)
$TSRA$	Total Reverse Supply Chain (Store Removal) Amount ($)
$OVPF1_v$	Original Vendor v Penalty Fee 1 (%)
$OVPF2_v$	Original Vendor v Penalty Fee 2 (%)
$OVPF3_v$	Original Vendor v Penalty Fee 3 (%)
$VPF1Min_v$	Minimum Vendor v Penalty Fee 1 (%)
$VPF1Max_v$	Maximum Vendor v Penalty Fee 1 (%)
$VPF2Min_v$	Minimum Vendor v Penalty Fee 2 (%)
$VPF2Max_v$	Maximum Vendor v Penalty Fee 2 (%)
$VPF3Min_v$	Minimum Vendor v Penalty Fee 3 (%)
$VPF3Max_v$	Maximum Vendor v Penalty Fee 3 (%)
$ORT1_v$	Original Vendor v Return Threshold 1
$ORT2_v$	Original Vendor v Return Threshold 2
$RT1Min_v$	Minimum Vendor v Return Threshold 1
$RT1Max_v$	Maximum Vendor v Return Threshold 1
$RT2Min_v$	Minimum Vendor v Return Threshold 2
$RT2Max_v$	Maximum Vendor v Return Threshold 2
TVR_v	Total Available Vendor v Funds
EVR_v	Expected Vendor v Return amount
HVR_v	Historical Vendor v Return amount
$BPE1_v$	1 if expected return of vendor v is greater than $ORT1_v$, 0 otherwise
$BPE2_v$	1 if expected return of a vendor v is greater than $ORT2_v$, 0 otherwise
$BPH1_v$	1 if historical return of a vendor v is greater than $ORT1_v$, 0 otherwise

$BPH2_v$	1 if historical return of a vendor v is greater than $ORT2_v$, 0 otherwise

Decision Variables:

$ds_{s,p}$	1 if product p in store s should be scrapped, 0 otherwise
$rt_{s,p}$	1 if product p in store s will be returned, 0 otherwise
$rt_{s,p,d}$	1 if product p in a store s will be returned to RC d, 0 otherwise
ac_d	1 if RC d is activated, 0 otherwise
$btr_{p,d,d'}$	1 if RC product p is transferred from RC d to RC d', 0 otherwise
$tr_{p,d,d'}$	product p quantity transferred from RC d to RC d'
$lq_{p,d}$	product p quantity liquidated at RC d
vr_p	product p quantity returned to a vendor
$vr_{p,d}$	product p quantity returned to a vendor from a RC d
$dp_{p,d}$	product p quantity returned to a vendor for a Deposit Value from RC d
wh_p	product p quantity returned to a warehouse
$wh_{p,d,w}$	product p quantity returned to a warehouse w from RC d
vf_v	total vendor v used funds
x_{sc}	1 if scenario sc is realized, 0 otherwise
$vpf1_v$	Vendor v Penalty Fee 1 (%)
$vpf2_v$	Vendor v Penalty Fee 2 (%)
$vpf3_v$	Vendor v Penalty Fee 3 (%)
$rt1_v$	Vendor v Return Threshold 1
$rt2_v$	Vendor v Return Threshold 2
$bve1_v$	1, if expected returns to vendor v are greater than $rt1_v$, 0 otherwise
$bve2_v$	1, if expected returns to vendor v are greater than $rt2_v$, 0 otherwise
$bvh1_v$	1, if historical returns to vendor v are greater than $rt1_v$, 0 otherwise
$bvh2_v$	1, if historical returns to vendor v are greater than $rt2_v$, 0 otherwise

Declaration of Authorship

I, Mehmet Erdem Coskun, declare that this thesis titled, "Optimization Models For Retail Reverse Supply Chains" and the work presented in it are my own. I confirm that:

- In Chapter 1, I have introduced the readers to the basic definitions, concepts and frameworks in RSC and defined the motivation behind my research and this thesis.
- In Chapter 2, I have reviewed the literature on reverse logistics, reverse supply chains and closed-loop supply chains, focusing on RRSC and identifying gaps in the literature.
- In Chapter 3, I have designed and developed a RRSC model to optimize product allocations subject to profit-loss budget, developed, and tested a heuristic algorithm to solve the resulting problem efficiently.
- In Chapter 4, I have designed and developed a comprehensive RRSC model to manage product returns to optimize inventory returns and redistribution in the supply chain. I have developed a multi-stage heuristic algorithm and tested it numerically to confirm its superiority over a commercial optimization solver.
- In Chapter 5, I have designed and developed a mixed integer nonlinear programme (MINLP) to find optimal penalty fees and a return thresholds that aids with retailer-vendor product return contract negotiations. I developed a decomposition heuristic that finds solutions efficiently and provides insights about the optimization model.
- In Chapter 6, I have summarized my work in the previous chapters and provided future guidance for potential areas of research in the RRSC field.

Chapter 1

Introduction

1.1 Background and Terminology

The material flow opposite to the conventional supply chain flow is the concern of the field of reverse logistics (RL) (Stock 1992, Kopicki et al. 1993, Fleischmann et al. 1997). The Reverse Logistics Executive Council provides the widely accepted definition of reverse logistics as: 'The process of planning, implementing, and controlling the efficient, cost effective flow of raw materials, in-process inventory, finished goods, and related information from the point of consumption to the point of origin for the purpose of recapturing value or of proper disposal' (Rogers and Tibben-Lembke 1999). Considering the above definition, reverse logistics deals with the collection, transportation and distribution activities of products (used or new products-parts-materials (PPM)) that are not needed anymore by its users. In this thesis, users will mostly refer to businesses such as sales centers. Reverse logistics encompasses activities from the collection of unwanted PPMs (UNPPMs) from end users to remanufactured products, that may be reused by consumers, or the disposal of the collected items. In addition to the activities of reverse logistics, in reverse supply chain (RSC) we are also concerned with coordination and collaboration issues between the different supply chain partners that are involved in the reverse flow activities. In this thesis we adopt this wider view where we consider decisions such as location of distribution centres as well as reverse flow contract negotiation issues. Companies need to design networks that efficiently collect recoverable PPM from end users, inspect those products to assess their quality, assess the value or material recovery options, perform any reprocessing to make them reusable again and, if possible, redistribute the reprocessed products to the markets (primary or secondary).

Recovered products can be reused in markets in different ways. In Table 1.1 we show the different properties of recovered products with some examples. RL modelling types depend largely on what type of recovery is being used (Fleischmann et al. 1997).

Property	Examples
Form of reuse	• Direct in primary market as a new product. • Direct in secondary market as repaired/refurbished product • Indirect reuse as parts or material to produce new item
Types	• main product for reuse • repair or refurbishment • parts of the product for recovery • parts/material for recycling
Parties	• collection points • inspection/testing/sorting locations • disassembly facilities • recovery sites • remanufacturing centers • (re)processing centers • incineration and landfill sites, distribution • return centers
Reason	• material recovery • cost reduction of source raw materials • parts or products • scarce raw materials • waste reduction • reduction of disposal costs • promoting 'green' image • legal obligations of waste generation

Table 1.1: Product recovery properties.

The main reason for a product to be recovered is to regain the value that still incorporated within its used or end-of-life version. There are many ways to regain value from a returned product:

1. **Direct Reuse**: End-of life or returned products can be reused and resold directly, after some detail cleaning and minor maintenance without being repaired or reprocessed, in the primary market or secondary markets. Some of the examples for direct reuse are pallets, bottles, and containers.

2. **Repair**: Used or returned products can be repaired in order to bring them into a working condition, possibly with a loss in the quality, with a fraction of its original manufacturing costs and then can be reused or resold in the primary or, mostly, in the secondary markets. Hence in a product repair process, product structures

are kept intact. Some of the repaired product examples are industrial machines, electronic equipment and appliances.

3. **Remanufacture**: Used or returned products which can not easily be reused directly or repaired with a small cost, can be brought back into a new condition or a desired level quality after disassembly, overhaul, part replacement and assembly operations. Products get disassembled so that their non-working parts are replaced with working parts coming from some other dismantled product(s) for the purpose of remanufacturing of the same product (remanufacturing), or some working-condition parts are used in the remanufacturing process of some other product (retrieval). Depending on the remanufacturing condition, product identities and structures might be altered in a product remanufacturing process. In most RL literature it is assumed that the remanufacturing process preserves the product identity and structure. However, if the collected product is dismantled into its parts so as to be used in other products' remanufacturing process then the identity of the collected product would not be kept intact. Some of the remanufactured product examples are computers, machinery tools and engines.

4. **Recycle**: Used or returned products which can not be recovered in whole or in part, might contain recoverable materials which can be obtained by performing the necessary disassembly, sorting and reprocessing operations that is referred to as recycling. As a results, product structures can not be conserved in product recycling. We only recover the materials we need for some other or the same manufacturing process. The remaining parts, items or materials that can not be recovered are scrapped and end up either in landfills or to be incinerated. Some of the materials that are recovered after a recycling process are metal, glass, paper, and plastic.

In general, manufacturers tend to perform direct reuse, repair and remanufacturing options in-house because they have knowledge about the manufactured product. However, recycling is often carried out by a specialized third party company since material recovery from a used product is a total different process than recovering a known product by repairing or remanufacturing (Thierry 1997). Furthermore, if a company does not have some parts of the RL processes in order, in addition to the actual recovery processes, RL activities such as collection of used products and/or reverse distribution of the items can also be carried out by a specialized third party logistics (3PL) provider.

1.1.1 Product or Material Recovery Options and Activities in RSCs

There might be many echelons in a RL channel. Possible operations in these echelons are: collection, inspection/testing, sorting, dismantle/segregation/disassembly, repair, remanufacture, refurbish, recycle, reuse, liquidate and disposal/discard/scrap/incineration/landfill with the regular forward channel transportation/distribution/redistribution, handling, intermediate processing and storage activities. Even though the collection function of the RL is similar to the forward supply chain activities, where the products are procured, the remaining activities are more specific to RL.In *Figure 1.1* We illustrate the general framework of a RL network with possible activities, actors, and product recovery facilities . *Figure 1.1* is a very detailed version of a RSC network system for products that are complex in nature (e.g., products with modular parts such as cars and computers) and includes all the possible activities, actors, and facilities.

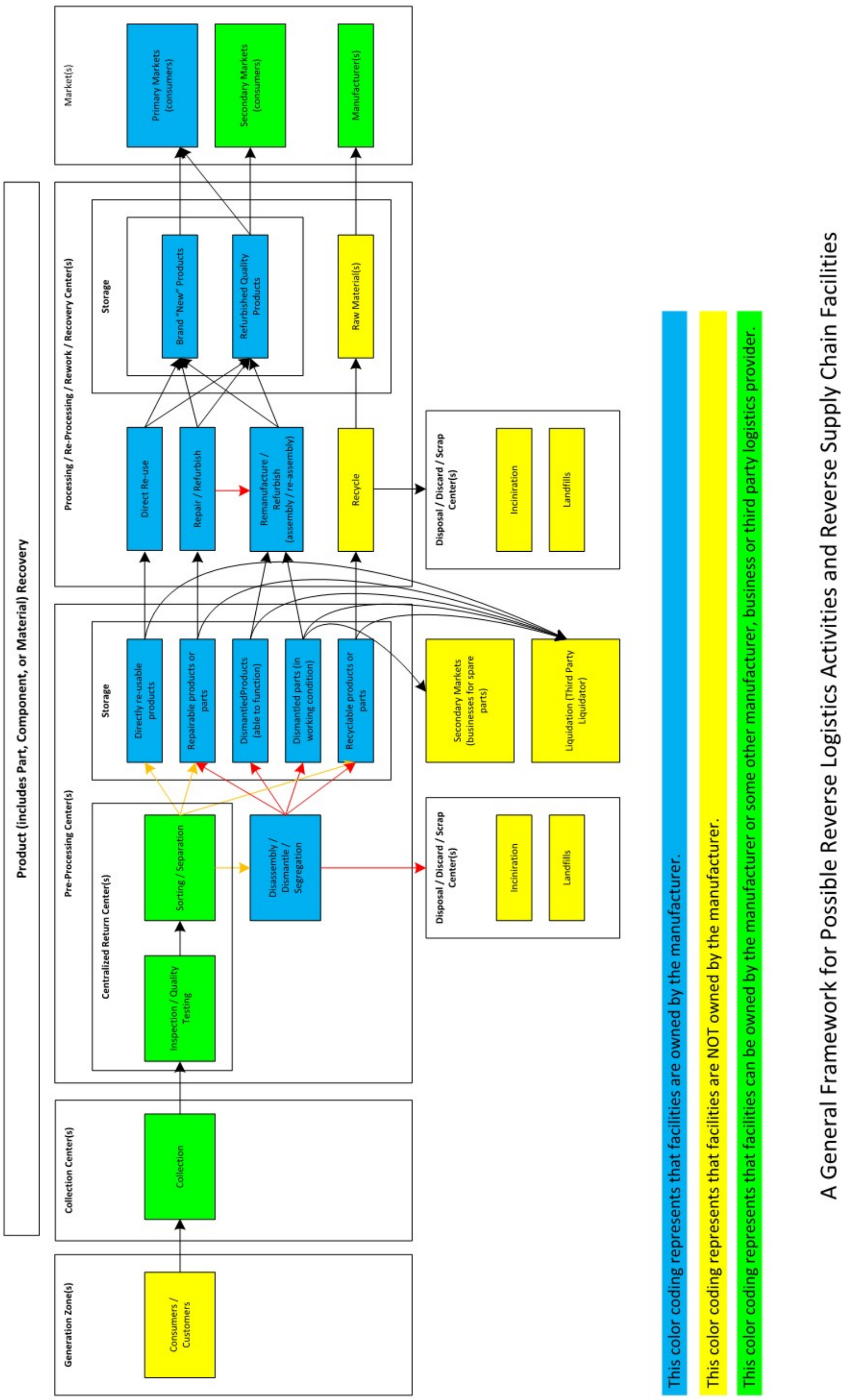

FIGURE 1.1: Reverse Logistics Diagram, A Comprehensive Look at a Modular Product's Reverse Flow

Collection is the first operation that needs to be done in a RL network. It refers to all activities of collecting the used (unwanted, end-of-life, end-of-use products, and also defective, damaged, faulty, recalled, under-warranty products) or new products (unsold, unwanted, end-of-life, defective, damaged, faulty products especially in business-to-business settings for excessive or seasonal inventory) from end consumers, customers, or clients at certain collection locations. It is the physical movement of products from the consumer to the company or the third party collector. The collection operation may include take-back, purchasing (buy-back), transportation, and storage activities. Some of the examples of collection points are recycling bins, retail stores, specialized collection locations, and centralized return centers/stores/warehouses. Deposit-refund systems are also developed by the manufacturers in order to promote product returns for collecting product casing, packing, containers, bottles and cans. Deposit-refund might also be considered as a purchasing (buy-back) function of the collection operation where the manufacturer refunds a certain amount for the returned product. Some companies, or governments/municipalities, also provide collection centers under an incentive program for some certain types of products. For example, consumers recycle their used batteries directly to specialized recycling bins.

Inspection and quality-testing denotes operations that determine whether a collected product is recoverable. If the product is recoverable then inspection and quality checking identifies to what extent the product can be recovered by additional processing. If a product does not pass the inspection and quality testing for product recovery, then the product (or some of its parts) would be sent for disposal. If the collected products pass the inspection and quality-testing stage, they will be routed to related product (and/or material) recovery operations (direct reuse, repair, remanufacture, recycle), which can also be observed in *Figure 1.1*, depending on the results of the inspection stage. This stage is called the sorting/separation stage. In general, inspections and quality-testing, and the sorting stages occur in the same location.

For the products that passed the inspection stage and will be routed for remanufacturing, they need to be dismantled/disassembled first. Dismantling / disassembly is a stage where collected products are disassembled into their constituent parts or components for the next processing stage. Depending on whether the product will be dismantled totally or not, the product can be remanufactured into a 'working' product or dismantled into its constituent components so that they can be used in some other products' remanufacturing process or be sent to spare parts market for sale. As for the remanufacturing process of a product, the full set of parts or components are needed. Therefore, if there are any shortages of parts for the remanufacturing of the product, they can be acquired by ordering from a supplier or using old parts that were extracted from other disassembled products. Since the disassembly stage is followed by the remanufacturing stage, the disassembly/dismantling stage can occur at the same location as the remanufacturing stage in some RL networks and hence a separate disassembly center will not be needed. However, since some companies offer a large variety of products and those products are composed of many different parts and components, the RSC may need separate disassembly centers other than the remanufacturing facilities.

In product recovery stages, a product or some of its parts can be recovered and reused. However, collected products can also be sold to a third party buyer This stage is referred as the liquidation stage in RL, where a value of the product is recovered rather than its physical stock.After the re-processing centers recover as much value out of the collected products for remanufacturing some portion of the unusable parts might be used for raw material recovery. This process is called the recycling stage where certain raw materials from disposed items can be extracted for building new products. Some of the main raw materials are metals and plastics. Generally, the recycled materials need to be further processed (melted or chemically treated) by third party recycling companies or municipalities in order to be used in the same or other industries. The last, and least wanted option, is the disposal of the collected products if the company cannot recover any material or value from the collected product.

When a reverse distribution channel is to be designed, determining suitable echelons is the most crucial issue. For example, early inspection and testing close to collection sites instead of centralized inspection and testing close to return/distribution center might save on transportation costs of useless products. However, a large number of inspection testing facilities close to collection sites might involve heavy costs compared to few centralized inspection and testing facilities with relatively reasonable costs close to return/distribution centers. As shown in *Figure 1.1*, some of the echelons in the RL network may include several types of facilties. A RL network will include some or all of these echelons/facilities and above mentioned RL functions/activities depending on the company, its supply chain network design and size, distribution levels and options, products they produce/sell and materials they were built from, market needs and government regulations.

1.1.2 Differences Between Forward and Reverse Supply Chains

There are some significant differences between a forward supply chain and a reverse supply chain. The ability to forecast volumes, both the supply and demand quantities, is one of the major differences. In forward supply chains, products, in general, have a known expected demand.Supply quantity can also be estimated since the production quantity of a manufacturer can be assumed as a known factor. However, forecast for a recovered product is harder to estimate since the quantity of supply or demand (for secondary market) is highly random. In addition, the quality and timing would also be uncertain which makes the problem much more complex to analyse mathematically. Another major difference is the unclear disposition routes of the productsIn a forward supply chain, the echelons and routes for a product are known for a product, however, a collected product's route or location are not known until they are inspected or disassembled There are also other differences between a reverse supply chain and forward supply chain such as the RL having invisible cost structure, distribution speed not being a priority and many-to-one echelons (i.e. collection centers-to-manufacturer) structure instead of one-to-many (i.e. production facility-to-stores) echelon structure (Tibben-Lembke and Rogers 2002, Tibben-Lembke 2002).

Comparing the fundamental characteristics of RL to forward distribution, one of the main differences is the uncertainty of supply. In traditional (forward) supply chains, the supply amount is typically assumed known with certainty. However, in RL, the supply quantity is almost always uncertain.Thus, in most literature studies we find that with the exception of demand, the supply quantity, quality and timing are, in general, known factors or decision variables in forward supply chains. In RL models, especially if the product will recovered, the demand is usually assumed uncertain since both its quality and quality are difficult to predict with accuracy.

1.1.3 Reverse Supply Chain Vs. Closed-loop Supply Chains

RL channels can be within an closed(open)-loop supply chain system where a product is (not) returned to its original manufacturer or the channel is (not) integrated with the forward supply chain.In general, forward distribution and RL are implemented in a sequential manner. However, there might be possible cost savings if the forward and reverse channels are integrated in as a closed-loop supply chain network. The integration can take place at different levels. For example, while recycling functions or third party product recovery often occur in an open-loop supply chain system, direct reuse, repair or remanufacturing functions often occur in a closed-loop supply chain. The problem of deciding about the level of integration is not an obvious one as it relied in several factors such as routing levels, distribution channels and markets involved. In reality most forward distribution systems are not equipped to handling product movement in the reverse channel Jayaraman et al. (1999) . This is mainly due to the fact that returned products require special handling in collection, transportation, storing, or sorting

1.2 Motivation

Although RSC and RL is widely studied, we have observed that almost all the related literature concentrated their efforts on the RL network design issues where their goal is to minimize costs while deciding about which RL facilities to open. These location-allocation models address mostly strategic decisions and assume all products are returnable. There is a need for research on tactical and operational models to efficiently operate exisiting RL networks.

As for the RSC literature, we find that they concentrated their efforts on modelling the RSC framework from a manufacturer's point of view, hence modeling the problem as a network optimization problem, where the manufacturer is trying to collect its used/new/end-of-life products from supply generation zones. The models objective were to identify where to open collection centers in order to optimize costs for collecting these products. Most of the models were location models, therefore modeling a binary selection problem, for the RSC of the manufacturer which included facilities that are owned by the manufacturer such as collection centers, sorting, quality checking, disassembly/dismantling, repairing, and remanufacturing facilities.These models assumed all items are collected (or processed) where they assumed an uncertain supply, which brought attention to the complexity of the problem. However, these models did not consuider product selection or allocation since all items were returnable.

In many realistic cases, a manufacturer of a product is not the only entity that is capable of collecting products/material via its RSC network and hence a decision maker of its RL. Depending on the business strategy, a manufacturer might sell products to other businesses and then those products eventually get sold to end consumers If a manufacturer sells products to consumers using interim channels, they will not be able to collect the used items from the consumers or new-but-unsold products at sales locations because the products leave the manufacturer's supply chain at a certain point and can only come back via returns channels of the parties that sold them, which in most cases, is a retail company. Therefore, we find that there is a need to consider RSC models that have the businesses that sell to consumers as the main decision maker. These businesses, like retailers, have their own forward and reverse supply chainswhich is separate from that of the manufacturer's supply chain As a result, the retailers' own network structure, especially RSC network, should be modelled separately and carefully considering the complex echelons/layers of their network.RRSC environments has not been considered or modelled in the existing vast RL literature, with the exception of the work of Yuliawati et al. 2021; Kaboudani et al. 2020; Das 2012, which offer some starting modeling framework.

Inspired by a need from one of our industrial partners and the gap we identified in the literature, we concentrated our efforts on modelling the RL activities of a retail company. In the existing RL literature, there is no framework or model that considers the case when only a subset of product is returned This is especially more important for the RRSC since retail companies optimize their inventory based on either inventory levels or budget constraints. Therefore, RSC considers only a portion of the available products (whether they are unsold, damaged, faulty, end-of-life or customer-returned) to be reverse flowed while taking into account inventory and/or budget limitations. Budgetary constraints are crucial to a company's bottom line because it directly affects the fiscal year budget. Therefore, in our first model, we focus on these issues. We consider operational decisions of a retail company, a selection process of products to be return flowed, with the allocation decisions that will optimize the reverse distribution activities.

In our second problem, we address another important RRSC that takes into account inventory optimization via RL activities. We design the supply chain network of the retailer where we consider its stores, warehouses, distribution centers, vendors and liquidators. The logical paths of moving products among these facilities are carefully constructed and we consider physical and financial limitations of the problem via capacity, demand, vendor refunds. We consider a realistic retail environment by incorporating the company's owned stores, its franchise stores and their conflicting objectives. In addition to the operational decisions, we also incorporate the strategic network design problem in our model to find which distribution/return centers should be used. The objective of our model is also designed in a way that captures all the complexities (cost of physically moving items, penalties, margin losses and activation costs) of a retail supply chain and its detailed cost structure related to this activity.

Large retail companies deal with hundreds, and sometimes thousands, of different vendors. To address the issue of returned or unsold items, retailers and their vendors have developed product return agreements with a certain multi-layered penalty structure that affect both parties positively and negatively at the same time. From the retail company's perspective, returning unsold, customer returned, damaged, broken or faulty products back to its original vendor opens inventory space for new or already selling products, increases inventory health, and its cash balance because of the money received from vendors for the returned items. Even though this is a positive effect for the retailers, they have to pay penalties for every product they return back to its vendor and this results in loss of capital which could have been used for purchasing products that have a potential to be sold. From the vendors' perspective, receiving unsold, customer returned (without damage), broken or faulty products may affect the vendors negatively since they receive products they do not know the condition of and they have to incur the receiving, unpacking, inspection and sorting costs. Also, vendors' reputation and recognition might be damaged from having a high percentage of broken or faulty products. However, returned products can be resold to another retailer as is or it can be further processed internally for quality control purposes. Since the vendor receives products that can potentially be resold directly or with some minor internal processing to other retailers, and at the same time charges a penalty to the retailer for every returned product, they benefit from the returns. It is therefore crucial for retailers to identify and negotiate optimal penalty strategies with vendors.We consider this problem in Chapter 5, an area that has not received mucj attention in the literature.We take a retailer's perspective to re-negotiate an existing product return contract using historical returns and future expected returns and identify the optimal penalty strategy.

1.3 Thesis Structure

The rest of this thesis is structured as follows. We review the extensive literature on RSC in Chapter 2 and provide evidence of the gap and lack of research specifically in retail RSCs. We also outline our main thesis contributions and how they fill in some of the identified research gaps. Based on the distinct characteristics of RRSC, three major issues are then studied in Chapters 3–5. Reverse logistics operations at a retailer are often constrained by the terms of the vendors contracts. One such important constraints is the level of refund that will be obtained from the vendors. Such refund feeds directly in the retailer's budget and they perform their RL operation in order to keep their budget variances under control. We refer to such constraints as budgetary constraints and address them in Chapter 3. Inventory impacts of RL operations in retail are considered in Chapter 4. In particular, we consider the choice of distribution centers and rerouting of returned stock. We then make use of our oprimization models to derive insight for vendor contract design. We consider optimal contract-terms, such as penalty and thresholds, in Chapter 5. In all three modelling chapters we propose novel models and solution approaches and conduct numerical experiments. Results, comparisons and useful insights are then discussed. The thesis ends with a conclusion and a discussion on potential future areas of research in the RRSC field in Chapter 6.

1.4 Relationship between RRSC Models

When a retailer wants to clean excessive inventory from all of its stores, they have two options: allocating inventory based on a given profit-loss budget or do a full-fledged inventory optimization. If the retailer decides to move forward to spend a pre-calculated budget to clean a certain portion of its inventory, the model we discuss in Chapter 3 can be utilized since this model reverses margins that are lost due to income loss against a given budget. However, if the retailer decides to move forward with a full-fledged inventory optimization to minimize all of its reverse supply chain costs, then the model we discuss in Chapter 4 can be utilized since in that chapter our goal is to optimize all of the related reverse supply chain costs.

The above models can generally be used sequentially during a fiscal year. In general, a RRSC activity is launched several times during a year and a retailer might want to spend available profit-loss budget equally during the RRSC activities. However, a retailer might also want to optimize all of its inventory costs during the year and spend the remaining budget optimally at the end of its fiscal year. Therefore, the models that are defined in Chapter 3 and Chapter 4 are highly interrelated and can be used based on company inventory management strategy and objectives during a fiscal year.

The problem that is discussed in Chapter 5 is a strategic decision making problem to minimize penalty costs that will paid to many vendors in the upcoming return cycles. The optimal contract terms/parameters that will be identified in this model, if accepted by the vendors, will be used for several years until the return contracts are due. Therefore, the optimized contract parameters that will be identified in this problem (in order to better negotiate with every vendor), is one of the most important parameters to implement in the inventory optimization model that is discussed in Chapter 4. Even though the operational model that is discussed in Chapter 4 runs several times a year to optimize RRSC inventory costs, some of the most important parameters that are being used to do that is the result of the optimal contract parameters that is extracted from the strategic model discussed in Chapter 5. Therefore, all the 3 problems we have discussed in this thesis are highly interrelated and should be used in a certain order to optimize all related RRSC costs.

Chapter 2

Literature Review

In this chapter we summarize the most relevant literature, with a focus on reverse supply chains in general and for retailing specifically, identify research gaps, and outline our main thesis contributions.

2.1 Reverse Supply Chain Literature

The reverse supply chain (RSC) literature started gaining interest from the early 1990s (e.g., see Min 1989; Caruso et al. 1993; Kroon and Vrijens 1995; Melachrinoudis et al. 1995; Fleischmann et al. 1997; Spengler et al. 1997; Marin and Pelegrin 1998; Barros et al. 1998; Jayaraman et al. 1999; Louwers et al. 1999; Krikke et al. 1999). However, in the past two decades the interest and published work have gone up significantly which included many distinct approaches from frameworks, return policy designs, RSC contracts, network design models to allocation models. The early models were mostly investigating recycling end-of-life cycle products such as carpets, used cell phones, photocopiers, solid waste, steel by-products, sand from construction waste, hazardous waste, and returnable containers. They modeled the RSC as a product recovery and distribution network where the authors concentrated their efforts mostly on the location-allocation decisions of collection centers, re-manufacturing facilities, or recycling facilities and the distribution and allocation of the collected materials.

Most of the literature modelled the location and allocation decisions of a RSC from the manufacturer's point of view where the manufacturer was collecting end-of-life products from collection centers, recycling bins or waste bins. The manufacturer's objective was mostly to identify where to open collection centers, where to put recycling or waste bins, where to open disassembly, repair and/or re-manufacturing facilities and how to distribute the load among these facilities (e.g., see Ashayeri and Tuzkaya 2011; Ferguson et al. 2011; Tuzkaya et al. 2011; Vidovic et al. 2011; Nenes and Nikolaidis 2012; Alumur et al. 2012; Das and Chowdhury 2012; Hosseinzadeh and Roghanian 2012; Dat et al. 2012; Niknejad and Petrovic 2014; Roghanian and Pazhoheshfar 2014; Soleimani and Govindan 2014; Alumur and Tari 2014; John et al. 2015; Godichaud and Amodeo 2015; Yun 2015; Li et al. 2016).

From the 2000s on, the interest in RSC literature has grown exponentially and there have been hundreds of articles (with the inclusion of closed-loop supply chain, CLSC, literature) published in various journals since then, where most of them model location or location-allocation decisions of recoverable end-of-life cycle products or waste material similar to the prior studies. The proliferation of study in this area can be witnessed by the fact that 51 review papers were published from as early as 1997 to 2021, at a rate of two reviews per year. The interested reader is referred to the most recent reviews of Van Engeland et al. (2020) and Ambilkar et al. (2021), where the former has presented a categorization of the published literature.

2.2 Reverse Retail Supply Chains

Even though there has been a considerably large amount work in the RSC (and CLSC) area, the related literature in RRSC is very scarce, especially that which is concerned with modelling. The same observation was noted int the review papers by Dias et al. (2019) and Borba et al. (2020), were the also review qualitative and conceptual works. Only Yuliawati et al. (2021), Das (2012) and Wojanowski et al. (2007) have presented modelling frameworks of the RRSC activities.

In this thesis we focus on RSC in product recovery-remanufacturing, which is s departure from the main stream literature. In the remainder of this section we focus on reviewing the RSC literature in retail environments. We consider both studies that cover the literature on RSC in retail as well as the effects of RL on the retailer. Tibben-Lembke and Rogers (2002) compared the forward and reverse logistics (RL) in a retail environment with a focus on the reverse flow of the productsHorvath et al. (2005) developed a Markov chain approach to model the expectations, risks, and potential shocks associated with cash flows stemming from retail RL activities. Chaves and Alcantara (2006) investigated the conflict between the industry and the retail from a manufacturer's point of view through a qualitative exploratory research in two major companies of the food sector. Bernon and Cullen (2007) created a framework for managing RL through adopting the three management approaches of integration, collaboration and evaluation and argued that the level of returns currently experienced by retailers could be reduced significantly if organizations managed product returns in a holistic way. Wojanowski et al. (2007) studied the interplay between industrial firms and government concerning the collection of used products from households and presented a retail-collection (drop-off facility) network to determine the sales price that maximize the firm's profit under a given deposit-refund. They have determined the net value that can be recovered from a returned product as a key driver for the firm to voluntarily engage in collection. They have showed that a minimum deposit-refund requirement would not achieve high collection rates for products with low return value.

Moise et al. (2008) discussed the importance of RL to retailers for meeting short-run financial obligations or opportunities. Sonya Hsu et al. (2009) studied the business process of RL by focusing on studying the activities of the distribution center of a major department store and found that the biggest problem a central return center is facing is the time required for managing damages when no return authorization is forthcoming from the vendor. Jack et al. (2010) investigated the capabilities of RL for retailers to enhance their return policies and improve their overall cost position through surveying retailers, They identified that resource commitments and contractual obligations positively influence RL capabilities and that these capabilities result in cost savings. They also reported that RL capabilities partially mediates the relationship between resource commitments, contractual arrangements, and RL cost savings. Bernon et al. (2011) presented a conceptual framework for managing retail RL operations. They observed that it needs to be managed as an integrated supply chain activity because of its multi-faceted operation structure and proposed operational performance, organisational integration and management reporting and control management dimensions.

Das (2012) proposed a mixed-integer programming (MIP) model for the strategic production and distribution planning of a supply chain integrating RL system which included collection, recovery and marketing of recovered products, in addition to returned components and packing/wrapping materials. They used retail outlets as a two-way channel for marketing new products, collecting used/returned products and re-marketing recovered products as a way of promoting an effective product recovery system in SC operation and optimizing costs. The model followed a two-step process that addresses strategic decisions about product recovery in the first step, and the integration of the recovery process into overall SC decisions in the final step. Bernon et al. (2013) explored supply chain integration enabling practices, their benefits and barriers in a retail product returns process context through a case study of an original equipment manufacturer and two retailers. They found out that management of retail product returns can significantly benefit both an OEM and its customers when appropriate SCI enabling practices are deployed but barriers are driven by the characteristics of product returns processes. Olariu (2014) presented a theoretical approach to retail RL identifying key aspects from network design, facility location, outsourcing, green supply chain management, organization integration to information technology.

Bajor and Babic (2014) studied retail level returns in a Croatian market and looked at characteristics of returns and routing these products from the retail level. They found that the majority of products in return are directed from final consumers and are non-current inventories of the distribution chain. Bernon et al. (2016) explored the subsequent impact on the levels of consumer retail returns experienced through online sales and the emergent returns management strategies used by retailers in relation to network configuration and returns management processes. They found that return rates for online retailing can be double those for stores, while return levels for “considered purchases” remain similar. Dias and Braga Junior (2016) analyzed the practices of RL by a retailer and measured the amount of waste generated by each department via monitoring the amounts of cardboard and plastic discarded by each department. Beh et al. (2016) examined the role of an alternative approach, second-life retailing in RL. They demonstrated the essential characteristics of second-life retailers and showed that it could bring additional revenues, enhanced sustainability and democratization of consumption.

Mostert et al. (2017) have done a qualitative study to explore the perspectives of retailers regarding supply chain integration in the context of product returns for consumer electronics. They identified that retailers made efforts to increase internal integration relating to improving information availability, aligning cross functional processes and improving inter-firm relationships and external integration efforts to improve the intra-firm flow of information, reduce the number products returned to suppliers, expedite the returns process in specific instances and align processes. Bernon et al. (2018) presented a conceptual framework that supports the adoption of circular economy values within retail RL operations and found that embedding circular economy values necessitates the adoption of a multi-faceted approach.

Beitelspacher et al. (2018) explored the relational implications in the business-to-business (B2B) context and found out that when salespeople respond to returns by engaging in relationship building behaviors, these behaviors are noted by the retailer, which in turn results in fewer returns in a future time periods. Panigrahi et al. (2018)

identified critical factors and developed a RL strategic framework for improving customer satisfaction and managing retail returns through reviewing published work and interviewing logistics managers in the retail industry. Dias et al. (2019) reviewed the retail RL literature between 2007 and 2016 and only found 10 publications for the final analysis of the study that are relevant. Demajorovic et al. (2019) analyzed the development of the relationships between the internal areas of a retail company, its suppliers and customers involved in the management of RL in two contexts: product return and returned packaging used in moving product logistics. Their results showed conflicts and lack of objective alignment between the external actors, i.e., suppliers, producers and partners, and within the focal firms, between the commercial and logistics departments due to the perception that RL is a cost-generating process and thus undermining the effectiveness of the reverse flows of products and packaging in retail.

Xu et al. (2019) probed into the existing strategic models, reviewed relevant literature and put forward effective management strategies of RL in retail industry. Junior et al. (2020) analyzed the willingness to implement RL in supermarket retail and used logistic regression to generate a model for evaluating the disposition of products. They have also observed the absence of a model for implementation and guidance of RL in a retail environment. Frei et al. (2020) investigated the extent to which sustainable practices and circular economy concepts have been implemented in retail returns systems and identified vulnerabilities, barriers, and challenges to the implementation of sustainable circular practices via in-depth interviews, observations, retailer website reviews and retail community workshops. Borba et al. (2020) proposed a theoretical description of RL applied to omnichannel retail. They presented a conceptual framework for a holistic view and identified 43 return barriers including high investments, product restocking, additional transportation costs and poor communication. Gustafsson et al. (2021) analyzed the current performance of a retailer's e-commerce and return operations by estimating costs generated by product returns, including product handling costs, tied up capital, inventory holding costs, transportation costs, and order-picking costs from 2,229 return transactions at a Scandinavian fashion footwear retailer and identified that the cost of a return is approximately 17% of the prime cost. The major cost elements were product handling costs and transportation costs, which together amount to 72% of the total costs. They tested a digital product fitting technology with the retailer's products and provided estimations on how such technology could affect product returns. They observed that fitting technology can cut fit-related return costs by up to 80%. Yuliawati et al. 2021 proposed a retail-oriented closed-loop supply chain (ROCLSC) where a retailer takes charge of the collection, distribution and remanufacturing processes in a two stage, manufacturer and retailer, supply chain system. They developed a mathematical model to maximize the profit of each party. They introduced a fixed rate and flat rate mechanisms used in the business-to-business (B2B) systems. They showed that the retailer will get higher profits when the product returns are acquired through the fixed rate mechanism

We remark that almost all the literature on retail RL is concerned with product returns instead of how the retailer internally handles these returns in order to minimize its costs. Some of the above articles discuss the potential cost aspect of these returns to the manufacturer, however, none of them developed models that would address this issue.In our work, we concentrate our efforts on the modelling of RSC network of the retailer and optimize its costs from the retailers point of view.

2.3 Gaps in Modelling Frameworks in RRSC Literature

Even though there are studies done on the RRSC literature, most of the research that was published was on the exploratory and qualitative side. Researchers developed conceptual frameworks, interviewed retail supply chain managers, surveyed employees and done case studies. The lack of quantitative work can easily be observed from Table 2.1 where we find only four studies that presented quantitative models: Horvath et al. 2005; Wojanowski et al. 2007; Das 2012; Yuliawati et al. 2021.

Bernon et al. (2011) pointed out that the the area of retail RL is still immature and developing. In the RRSC literature review by Dias et al. (2019), only 10 relevant studies were found in the literature. They observe that "another factor that reaffirms this evidence [lack of studies] is the predominant exploratory character of the research found" and call for thBy 2020, Junior et al. (2020) als reaffirms that "in the literature, there are no studies that suggest a model for implementation and guidance for RL". Borba et al. (2020) added that "the link between RL and omnichannel area is recent and publications are still scarce." Yuliawati et al. (2021), have also referred to the lack of research in this area and stated that "research around CLSC, which focuses on the retailer as a leader of the supply chain, which according to Yi et al. (2016) is called the Retailer-Oriented Closed-loop Supply Chain (ROCLSC), is still very limited." As we have mentioned earlier, the studies published in RL are carried out by the manufacturers in the RSC literature section. This fact has also been observed by Yuliawati et al. (2021), who stated that

Author(s)	Qualitative Research	Quantitative Research	Mixed Research	Literature Review	Conceptual Framework	Case Study	Survey	Observation	Interview	Documentary	Focus Group	Workshop	Data Generation	Model Development	Physical RRSC Network Design	Retailer Liquidity	Budget Management	Inventory Optimization	Solution Methodology Development	Franchise Retail Stores	Product Selection Decisions	Product Allocation Decisions	Facility Selection Decisions	Vendor Contract Parameter Optimization	Vendor Penalty Structure Management	Rule-of-Thumb Mechanism	Operational Decision Making	Tactical Decision Making	Strategic Decision Making
Tibben-Lembke and Rogers (2002)	x							x																					
Horvath et al. (2005)		x			x					x				x		x													
Chaves and Alcantra (2006)	x						x	x	x																				
Bernon and Cullen (2007)	x				x	x		x		x																			
Wojanowskiet al. (2007)		x											x	x									x				x		
Moise et al. (2008)	x				x			x								x													
Sonya Hsu et al. (2009)	x							x	x																				
Jack et al. (2010)		x					x																						
Bernon et al. (2011)	x							x		x	x																		
Das (2012)		x											x	x								x	x					x	
Bernon et al. (2013)	x				x	x		x																					
Olariu (2014)	x							x																					
Bajor and Babic (2014)	x				x		x	x		x																			
Bernon et al. (2016)			x					x	x	x		x																	
Dias and Braga (2016)			x					x	x	x																			
Beh et al. (2016)	x							x	x																				
Mostert et al. (2017)	x								x																				
Bernon et al. (2018)	x				x				x	x																			
Beitelspacher et al. (2018)			x				x	x		x																			
Panigrahi et al. (2018)	x				x				x																				
Dias et al. (2019)	x			x				x		x																			
Demajorovic et al. (2019)	x					x	x	x	x	x																			
Xu et al. (2019)	x																												
Frei et al. (2020)	x						x	x	x	x		x																	
Borba et al. (2020)	x			x	x			x		x																			
Gustafsson et al. (2021)			x			x	x		x	x																			
Yuliawati et al. (2021)		x											x	x													x		
Coskun and Hassini (2022a)		x											x	x			x		x		x	x					x		
Coskun and Hassini (2022b)		x											x	x	x			x	x	x	x	x	x				x	x	
Coskun and Hassini (2022c)		x											x	x					x					x	x	x			x

TABLE 2.1: RRSC Literature Review

> Unfortunately, most of the research has focused solely on Manufacturer-Oriented CLSC. The reason for this may be most of the previous research consider that the RL management activities are carried out by the manufacturer, not the retailer, which makes related research irrelevant. However, retailers as actors who are closer to consumers should be able to further investigate their role in the CLSC system.

We summarize the relevant literature in Table 2.1. We classify the published papers and identify gaps, some of which will be filled in by our current work.

2.4 Gaps of Retailer-Vendor Buy-Back Contracts in RRSC

Based on Guo et al. (2017)'s literature review of supply chain contracts in RL, there are 18 studies that considered buy-back contracts . The contracts were between manufacturer, re-manufacturer, third-party collector and retailer. Only Jeong (2012) and Matsui (2010)) considered contracts between a manufacturer and a retailer. These contracts were led by the the retailer. Jeong (2012) studied the collection and transmission of customers' product expectations and market demand information. Matsui (2010) explored the influence of the demand uncertainty. A recent work, by Guo et al. (2021), developed a selective buyback contract model where the retailer can forecast and determine the return quantity independent of the supplier by a put optioninstead of passively accepting the return quantity determined by the supplier at the ordering stage before the sales season, and the exercise price is not fixed. They have shown that contract can coordinate the supply chain by adjusting the option exercise price and giving the retailer more choice. The supplier receives risk compensation from the put options.

Based on the RSC of the retailer, we also consider a buy-back contract between the retailer and its vendors, however, unlike the existing literature that uses game theory, we use optimization models to identify the optimal contract parameters in a multi-layered penalty structure to re-negotiate the existing contracts with all of its vendors.

2.5 Main Thesis Contributions

2.5.1 Contributions on RRSC Modeling and Problem Objectives

Our research is the only study in RRSC literature where an independent retailer's RSC is modelled extensively. We contribute to the RRSC literature via modelling three different objectives. In the first model, we are the first to account for budget management, where a certain refund budget, a.k.a. Profit-Loss budget, is considered. The retailer has to spend the profit-loss budget to manage margin losses due to losses in income of returned items. In the second model, we are the first to consider a comprehensive inventory optimization model where find optimal stores' inventory levels via allocating the right products to the best available disposal option. In the thirdmodel, we are the first to study how a retailer should optimize its 'buy-back' contract parameters in a multi-layered penalty framework for product returns to its vendors. We use optimization models to identify the optimal return penalty and/or threshold parameters to re-negotiate the return contracts towards existing contract parameters.

2.5.2 Contributions on Levels in Decision Making

We contribute to the RRSC literature on all levels in the decision making.In our first and second models, we consider operational decisions via selection and allocation of products within the supply chain network. In the second model, we include tactical decision making via incorporating facility selection/location (activation of distribution/return centers). In the third model, we make strategic decision via identifying optimal return penalty and return threshold parameters for the retailer.

2.5.3 Contributions on Consideration of Franchisees and Their Objectives

W provide the first model in RRSC literature where we separate retail stores into different store type categories, namely company-owned and franchise stores, and their potentially conflicting objectives related to optimizing their inventory. The RSC network we consider incorporates individual franchises as a new network member. We manage the franchise store's individual objectives and this might be conflicting with the retailer's overall objective. We incorporate this conflict in our problem and resolve it via introducing inventory transparency among franchise stores through several constraints that manage proportional inventory returns.

2.5.4 Contributions on Solution Methodology

We contribute to the literature by developing heuristic algorithms to solve selection and allocation decisions in RRSC. In the first and second problems, we develop constructive heuristic algorithms, that make use of the problem structure, to solve budget management and inventory optimization problems, respectively. We show that they lead to close to optimal solutions with reasonable solution times. The heuristics we have developed can be transformed to similar selection and allocation decision making processes in other similar budget balancing and network allocation problems. In the third problem, we propose a decomposition approach to identify under which conditions there exists a better penalty and/or threshold structure for contract re-negotiation, and how we can calculate the optimal penalty structure for the retailer. Our approach provides managers with a plug-and-play framework that can be used to negotiate similar kind of contracts without the need to solve complex optimization problems.

Chapter 3

Retail Reverse Supply Chain Optimization under Profit-Loss Budgetary Limitation

3.1 Introduction

In this chapter, we consider a RRSC budget planning of an independent retailer where its supply chain network consists of multiple vendors, a warehouse, and multiple stores. Some portion of the retailer's inventory does not sell as expected and the retailer needs to salvage them optimally in order to refill the space with better products and hold more effective inventory that has a higher sell through rate to increase sales and hence profit. The goal of the retailer is to decide which products should be chosen from stores to be relocated within the network in order to acquire a healthier inventory level against a given 'Profit-loss' budget that should be consumed.

A profit-Loss budget is a budget that is calculated every year by the accountants of the company in order to reverse the margins that are lost due to non-selling of the purchased products by the retailer. During a fiscal year, many products are purchased from vendors hoping that they will be sold to customers, however, some portion of these purchases ends up not selling as well, or not selling at all, for various reasons such as customer desires, product unpopularity, competitive products in the market, price and/or quality of the product, etc. Over time, these products accumulate in store inventories and fill up scarce space in the shelves and/or warehouses. In order to relieve this limited and highly productive space to products that have a higher chance of sale, the retailer needs to salvage some or all of this ineffective inventory optimally via a reverse logistics process from time to time. Every year the retailer forecasts a budget (using expected effective and ineffective inventory levels, historical sales & non-sales of the purchased products, and historical inventory removal amounts) that accounts for this situation and some portion of the ineffective inventory has to be removed from store shelves/inventory where the total expenditure of the RSC process has to fit within this pre-calculated budget. The basic idea of the 'profit-loss' budget is to account for and forecast the margin reversal cost of upcoming product returns that will be disposed of in the current year. Once it is calculated, the profit-loss budget can then be used to remove the profits of the 'could-have-been sold' products from company books (these are the products that are already purchased within the year and presumed to be sold but not actually sold due to many reasons as mentioned above, therefore presumed profit margins on the books should be reversed). Therefore, this budget gets calculated by the financial team of the company for financial planning and most importantly taxation reasons. We note that the retailer has to file taxes before customer sales are realized. The profit-loss budget is therefore based on customer sales predictions.

During a fiscal year, many products get sold to stores by the retailer and their possible profit gains are incorporated and accounted for as gains. When some of these products are not sold to the end customer by the stores, and some portion of this inventory is removed from store inventories (when the retailer wants to make some inventory clearance via returns and relocations), the margins of the disposed inventory that are entered in the books has to be reversed, meaning that retailer's 'profit-loss' (margin recovery costs, unrecoverable costs, vendor return penalties, and similar costs of doing business) should be calculated and then the books have to be readjusted. Since identifying the products that are not sold, disposing some of them, reversing the margins of the returned ones, fixing those entries on books by reversing their margins (with additional return related costs), and then adjusting the taxation side can be a very complex and time consuming act, the retailer uses forecasts to estimate this amount using historical data and expects the RSC activity to fulfill this obligation of consuming the forecasted budget instead of reversing the margins of the products that would most ideally be disposed.

As a result, when it is time to clear some portion of the inventory via a reverse supply process, the retailer has to use a budget that is predetermined/forecasted by the finance department. The goal is to limit the total profit losses due to these returns according to the pre-determined budget, which we refer to as the 'Profit-loss' budget.

In a real retail environment, if some of the products at the stores are not selling, the retailer identifies those products as ineffective inventory and would choose a portion of these ineffective products to be relocated within the network, at other stores where there is a demand for it, or sent back to its vendor. These products (at stores which are deemed as ineffective) move upward in the supply chain network until they reach to their last destination which might be the products' original vendors or the warehouse (and then a store which has demand for those products).

If an ineffective Store-product can be returned or relocated to a warehouse or a vendor. It will be returned to a Warehouse, if the product has warehouse demand. Warehouse demand refers to aggregate demand from stores that are served from that warehouse. . If any product has a purchase order/replenishment entry (most probably from the vendor) in the regular forward supply chain system, the RSC process will consider these purchase orders as Warehouse demand because it is preferable to satisfy that demand from an internal source - in this case a store which has that product in their current inventory and deemed as ineffective - as opposed to ordering them from its original source. A product is returned to a Vendor to exhaust the 'Vendor Funds' if the product is under the warranty of returnable products and the retailer has related vendor funds to recover some portion of the product's cost (COG). In general, a retailer has vendor funds from a vendor that they regularly purchases products from. The amount of vendor funds are negotiated through a purchasing agreement between the retailer and the vendor and calculated as a certain portion of the sales in a certain period of time. For example, if the retailer purchases $500,000 worth of products from a vendor in a year, the return agreement might be negotiated so that the retailer can return 10% of the yearly sales with a 25% penalty next year, i.e., the retailer can return up to $50,000 worth of products and would pay up to $12,500 in penalties.

3.2 Problem Definition

To address the challenges of a profit-loss based reverse supply chain, we propose a model to aid the retailer in deciding on which store products should be returned and whether they should be returned to a warehouse or to a vendor. Thus, our model involves both a selection, and an allocation tasks. In this respect, our RRSC network is the only model in the literature addresses a RRSC environment by considering product selection as a decision variable. We consider product selection as a decision variable and do not pull all of the pre-determined ineffective inventory from stores because only a certain portion of the ineffective inventory has to be disposed of based company inventory holding policy, related costs, and budgets available to keep inventory on certain levels.

The retailer's goal is to efficiently use scare shelf space for products that are likely to sell.Since continuously pulling all the pre-determined ineffective inventory is not a practical option, retail companies clear their ineffective inventory periodically. They may fix the amount of cleared ineffective inventory, say \$1M worth of products annually, or keep a fixed percentage of effective inventory, say 85% at the start of each year. A more efficient policy is to find the optimal selection and allocation in order to minimize the RRSC costs, this is the objective of our model in this chapter.

In our RRSC budget optimization model, the retailer decides to clear certain amounts of inventory at certain points in time, such as at the end of each fiscal year or at the beginning of each quarter. This policy defines the basis of our problem since it entails the retailer to clear and dispose of a certain amount of its ineffective inventory. The retailer is obliged to dispose of some of its ineffective inventory and has to reverse the margins of the unsold products which were reported, in their previous year's financial and tax statements, as profits.

An important input to our model is the profit-loss, which is predetermined by the retailers finance department. The retailer's goals is to use this budget as efficiently as possible via the return of ineffective products in the inventory. The profit-loss budget is a budget that has to be spent as much as possible, against the profit margins that will be lost due to the relocation of products that have previously been entered in the retailers budget as profit generating sales.

In summary, our goal is to select a set of products' worth to be returned from a predefined amount (in terms of store \$ value) of the ineffective store products.The chosen ineffective store-products need to be routed to a warehouse or vendors based on factors such as demand at the warehouse, capacity of the warehouse, available vendor funds, and products' profit margins. The objective is to minimize all costs related to pulling inventory from stores and relocating them, internally or externally subject to using the available Profit-Loss Budget as much as possible.

3.3 Model

To solve the problem define in Section 2, we propose an mixed integer linear program (MILP) where the objective is to minimize all the related RRSC costs and use the available Profit-Loss Budget as much as possible. In the next section we define the optimization model that is suggested to find optimal usage of the profit-loss budget via allocating returned products. The model is formulated in (3.1)–(3.12).

$$\min \sum_{p=1}^{P} [(vr_p - wh_p)SP_p - 2vr_pCOG_p + (vr_p + wh_p)LC_p] + \sum_{v=1}^{V} VPF_v\, vf_v \tag{3.1}$$

$$\text{s.t.} \sum_{s=1}^{S} (1 - rt_{s,p})Q_{s,p} = vr_p + wh_p \quad \forall\, p \in P \tag{3.2}$$

$$\sum_{p=1}^{P} wh_p\, COG_p \leq WRA \tag{3.3}$$

$$wh_p \leq WDC_p \quad \forall\, p \in P \tag{3.4}$$

$$\sum_{p \in P_v} vr_p\, COG_p = vf_v \quad \forall\, v \in V \tag{3.5}$$

$$vf_v \leq VF_v \quad \forall\, v \in V \tag{3.6}$$

$$\sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}SP_pQ_{s,p} \geq TSRA \tag{3.7}$$

$$\left(\sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}(SP_p - LC_p)Q_{s,p}\right) + \left(\sum_{p=1}^{P} vr_p(LC_p - COG_p)\right) + \left(\sum_{v=1}^{V} VPF_v\, vf_v\right) \leq PLB \tag{3.8}$$

$$\left(\sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}(SP_p - LC_p)Q_{s,p}\right) + \left(\sum_{p=1}^{P} vr_p(LC_p - COG_p)\right) + \left(\sum_{v=1}^{V} VPF_v\, vf_v\right) \geq PLB - N \tag{3.9}$$

$$rt_{s,p} \in \{0,1\} \quad \forall\, s \in S,\ p \in P \tag{3.10}$$

$$vr_p,\ wh_p \in \mathbb{N} \quad \forall\, p \in P \tag{3.11}$$

$$vf_v \in \mathbb{R} \quad \forall\, v \in V \tag{3.12}$$

The objective function (3.1) represents the total costs of losses, penalties and the cost of gains that are redeemed when a product is returned back to another store or warehouse. Constraint set (3.2) ensures that for each product, the total number of items that are chosen to be pulled from stores is equal to the number of items that will be distributed to warehouses and vendors. Constraint (3.3) ensures that the maximum return amount (in terms of $ value) to the warehouse should be less than the capacity of that warehouse. Constraint set (3.4) ensures that for each product, the total number of items that are returned to the warehouse can not exceed the warehouse capacity (in terms of quantity) for that product Constraint set (3.5) ensures the value of the products sent to vendor is equal to the total vendor funds that should be used. Constraint set (3.6) ensures that the total vendor funds that should be used in order to return products can not exceed the available vendor dollars of that vendor. Constraint (3.7) ensures that a certain amount of ineffective store-products should be removed from all of the stores' inventories. Constraints (3.8) and (3.9) enforce the profit-loss budget. Constraint sets (3.10)–(3.12) define the set of binary, integer and continuous variables, respectively.

3.4 A Heuristic Algorithm

Our test have shown that commercial solvers run out of memory and storage space when solving problem (3.1)–(3.12). From our experience with the industrial partner, we have also learned that a commercial solver license can add significant operating costs. This has convinced us of the need to develop a heuristic algorithm. The heuristic pseudo code is shown in Subroutines 1, where an initial allocation is determined, and Subroutine 2, where the budget allocation is determined. We also illustrate the two subroutines graphically in *Figures 3.1* and *3.2*. A more detailed description of the heuristic is included in Appendix A1.

Subroutine 1: Heuristic Allocation - Initialization

Result: Write here the result

$L_1 = \{(s,p)$ for $s \in S, p \in P$ s.t. $Q_{s,p} > 0$ and $WDC_p > 0\}$ sorted by LC_p/SP_p;

$U = newSet()$;

$wh_p = 0,\ \forall p \in P$;

while $\mid L_1 \mid > 0$ *and* $\sum_{p \in P} wh_p \times COG_p < WRA$ **do**

  $(s,p) =$ pop top element from L_1;

  add (s,p) to U;

  if $\sum_{p' \in P} wh_{p'} \times COG_{p'} \leq WRA - Q_{s,p} \times COG_p$ *and* $Q_{s,p} + wh_p \leq WDC_p$ **then**

    $wh_p + = Q_{s,p}$;

  end

end

$WarehouseReturn = \sum_{p \in P} wh_p \times SP_p$;

$VendorReturn = TSRA - WarehouseReturn$;

$L_2 = \{(s,p,V_p)$ for $s \in S, p \in P$ s.t. $Q_{s,p} > 0$ and $(s,p) \notin U\}$ sorted by $-COG_p\ VPF_p/SP_p$;

$vr_p = 0,\ \forall p \in P$;

$vf_v = 0,\ \forall v \in V$;

while $\mid L_2 \mid > 0$ *and* $\sum_{p \in P} vr_p \times SP_p < VendorReturn$ **do**

  $(s,p,v) =$ pop top element from L_2;

  if $\sum_{p' \in P} vr_{p'} \times SP_{p'} \leq VendorReturn - Q_{s,p} \times SP_p$ *and* $(Q_{s,p} + vr_p) \times COG_p \leq VF_v$ **then**

    $vr_p + = Q_{s,p}$;

    $vf_v + = Q_{s,p} \times COG_p$;

  end

end

$Budget =$
$\sum_{p \in P} vr_p(SP_p - COG_p) + \sum_{p \in P} vr_p(LC_p - COG_p) + \sum_{v \in V} VPF_v\ vf_v + \sum_{p \in P} wh_p\ (SP_p - LC_p)$;

$VendorBudget \leftarrow Budget - WarehouseReturn$;

Subroutine 2: Heuristic Allocation for Budget Optimization

Result: Write here the result
if $Budget < PLB$ **then**
  $TotInv = \sum_{p\in P} vr_p \times SP_p + \sum_{p\in P} wh_p \times SP_p$;
  $ReturnCostRatio = VendorBudget / \sum_{s\in S} \sum_{p\in P} vr_p \times SP_p \times Q_{s,p}$;
  $ExpectedReturnCostRatio = (PLB - VendorReturnCost)/(TSRA - \sum_{p\in p} wh_p \times SP_p)$;
  $\forall p \in P : vr_p = 0$;
  $Picked = newSet()$;
  $p_s = p \in P$ s.t. $COG_p/SP_p \le ExpectedReturnCostRatio$ and
   $\forall p_2 \in P, COG_{p2}/SP_{p2} \le ExpectedReturnCostRatio \Rightarrow COG_{p2}/P_{p2} < COG_p/SP_p$;
  $StartCost = COG_{p_s}/SP_{p_s}$;
  while $\sum_{p\in P} vr_p \times SP_p < TSRA - WarehouseReturn$ **do**
    (s_u, p_u) = any $s_u \in S, p_u \in P$ s.t. $(s_u, p_u) \notin Picked$ and $COG_{p_u}/SP_{p_u} \ge StartCost$ and
     $\forall p_2 \in P, COG_{p2}/SP_{p2} \ge StartCost \Rightarrow COG_{p2}/SP_{p2} \ge COG_{p_u}/SP_{p_u}$;
    Add (s_u, p_u) to $Picked$;
    (s_d, p_d) = any $s_d \in S, p_d \in P$ s.t. $(s_d, p_d) \notin Picked$ and $COG_{p_d}/SP_{p_d} \le StartCost$ and
     $\forall p_2 \in P, COG_{p2}/SP_{p2} \le StartCost \Rightarrow COG_{p2}/SP_{p2} \le COG_{p_d}/SP_{p_d}$;
    $vr_{p_u} = rt_{s_u,p_u}$;
    Add (s_d, p_d) to $Picked$;
    $vr_{p_d} = rt_{s_d,p_d}$;
  end
else
  $P' = \{(s,p)$ for $s \in S, p \in P$ s.t. $vr_i = 0\}$ sorted by $Q_{s,p} \times SP_p - COG_p$;
  $P'' = \{(s,p)$ for $s \in S, p \in P$ s.t. $WDC_p > 0\}$ sorted by $Q_{s,p} \times SP_p - LC_p$;
  $WhFix = \emptyset$;
  while $\sum_{p\in P} vr_p \times (SP_p - COG_p) + WPF_v \times vf_v + wh_p \times (SP_p - LC_p) \le PLB$ *and* $|P''| > 0$ **do**
    (s,p) = pop top item in P'';
    $wh_p = 0$;
    $Q_{s,p} \times COG_p$;
    (s', p') = pop minimum item in P'' s.t. $Q_{s,p} \times COG_p \le Q_{s',p'} \times COG_{p'}$;
    $wh_{p'} = 1$;
    Add p' to WhFix;
    $diff = Q_{s,p} \times SP_p - Q_{s',p'} \times SP_{p'}$;
    $pick = 0$;
    while $pick < diff$ **do**
      (s'', p'') = pop minimum item from I' s.t. $Q_{s'',p''} \times SP_{p''} \ge diff$;
      $pick+ = Q_{s'',p''} \times SP_{p''}$;
      $vr_p = Q_{s'',p''}$;
    end
  end
  if $|P''| = 0$ **then**
    $wh_p = 0$ for all $p \in P$ except in $WhFix$;
    while $\sum_{p\in P} vr_p \times (SP_p - COG_p) + WPF_v \times vf_v + wh_p \times (SP_p - LC_p) \le PLB$ *and* $|P'| > 0$
     do
      (s', p') = pop minimum item from P';
      $vr_{p'} = Q_{s',p'}$;
    end
    if $|P'| = 0$ **then**
      Return infeasible;
    end
  end
  Return current assignment;
end

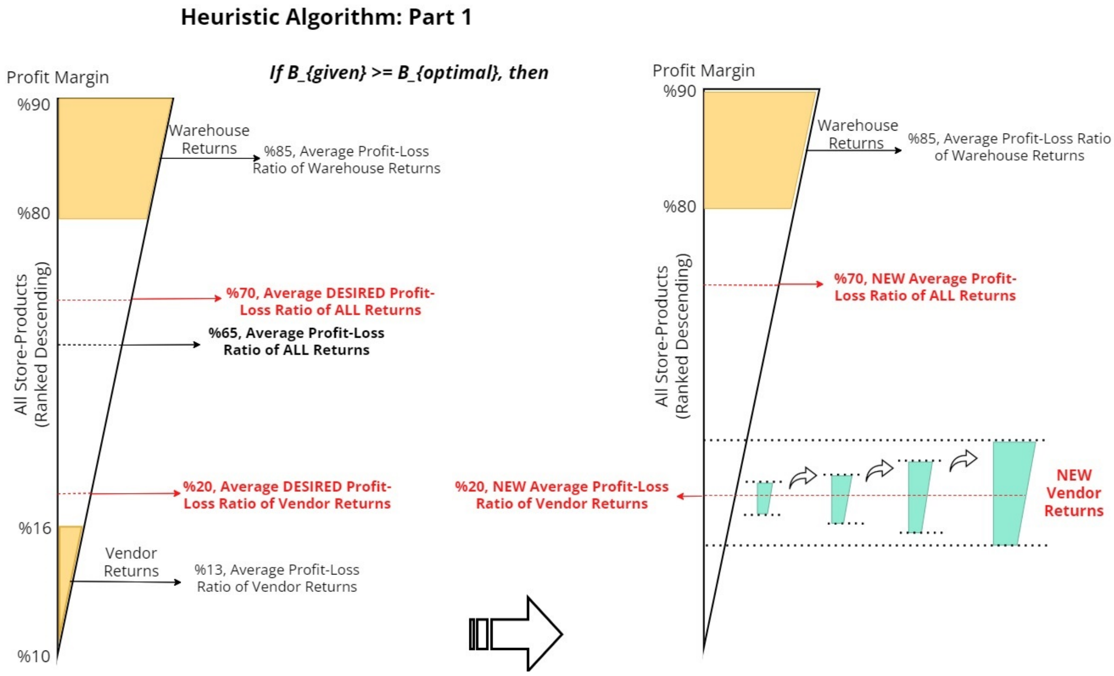

FIGURE 3.1: Heuristic Algorithm for Budget Optimization, Part 1

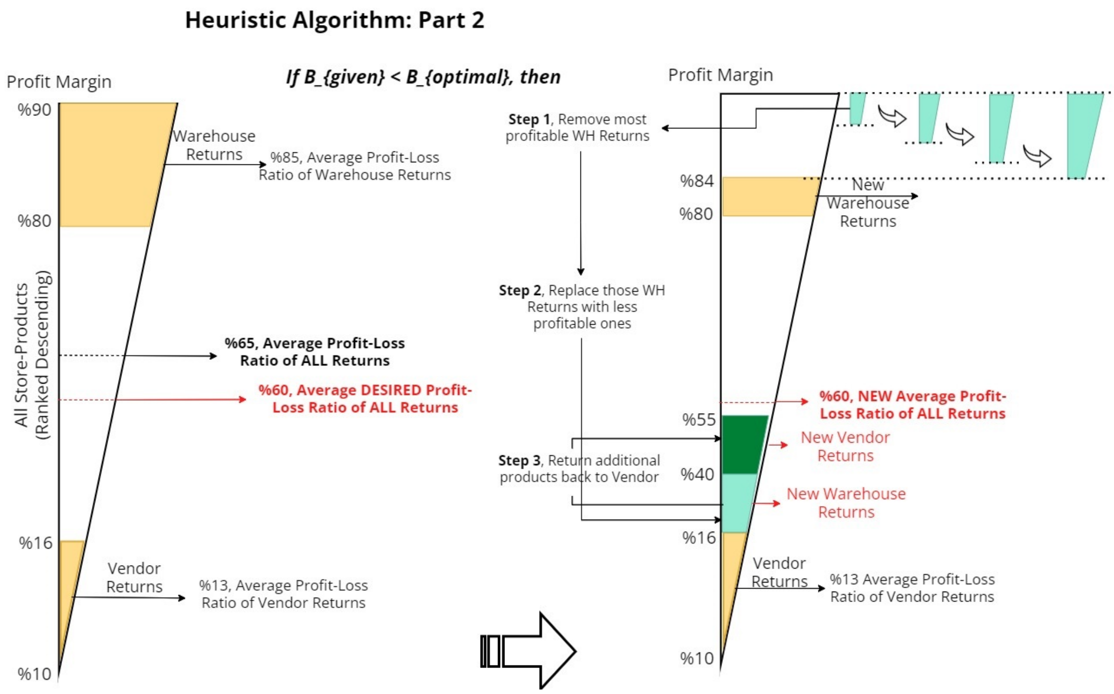

FIGURE 3.2: Heuristic Algorithm for Budget Optimization, Part 2

3.5 Test Problems, Computations and Numerical Analysis

In this section, we present our findings about the test problems we have solved using CPLEX and compare the results and performance of our heuristic to the results of the exact solver.

3.5.1 Test Cases and Data Generation

Our data generation was motivated from real industrial data. For every test problem, we have considered 1000 unique products, 500 unique stores, 20 unique vendors and a warehouse in a RRSC network. Each product is assigned to a random vendor. Each parameter of a product is generated with random data that fits a uniform distribution between certain bounds (lower and upper limit) that are consistent with values of the related parameters; COG, landed cost, and store purchase price. Warehouse demand for every unique product is also randomly generated with a uniform distribution. Overall warehouse and vendor parameters such as capacity and available vendor funds, respectively, are chosen in way that limits the returns of the ineffective inventory and hence allows returns to both parties. A range of penalty fees/rates are generated for the 20 vendors that allows and limits returns to a specific vendor.

All the problems are generated with relatively large size, 250,000 Ineffective Store-products. In total, 20 problems are generated, 10 unique test problems are created with 2 distinct scenarios; one with a high Profit-Loss Budget and one with low Profit-Loss Budget. As can be observed from *Table 3.1*, the 'odd' number test cases represent the high Profit-Loss Budget scenario of the problem, and the 'even' number test cases represent the low Profit-Loss Budget scenario of the problem.

Since these 250,000 store-product combinations are randomly generated where each product is assigned to a random vendor and all the parameters of the products; COG, landed cost, store purchase price values are randomly generated between a certain range, every test case's ineffective store-product, their quantity in each location, products' unrecoverable percentage, and profit margin differ from the other test cases and we have wide range of problems in terms of profitability, potential return amounts to the warehouse and vendors.

The RRSC budget optimization model is a mixed integer linear program (MILP) and is developed and coded with an exact solver software, IBM ILOG CPLEX Optimization Studio 12.8, on a PC with an Intel Core i7-8550U CPU @ 1.8 GHz, 4 Cores, 8 Logical Processors and 20 GB of RAM.

As we can observe from *Table 3.1*, CPLEX can solve the test problems to optimality within a very short amount of time even with huge problem sizes such as 250,000 store-product combinations that generates 750,000 decision variables and 2044 constraints. The total data intake process of CPLEX can range between 3-4 minutes, however, when the data intake process is done, CPLEX can solve the problems between 1-30 seconds depending on the complexity of data/problem, available total store removal amount, warehouse capacity, total available vendor funds, and most importantly profit-loss budget given to problem.

3.5.2 Comparison of CPLEX and Heuristic

In our test case problems, these large size problems are generated and solved both using CPLEX and our heuristic algorithm. However, the ingestion of these large size data by the tools took most of the solution time for the given problems. As can be observed from *Table 3.1*, the ingestion of the data took almost 3-4 minutes by the CPLEX and solution times of the problem ranged between 1-30 seconds. Therefore, we do not really worry about the solution times and their comparison of the algorithms since the ingestion of the data took most of the solution time of the given problems. Even though the solution times of our heuristic might take a little bit longer time than CPLEX in some scenarios, we can find solutions close to optimality within a very short amount of time. Our heuristic can solve problems within 4-5 seconds, on average, for the problem sizes that are given, 250,000 store-product combinations. Basically, our heuristic can calculate 250,000 row of data points with calculated profit/cost parameters within seconds, then rank them based on the condition and then extract the best possible store-products from the dataset. This way a close to optimal solution can be generated within a very short amount of time for even very large size problem such as problems with several million store-products.

Even though the need of using a heuristic to solve this problem might seem unreasonable because of the short solution times acquired by the state-of-the-art solver, the intention to develop a heuristic results from the curiosity of understanding the inner structure of optimal solutions so as to develop more sophisticated heuristics to solve not only this problem, but also more complex problems in RRSC (such as problem defined in *Chapter 4*) or in related/other fields (such as forward supply chains, logistics, finance, portfolio management, etc.) that have similar budgetary limitation issues. The cost of using state-of-the art solvers to solve this problem is also very high, especially for smaller companies. This might also be one of the main reasons why we need to develop similar heuristics to solve such problems.

As can be observed from *Table 3.1*, when a given problem is solved without any budgetary constraints, CPLEX can find an optimal solution mostly within 1-2 seconds. However, when budgetary constraints are applied, the solution times increase with the complexity of budgetary boundaries. When the test problems are solved using over-budget constraints, the solution times does not increase significantly, however, when the test problems are solved using under-budget constraint, the solution times might increase significantly. We have investigated the underlying reasons and details will be included in the next section. Our heuristic was developed based on this investigation and insights.

Test Case	# of Decision Variables	# of Binary Variables	# of Integer Variables	# of Constraints	Solution Methodology Used	Optimality Range	Total Time, Data Ingestion + Solution Time (minutes)	Solution Time (seconds)	TOTAL COST of REVERSE SUPPLY CHAIN ACTIVITY
Test 01	750,000	250,000	500,000	2,044	CPLEX	0.06%	4:28	0.98	$ 5,124,307
	750,000	250,000	500,000	2,044	Heuristic	0.15%		4.24	$ 5,128,942
Test 02	750,000	250,000	500,000	2,044	CPLEX	0.00%	5:32	2.21	$ 4,696,156
	750,000	250,000	500,000	2,044	Heuristic	0.67%		5.22	$ 4,727,950
Test 03	750,000	250,000	500,000	2,044	CPLEX	0.01%	4:26	0.97	$ 3,650,263
	750,000	250,000	500,000	2,044	Heuristic	0.14%		4.42	$ 3,655,202
Test 04	750,000	250,000	500,000	2,044	CPLEX	0.19%	6:00	33.70	$ 3,921,456
	750,000	250,000	500,000	2,044	Heuristic	0.71%		5.55	$ 3,941,892
Test 05	750,000	250,000	500,000	2,044	CPLEX	0.03%	4:39	5.01	$ 6,036,595
	750,000	250,000	500,000	2,044	Heuristic	0.05%		5.39	$ 6,037,702
Test 06	750,000	250,000	500,000	2,044	CPLEX	0.20%	5:20	30.81	$ 5,860,545
	750,000	250,000	500,000	2,044	Heuristic	0.20%		6.48	$ 5,860,036
Test 07	750,000	250,000	500,000	2,044	CPLEX	0.00%	4:46	1.45	$ 5,144,050
	750,000	250,000	500,000	2,044	Heuristic	0.07%		2.72	$ 5,147,445
Test 08	750,000	250,000	500,000	2,044	CPLEX	0.32%	4:55	21.64	$ 5,437,146
	750,000	250,000	500,000	2,044	Heuristic	1.30%		2.72	$ 5,492,711
Test 09	750,000	250,000	500,000	2,044	CPLEX	0.01%	4:43	1.34	$ 6,406,204
	750,000	250,000	500,000	2,044	Heuristic	0.04%		4.03	$ 6,407,868
Test 10	750,000	250,000	500,000	2,044	CPLEX	0.21%	5:11	24.02	$ 5,806,250
	750,000	250,000	500,000	2,044	Heuristic	0.64%		4.99	$ 5,831,153
Test 11	750,000	250,000	500,000	2,044	CPLEX	0.13%	4:35	15.91	$ 6,257,732
	750,000	250,000	500,000	2,044	Heuristic	0.23%		6.81	$ 6,264,013
Test 12	750,000	250,000	500,000	2,044	CPLEX	0.65%	4:45	22.84	$ 6,296,726
	750,000	250,000	500,000	2,044	Heuristic	1.17%		7.85	$ 6,329,751
Test 13	750,000	250,000	500,000	2,044	CPLEX	0.00%	4:25	2.48	$ 6,112,795
	750,000	250,000	500,000	2,044	Heuristic	0.10%		4.82	$ 6,118,763
Test 14	750,000	250,000	500,000	2,044	CPLEX	0.54%	5:02	30.26	$ 5,925,310
	750,000	250,000	500,000	2,044	Heuristic	0.64%		5.72	$ 5,931,081
Test 15	750,000	250,000	500,000	2,044	CPLEX	0.03%	4:24	0.98	$ 3,467,624
	750,000	250,000	500,000	2,044	Heuristic	0.18%		1.67	$ 3,472,921
Test 16	750,000	250,000	500,000	2,044	CPLEX	0.01%	4:56	1.45	$ 3,574,916
	750,000	250,000	500,000	2,044	Heuristic	0.38%		1.67	$ 3,588,396
Test 17	750,000	250,000	500,000	2,044	CPLEX	0.01%	4:23	0.98	$ 7,398,248
	750,000	250,000	500,000	2,044	Heuristic	0.06%		4.15	$ 7,402,199
Test 18	750,000	250,000	500,000	2,044	CPLEX	0.47%	4:59	25.27	$ 7,681,737
	750,000	250,000	500,000	2,044	Heuristic	0.71%		5.71	$ 7,700,564
Test 19	750,000	250,000	500,000	2,044	CPLEX	0.01%	4:26	2.65	$ 5,277,186
	750,000	250,000	500,000	2,044	Heuristic	0.10%		4.38	$ 5,281,754
Test 20	750,000	250,000	500,000	2,044	CPLEX	0.01%	4:34	2.76	$ 5,966,628
	750,000	250,000	500,000	2,044	Heuristic	0.45%		5.02	$ 5,993,424

TABLE 3.1: Test Problem Results for Budget Optimization Model

3.6 Results and Insights

When over-budget constraints are applied to a given problem, compared to the case of optimized inventory where best profitable store-product are allocated to the their new location, profit-loss budget is under spent. In order to resolve this issue, the optimal solution needs to identify a solution where more profit-loss budget can be spent while compromising inventory optimality. Since more budget is given to the problem than when the inventory is optimized, the solution needs to identify less profitable products and need to spend more of the potential profit-loss budget that is given. Since most profitable products are already relocated to other stores where there is demand in the case of the optimal inventory problem, similarly most profitable products will also be relocated to other stores in the case of the over-budget constraint problem. However, more profitable products need to be send back to their vendor which is not in line with the optimal inventory problem. We therefore developed the first part of our heuristic based on this fact and identified 'good' solutions close to optimality under over-budgetary boundaries.

When under-budget constraints are applied to a given problem, the optimal solution needs to spend the budget more wisely since the profit-loss budget is tighter than optimal inventory solution results. When the problem is solved without any budgetary constraints, the least profitable products are send backed to their vendor (with the use of minimal penalties). This phenomenon is in line with optimally spending the potential profit-loss budget. Therefore, optimal solution relies (mostly) on the efficient usage of the warehouse returns because warehouse returns use more of the available budget since it identifies most profitable products and relocates them. This causes spending most of the available budget and therefore an optimal solution needs to be identified where less profitable products should be relocated to warehouses, which is also not in line with optimal inventory problem. In certain scenarios where budgetary constraints are tighter, tightening the margins of warehouse returns might not be sufficient and an optimal solution might exist where more vendor returns (with less profitable products) are needed with less profitable products. We have also observed this phenomenon in our test problems where more products (wither lower profits margins than warehouse returned products, higher profit margin products than vendor returned products) are returned backed to their vendor. Since the least profitable products are already returned backed to their vendor, the additional vendor returned products are more profitable than products already returned to vendors but less profitable than warehouse returned products. We therefore also developed the second part of our heuristic based on these facts and identified ‘good’ solutions close to optimality for under-budgetary boundaries.

Chapter 4

Retailer's Inventory Optimization via Reverse Logistics

4.1 Problem Definition

In this chapter, we consider a realistic RSC network of an 'independent' retailer. We model a comprehensive RRSC network where there are multiple stores under different store types, namely, Company Owned and Franchise stores, multiple warehouses, multiple distribution centers, multiple vendors and liquidators. The objective of the retailer is to optimize inventory and minimize costs for relocating these excessive inventory within the network for both its own company owned stores and its individual franchise stores.

Each company owned and franchise store holds a certain amount of products that are not selling (or customer-returned, damaged, faulty, end-of-life) which is considered as ineffective inventoryHowever, there is a certain amount of inventory that the retail company stores and individual franchise stores can get rid of based on certain company rules, business strategies and contracts with the individual franchise stores. For the retail company, this amount is based on how much effective inventory and ineffective inventory they hold among all of the company owned stores. The retail company's goal is to hold a certain effective inventory level at the end of this process in order to keep the overall inventory at a certain healthy level. For the franchise stores their total ineffective inventory removal amount is calculated based on a ratio of the overall purchases of a prior year/term from the retail company. Hence it is a known and calculated amount for the overall franchise stores. Since the franchise total ineffective inventory removal amount is known and individual franchise stores have their own idea about how much ineffective inventory they want to get rid of, there is a conflict between the retail company and its individual franchise stores, and more importantly there is a bigger conflict among all of the individual franchise stores. The retailer's objective is to minimize overall costs both for company owned stores and its franchise store returns. However, the individual franchise stores' objective is to get rid of as many products as possible in order to minimize its own inventory costs. These different objectives create a conflict between the retailer and its franchise stores because each franchise store's removal amount can not be determined by them, as otherwise they want to get rid of whatever is ineffective in their inventory. Since franchise stores also compete among each other, this leads to a conflict among all members of the network. We deal with this issue by introducing an *inventory transparency mechanism* among franchise stores.

The transparency mechanism among franchise stores is enacted by the retail company declaring to all franchise stores how much total inventory they are holding compared the total stock among all the franchise stores. Based on each store's inventory level, we propose a relative inventory pull amount based on the relative inventory in each store and the total franchise store pull amount (e.g. Franchise store A will get an inventory pull amount of Total Franchise Stores' Pull Amount * Total Inventory Amount of Franchise Store A / Overall Inventory Amount of All Franchise Stores). This way each store gets an inventory removal amount based on how much inventory they hold compared to total franchise inventory amount which would solve the conflict between the retailer and its franchise stores, and the conflict among all franchise stores since every store gets an amount that is proportional to their overall contribution the retailer's franchise network. In general, this proposal will reward to out-performers and penalize the under-performers. Between two similar franchise stores that hold almost same amount of total inventory, the one with less ineffective inventory gets same return amount as the one with larger ineffective inventory and hence the one that holds less ineffective inventory gets a larger portion of its ineffective returns as a returnable amount and the one hold more ineffective inventory get a lesser portion of its ineffective return as a returnable amount. Since this is a desired outcome, the mechanism favors the out-performer. However, the benefits of this approach can be observed much more closely and clearly between stores that hold large but keep healthy inventory levels vs. stores hold smaller but unhealthy inventory levels. Since smaller stores with high ineffective inventory rate will be able to get rid of 'relatively' less products from its inventory, the risk penalization for under-performers can easily be observed. This approach will push under-performers to get them better at how to manage their inventory wisely (via discounts, clearance sales, order better products to sell, etc.) since inventory clean-up process does not favour them compared similar size stores with healthy inventory levels.

Our RSC model is also a network optimization problem for the retailer since we introduce a tactical (might also be considered strategic) decision making, in addition to operational decision making, of which RCs should be activated during the RSC activity. This decision is a big part of our problem since we model our problem considering physical structure of the actual network using specific transportation routes between stores to RCs, RCs to warehouses, RCs to vendors, and among returns centers. Based on a specific network configuration, when the potential routes are altered then the related costs get affected. We want to identify which RCs should be active with a given store supply, warehouse capacity and demand, potential transportation routes and processing fees at each RC compared to activation of each location.

In our problem, we have warehouses that have demand for certain products (which is the cumulative store demand that the warehouse serves to) which might be supplied from other stores that have unwanted inventory of those products (a.k.a stores' ineffective inventory). Each warehouse has a capacity limitation in terms of physical size and financial amount. A product can be sent back to its vendor, however, vendor has certain limit, a.k.a vendor funds, for receiving returned products. The available vendor funds are multi-layered meaning that vendor charges penalty for each product it receives and over certain thresholds the penalty rate increases. In the real retail world, the retailer purchases products from vendors and based on the purchases it makes during the year, the retailer negotiates return policy for the unsold products. In general, the available vendor funds are calculated through a portion of the previous terms' sales. Then the vendor decides how much of this available funds can be used through via multi-level penalties and multi-layer breaks of the vendor funds; the more you return, the more penalty you pay per $. We consider this important phenomenon and model it in our problem. Products can also be sent back to their vendor without using the available vendor funds if they have deposit values.

There are other options for products to be disposed such as liquidating or getting rid of them at their current location by throwing them into the trash. These are higher costly and the least desired options for a product to be disposed since most or all product value can be lost.

In order to visualize the above defined complex RRSC structure, please see *Figure 4.1* to observe how the items of a product in stores can travel within the network. Based on the *Figure 4.1*, you can observe that for any items of the chosen store-product, if they are be selected to participate to the RL activity, they have to be returned to its related RC. At this location they are received by the employees of the RC to be sorted, quality checked and identified to be sent to their next location which might be warehouses under its supply chain that have a demand for the product, fellow RCs and hence their warehouses that have demand have a demand for the product, back to the product's vendor for vendor return or to the liquidators. The RRSC network is very strict in terms of product movement where a product has to follow the exact path (as in *Figure 4.1*) to reach its final destination and any other route is not allowed by the rigid network design. When network design changes based on activating / de-activating one or more RCs, the routes get updated with a pre-determined routing structure but that does not change the rigidness of the network.

Since we deal with a problem where items move in the network where there any many echelons/layers, it only makes sense to incorporate tactical/strategic decision making process of which RCs should be activated / de-activated to participate into the RL activity. Overall optimization of upstream product flow can only be minimized by globally minimizing the all potential costs via adding design structure of the network, not only by processing best operational actions. For each specific configuration of the RSC network, the potential routes might significantly change for a product be located in its new location which will eventually affect all the cost structure on the network. Our model considers these effects and is designed to handle such complexities.

Based on all the above issues that need to be resolved, we can model this complex problem as an optimization program, a Mixed Integer Linear Program (MILP), with the objective of minimizing all the related (reverse) supply chain activity costs for relocating products within the network. The triggering mechanism to initiate this RL stem from the fact that retail company needs to pull certain amount of inventory from all of its stores, both company owned and franchise, to get to a health inventory level decided by pre-determined business rules and contracts with its franchise stores.

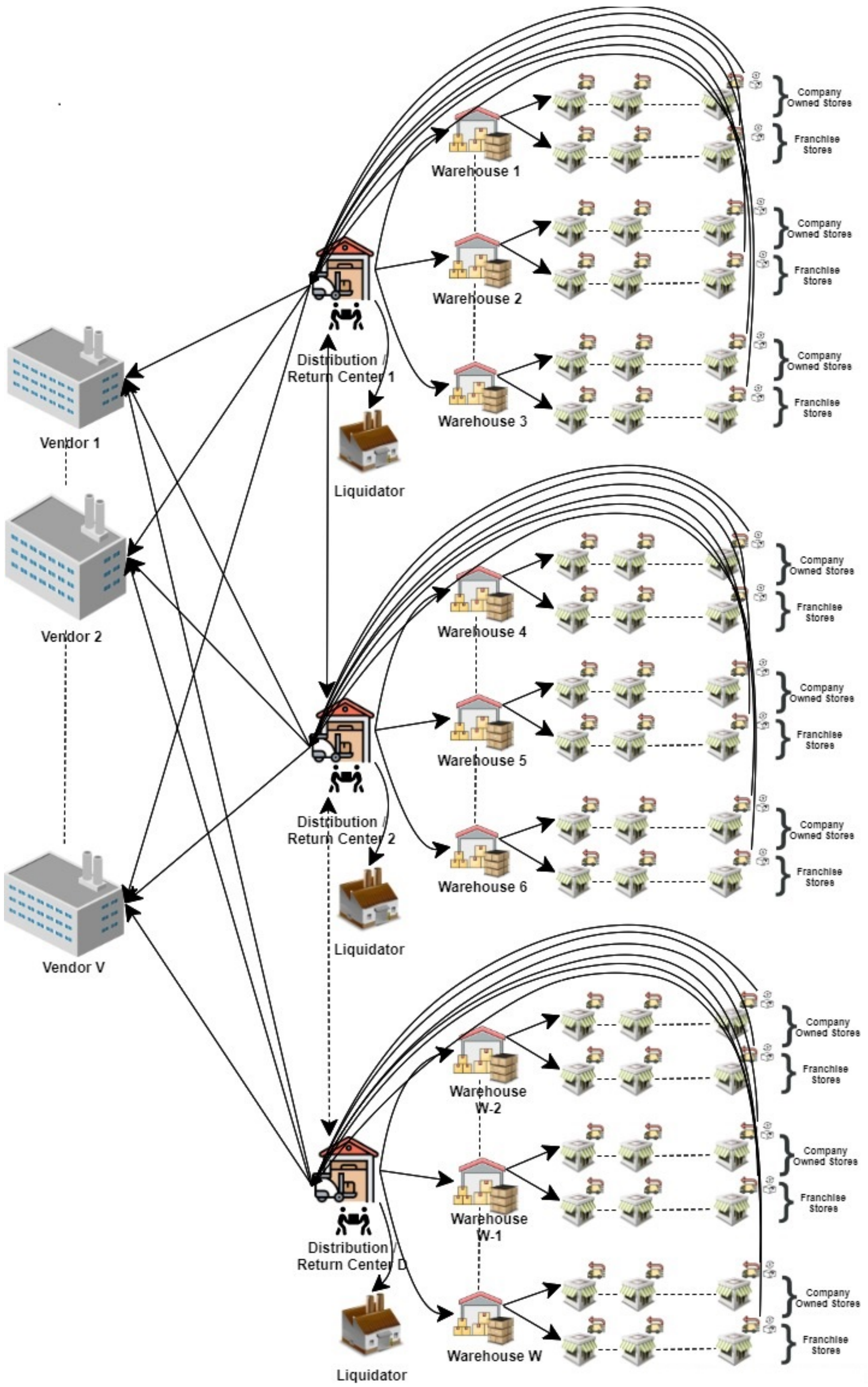

Figure 4.1: Comprehensive Retail Supply Chain Network and Product Movement

In order to understand and visualize the origin of the costs that we would like to minimize, the detailed cost structure of a product and the related cost derivatives are visualized and shown in *Figure 4.2*. The visual representation of a product's cost structure and the effects on relocating the product within the retail supply chain network helps us to develop our model. For any chosen product to be relocated within the retailer's RSC network, one or many of the costs in *Figure 4.2* will be incurred. As a result, the objective of our model is come up with a relocation schema of products within the network that would minimize overall costs while hitting the overall inventory goal of the retail company and also satisfying the and its individual franchisees' goals.

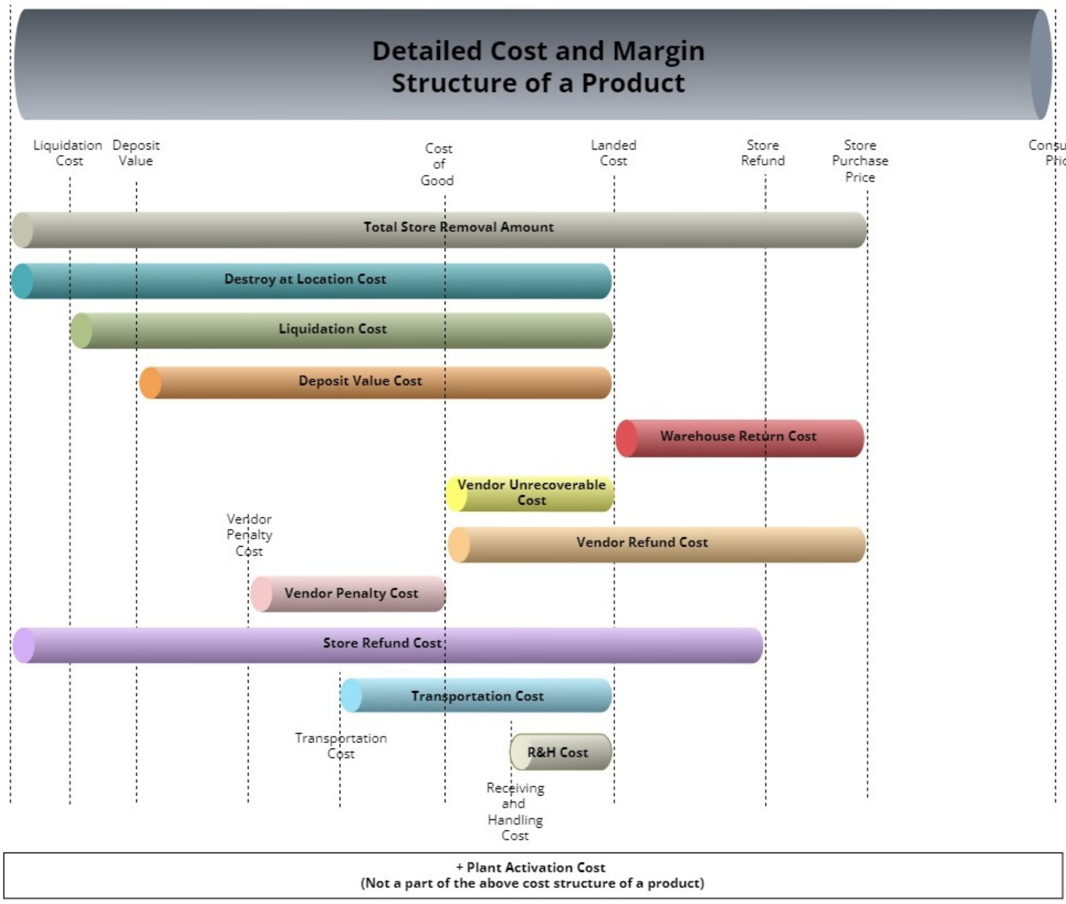

FIGURE 4.2: Visual Analysis of the Costs Incurred in a Retail Reverse Supply Chain

4.2 Optimization Model

We will first define the model then describe some of our assumptions.

4.2.1 Model

$$DestroyAtSiteCost + LiquidationCost + WarehouseReturnCost+$$
$$DepositValueCost + VendorRefundCost + VendorRefundUnrecoverableCost+$$
$$VendorPenaltyCost + TotalStoreRefund + TotalStoreRemoval+$$
$$TotalTransportationCost + Receving\&HandlingCost + PlantActivationCost$$

$$DestroyAtSiteCost = \sum_{s=1}^{S}\sum_{p=1}^{P} ds_{s,p}Q_{s,p}LC_p \tag{4.1}$$

$$LiquidationCost = \sum_{p=1}^{P}\sum_{d=1}^{D} lq_{p,d}\left(LC_p - COG_p LQR_p\right) \tag{4.2}$$

$$WarehouseReturn = \sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{w=1}^{W} wh_{p,d,w}\left(LC_p - SP_p\right) \tag{4.3}$$

$$DepositValueCost = \sum_{p=1}^{P}\sum_{d=1}^{D} dp_{p,d}(LC_p - DV_p) \tag{4.4}$$

$$VendorRefundCost = \sum_{p=1}^{P}\sum_{d=1}^{D} vr_{p,d}\left(SP_p - COG_p\right) \tag{4.5}$$

$$VendorRefundUnrecoverableCost = \sum_{p=1}^{P}\sum_{d=1}^{D} vr_{p,d}\left(LC_p - COG_p\right) \tag{4.6}$$

$$VendorPenaltyCost =$$

$$\sum_{v=1}^{V} \begin{cases} VPF1_v \, vf_v & \text{if } 0 \le vf_v \le RT1_v \\ VPF1_v \, RT1_v + VPF2_v \, (vf_v - RT1_v) & \text{if } RT1_v < vf_v \le RT2_v \\ VPF1_v \, RT1_v + VPF2_v \, (RT2_v - RT1_v) & \text{if } RT2_v < vf_v \le VF_v \\ +VPF3_v \, (vf_v - RT2_v) & \end{cases} \tag{4.7}$$

$$TotalStoreRefund =$$

$$TotalCompanyOwnedStoresStoreRefund +$$

$$TotalFranchiseStoresStoreRefund$$

$$TotalCompanyOwnedStoresStoreRefund =$$

$$\sum_{s^{(st=CS)}=1}^{S} \sum_{p=1}^{P} \Big(ds_{s,p} SP_p Q_{s,p} R_p + (1 - rt_{s,p}) SP_p Q_{s,p} R_p \Big)$$

$$TotalFranchiseStoresStoreRefund =$$

$$\sum_{s^{(st=FS)}=1}^{S} \sum_{p=1}^{P} \Big(ds_{s,p} SP_p Q_{s,p} R_p + (1 - rt_{s,p}) SP_p Q_{s,p} R_p \Big) \tag{4.8}$$

$$TotalStoreRemovalAmount =$$

$$TotalCompanyOwnedStoresStoreRemovalAmount +$$

$$TotalFranchiseStoresStoreRemovalAmount$$

$$TotalCompanyOwnedStoresStoreRemovalAmount =$$

$$\sum_{s^{(st=CS)}=1}^{S} \sum_{p=1}^{P} \Big(ds_{s,p} SP_p Q_{s,p} + (1 - rt_{s,p}) SP_p Q_{s,p} \Big)$$

$$TotalFranchiseStoresStoreRemovalAmount =$$

$$\sum_{s^{(st=FS)}=1}^{S} \sum_{p=1}^{P} \Big(ds_{s,p} SP_p Q_{s,p} + (1 - rt_{s,p}) SP_p Q_{s,p} \Big) \tag{4.9}$$

$$
\begin{aligned}
&TotalTransportationCost = \\
&\quad TransportationCostFromStoresToReturnCenters + \\
&\quad TransportationCostAmongReturnCenters + \\
&\quad TransportationCostFromReturnCentersToVendors + \\
&\quad TransportationCostFromReturnCentersToWarehouses + \\
&\quad TransportationCostFromReturnCentersToLiquidation \\
&\\
&\quad TransportationCostFromStoresToReturnCenters = \\
&\Big(\sum_{p=1}^{P-}\sum_{s=1}^{S}\sum_{d=1}^{D}(1-rt_{s,p,d})\; Q_{s,p}\; VOL_p\; TCD_{s,d}\Big) \\
&\quad TransportationCostAmongReturnCenters = \\
&\Big(\sum_{p=1}^{P-}\sum_{d=1}^{D}\sum_{d'=1}^{D} tr_{p,d,d'} VOL_p\; TCT_{d,d'}\Big) \\
&\quad TransportationCostFromReturnCentersToVendors = \\
&\Big(\sum_{p=1}^{P-}\sum_{d=1}^{D} vr_{p,d} VOL_p\; TCV_{d,v}\Big) + \\
&\Big(\sum_{p=1}^{P-}\sum_{d=1}^{D} dp_{p,d} VOL_p\; TCV_{d,v}\Big) \\
&\quad TransportationCostFromReturnCentersToWarehouses = \\
&\Big(\sum_{p=1}^{P-}\sum_{w=1}^{W}\sum_{d=1}^{D} wh_{p,d,w}\; VOL_p\; TCW_{d,w}\Big) \\
&\quad TransportationCostFromReturnCentersToLiquidation = \\
&\Big(\sum_{p=1}^{P-}\sum_{d=1}^{D} lq_{p,d}\; VOL_p\; TCL_d\Big)
\end{aligned} \tag{4.10}
$$

$$
\begin{aligned}
&Receving\&HandlingCost = \\
&\sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{d=1}^{D}(1-rt_{s,p,d})\; Q_{s,p}\; WE_p/VOL_p\; RHF_d + \\
&\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{d'\in D,\text{where } d'\neq d} tr_{p,d,d'}\; WE_p/VOL_p\; RHF_{d'}
\end{aligned} \tag{4.11}
$$

$$
\begin{aligned}
&PlantActivationCost = \\
&\sum_{d=1}^{D} ac_d\; DAC_d
\end{aligned} \tag{4.12}
$$

$$\text{s.t.}\quad ds_{s,p} \leq rt_{s,p} \qquad \forall\, s \in S, p \in P \tag{4.1}$$

$$1 - rt_{s,p} = \sum_{d=1}^{D} (1 - rt_{s,p,d}) \qquad \forall\, s \in S, p \in P \tag{4.2}$$

$$rt_{s,p} \leq rt_{s,p,d} \qquad \forall\, s \in S, p \in P, d \in D \tag{4.3}$$

$$\sum_{s=1}^{S} (1 - rt_{s,p}) Q_{s,p} = \sum_{d=1}^{D} vr_{p,d} + \sum_{d=1}^{D} dp_{p,d} + \sum_{d=1}^{D} lq_{p,d} + \sum_{d=1}^{D}\sum_{w=1}^{W} wh_{p,d,w} \qquad \forall\, p \in P \tag{4.4}$$

$$\sum_{s=1}^{S} (1 - rt_{s,p,d}) Q_{s,p} + \sum_{d \in D, \text{where } d \neq d'} tr_{p,d,d'} = vr_{p,d} + dp_{p,d} + lq_{p,d} + \sum_{w=1}^{W} wh_{p,d,w}\, E_{d,w} + \sum_{w=1}^{W} wh_{p,d,w}\, (1 - E_{d,w}) + \sum_{d' \in D, \text{where } d' \neq d} tr_{p,d,d'} \qquad \forall\, p \in P, d \in D \tag{4.5}$$

$$\sum_{p=1}^{P}\sum_{d=1}^{D} wh_{p,d,w} COG_p \leq WRA_w \qquad \forall\, w \in W \tag{4.6}$$

$$\sum_{d=1}^{D} wh_{p,d,w} \leq WDC_{p,w} \qquad \forall\, w \in W, p \in P \tag{4.7}$$

$$\sum_{d \in D, \text{where } d \neq d'} tr_{p,d,d'} \leq \sum_{w=1}^{W} wh_{p,d,w} + \sum_{d'=1}^{D} tr_{p,d,d'} \qquad \forall\, p \in P, d' \in D \tag{4.8}$$

$$tr_{p,d,d'} \leq btr_{p,d,d'}\;\; M \qquad \forall\, p \in P, d \in D, d' \in D \tag{4.9}$$

$$btr_{p,d,d'} + btr_{p,d',d} \leq 1 \qquad \forall\, p \in P, d \in D, d' \in D \tag{4.10}$$

$$\sum_{p=1}^{P}\sum_{d=1}^{D} vr_{p^{(v)},d}\, COG_p = vf_v \qquad \forall\, v \in V \quad (4.11)$$

$$vf_v \leq VF_v \qquad \forall\, v \in V \quad (4.12)$$

$$\Big[IEI_{st} - \sum_{s^{(st)}=1}^{S}\sum_{p=1}^{P}\Big(ds_{s^{(st)},p} SP_p Q_{s^{(st)},p} + \ (1-rt)_{s^{(st)},p} SP_p Q_{s^{(st)},p}\Big)\Big] \ \leq$$

$$NIER\,\Big[EI_{st} \ + IEI_{st} - \sum_{s^{(st)}=1}^{S}\sum_{p=1}^{P}\Big(ds_{s^{(st)},p} SP_p Q_{s^{(st)},p} + (1-rt_{s^{(st)},p}) SP_p Q_{s^{(st)},p}\Big)\Big] \qquad \forall\, st = \{CS\} \quad (4.13)$$

$$\sum_{s^{(st)}=1}^{S}\sum_{p=1}^{P}\Big(ds_{s^{(st)},p} SP_p Q_{s^{(st)},p} + (1-rt_{s^{(st)},p}) SP_p Q_{s^{(st)},p}\Big) \geq FRSA_{st} \quad \forall\, st = \{FS\} \quad (4.14)$$

$$\sum_{p=1}^{P}\Big(ds_{s^{(st)},p} SP_p Q_{s^{(st)},p} + (1-rt_{s^{(st)},p}) SP_p Q_{s^{(st)},p}\Big) \geq$$

$$\Bigg(\Big(EI_s \ + IEI_s\Big) \Big/ \sum_{s^{(st)}=1}^{S}\Big(EI_s \ + IEI_s\Big)\Bigg) FRSA_{st} \qquad \forall\, s \in FS \quad (4.15)$$

$$(1-rt_{s,p,d}) \leq E_{s,d} \qquad \forall\, s \in S, p \in P, d \in D \quad (4.16)$$

$$\sum_{d' \in D, \text{where } d'=d} tr_{p,d,d'} = 0 \qquad \forall\, p \in P, d \in D \quad (4.17)$$

$$tr_{p,d,d'} \leq E_{d,d'} \;\; M \qquad \forall\, p \in P, d \in D, d' \in D \tag{4.18}$$

$$wh_{p,d,w} \leq E_{d,w} \;\; M \qquad \forall\, p \in P, d \in D, w \in W \tag{4.19}$$

$$1 - rt_{s,p,d} \leq ac_d \qquad \forall\, s \in S, p \in P, d \in D \tag{4.20}$$

$$\sum_{d \in D, \text{where } d \neq d'} tr_{p,d,d'} \leq ac_{d'} \;\; M \qquad \forall\, p \in P, d' \in D \tag{4.21}$$

$$\sum_{d' \in D, \text{where } d' \neq d} tr_{p,d,d'} \leq ac_d \;\; M \qquad \forall\, p \in P, d \in D \tag{4.22}$$

$$wh_{p,d,w} \leq ac_d \;\; M \qquad \forall\, p \in P, d \in D, w \in W \tag{4.23}$$

$$vr_{p,d} \leq ac_d \;\; M \qquad \forall\, p \in P, d \in D \tag{4.24}$$

$$lq_{p,d} \leq ac_d \;\; M \qquad \forall\, p \in P, d \in D \tag{4.25}$$

$$\sum_{sc=1}^{SC} x_{sc} = 1 \tag{4.26}$$

$$ac_d = \sum_{sc=1}^{SC} E_{d,sc} \;\; x_{sc} \qquad \forall\, d \in D \tag{4.27}$$

$$(1 - rt_{s,p,d}) \leq \sum_{sc=1}^{SC} E_{s,d,sc} \;\; x_{sc} \qquad \forall\, s \in S, p \in P, d \in D \tag{4.28}$$

$$tr_{p,d,d'} \leq \sum_{sc=1}^{SC} E_{d,d',sc} \;\; x_{sc} \;\; M \qquad \forall\, p \in P, d \in D, d' \in D \tag{4.29}$$

$$wh_{p,d,w} \leq \sum_{sc=1}^{SC} E_{d,w,sc} \;\; x_{sc} \;\; M \qquad \forall\, p \in P, d \in D, w \in W \tag{4.30}$$

$$ds_{s,p},\ rt_{s,p} \in \{0,1\} \qquad \forall\, s \in S, p \in P \tag{4.31}$$

$$ac_d \in \{0,1\} \qquad \forall\, d \in D \tag{4.32}$$

$$btr_{p,d,d'} \in \{0,1\} \qquad \forall\, p \in P, d \in D \tag{4.33}$$

$$x_{sc} \in \{0,1\} \qquad \forall\, sc \in SC \tag{4.34}$$

$$lq_{p,d},\ tr_{p,d,d'} \in \mathbb{N} \qquad \forall\, p \in P, d \in D \tag{4.35}$$

$$dp_{p,d},\ vr_{p,d} \in \mathbb{N} \qquad \forall\, p \in P, d \in D \tag{4.36}$$

$$wh_{p,d,w} \in \mathbb{N} \qquad \forall\, p \in P, d \in D, w \in W \tag{4.37}$$

$$vf_v \in \mathbb{R} \qquad \forall\, v \in V \tag{4.38}$$

There are 12 main components of the objective function. The **Destroy at Site Cost** (4.1) is the cost of destroying products at the store without moving them to another location to be processed. **Liquidation Cost,** (4.2) is the cost of selling the products for their scrap price, e.g. parts, metal content, etc. **Warehouse Return Cost** (4.3) is the cost of gains (differences between product's Store Purchase Price and Landed Price) that are redeemed when a product is returned back to another store/warehouse. Since returning products to Warehouse for re-sale is the most suitable and, in general, least costly option among other choices, we want to maximize warehouse returns. We also do not only want to maximize warehouse returns but also want to return the most profitable products among other warehouse-returnable products since we will sell these products in their new location to be able to make the most profit as possible. **Deposit Value Cost** (4.4) is the cost of losses when we return a product back to its vendor for the recovery of the product's deposit value. **Vendor Refund Cost** (4.5) is the cost of losses because of the margin differences between a product's refund and its COG when we return it back to its vendor for the recovery of the product's whole COG. **Vendor Refund Unrecoverable Cost** (4.6) is the cost of losses that are unrecoverable between a product's the landed cost and its COG differences when we return it back to its vendor. **Vendor Penalty Cost** (4.7) is the cost of penalties that is charged to the retail company between various return thresholds. **Total Store Refund** (4.8) is the cost of losses incurred by the difference between the product's store price and its pre-determined refund. Each product's refund differs based on product's type, durability, reusability, and resell-ability. **Total Store Removal Amount,** (4.9) is the total store price amount that will be returned to stores. **Total Transportation Cost** (4.10) is the cost of transporting a product from one location to another location. Transportation costs are calculated per unit bases (in terms of cm^3) - averaging the total money spent of transportation from one location to another location in the past year / total volume carried. Therefore, transportation costs depend on the volume of the products and which locations they are carried among. **Receiving & Handling Cost** (4.11) is the cost of receiving and handling of a product at a RC. The cost depends on the weight and size/volume of the product (g/cm3) and the distribution center. Per unit cost is calculated based on previous years' average using number of products received, volume and weight processed, number of employees worked. Since different RCs have different supply and demand, per unit cost varies among different distribution centers. **Plant Activation Cost,** (4.12)**,** is the cost of activating a RC for the RSC activity to receive, clean, re-package, sort and send products.

Constraint Set (4.1) ensures that if the product is returned to the RC (if $rt_{s,p} = 0$), then the product will NOT be destroyed at site (then $ds_{s,p} = 0$) or if the product is destroyed at site (if $ds_{s,p} = 1$), then the product will NOT be returned to the RC (then $rt_{s,p} = 1$) or if the product stays at store, the product will NOT be destroyed at site ($ds_{s,p} = 0$) and the product will NOT be returned to the RC ($rt_{s,p} = 1$). **Constraint Set** (4.2) ensures that a store-product can only be returned to one RC. **Constraint Set** (4.3) ensures the logic between returning a product to a certain distribution center and returning a store-product in general, e.g., if a store-product is NOT returned to a distribution center, then a store-product CAN still be returned in general, if a store-product is returned to a distribution center, then the store-product has to be returned. **Constraint Set** (4.4) ensures that for each product, thst total number of items that are leaving the stores should be equal to the number of items that will be returned to the vendor, will be returned to the warehouses and liquidated. If there is only one distribution exists then products coming into the RC should be equal to the number of items that are going out from the RC. **Constraint Set** (4.5) ensures that the amount of items received by a RC, both from stores and transfers from other distribution centers, is equal to items that are going out, namely to vendors, warehouse replenishment, liquidation, transfers to related distribution center(s). **Constraint Set** (4.6) ensures that for each Warehouse, maximum return amount (in terms of $ value) to a Warehouse should be less than the capacity of that Warehouse. (Every Warehouse except products up to a certain amount, in terms of $ value.) **Constraint Set** (4.7) ensures that for each Warehouse and product, total number of items that are returned to each warehouse cannot exceed the Warehouse Capacity (in terms of quantity) for that product (There is a Warehouse demand for a specific product in every Warehouse and we cannot return products more than the certain limit to a Warehouse. This amount is actually the total demand of that product within the warehouse's distribution network, e.g. all of the stores' total demand.) **Constraint Set** (4.8) ensures that the received transfers cannot be sent to vendors or liquidation. They can only be sent to warehouses or further transferred to other related distribution center(s). **Constraint Set** (4.9) defines the logic whether a product is transferred or not via the existing route. If it the path is not used then the transferred amount is 0. **Constraint Set** (4.10) ensures that if a product is transferred from a distribution center to another related distribution center, then reverse path should NOT exist since it is not logical send products back and forth and the existence of such practice would increase the transportation costs. **Constraint Set** (4.11) ensures that for each Vendor, the value of the products that are going back to a Vendor is equal to total vendor funds that should be used. **Constraint Set** (4.12) ensures that for each Vendor, total vendor funds that should be used in order to return products can not exceed the available vendor dollars of that vendor. **Constraint Set** (4.13) ensures that after the removal of the ineffective inventory from company owned store, the remaining value of the ineffective store-products in overall inventory should be less than a certain / predetermined ratio, (say 10% based on our example in the model). **Constraint Set** (4.14) ensures that certain amount of ineffective inventory will be removed from all of the franchise stores' inventory. **Constraint Set** (4.15) ensures that the amount of ineffective inventory that will be removed from a franchise store's inventory, will be proportional to total inventory it carries compared to the other stores and/or total franchise stores' inventory (e.g., if Store A carries $100K of inventory and Store B carries $200K of inventory, then 2 times of unproductive inventory will be removed from Store B compared to Store A because Store B holds inventory twice as much and therefore should be able get rid of more ineffective inventory). This constraint is the assurance to recognize the transparency among franchise stores so that each franchise store can observe how much inventory is being held in each franchise store and therefore how much inventory are being removed from its store vs. other stores. **Constraint Set** (4.16) ensures that a store can only send items of a product to RC if a path from the store to the RC exits. **Constraint Set** (4.17) ensures that the a RC should not transfer items of a product to itself. **Constraint Set** (4.18) ensures that a distribution center can only transfer items of a product to another distribution center if a path from one RC to another one exits. **Constraint Set** (4.19) ensures that a distribution center can only send items of a product to warehouse if a path from the RC to the warehouse exits. **Constraint Set 4.20** ensures that if a distribution center is NOT activated, then the stores can NOT return products to this distribution center. **Constraint Set** (4.21) ensures that if a distribution center is activated then it can receive transfers from other related distribution centers, if not then there cannot be any transfer. **Constraint Set** (4.22) ensures that if a distribution center is activated then it can transfer products to other related distribution centers, if not then it cannot transfer any products. **Constraint Set** (4.23) ensures that a distribution center can only send items of a product to its warehouses if the RC is activated. **Constraint Set** (4.24) ensures that a distribution center can only send items of a product to vendors if the RC is activated. **Constraint Set** (4.25) ensures that a distribution center can only send items of a product for liquidation if the RC is activated. **Constraint Set** (4.26) ensures that only one scenario, out of all the possible scenarios, can be realized in the end. **Constraint Set** (4.27) ensures that when one scenario is being realized, related RCs are either activated or deactivated based on the rules of that scenario. **Constraint Set** (4.28) ensures that when one scenario is being realized and if a RC is not activated within that scenario, then returning store-products to that RC was not allowed. This constraint set is almost the same constraint as Constraint Set(4.21). **Constraint Set** (4.29) ensures that when one scenario is being realized and if a RC is not activated within that scenario, then transferring products from that RC to another RC was not allowed. **Constraint Set** (4.30) ensures that when one scenario is being realized and if a RC is not activated within that scenario, then sending products for warehouse return from that RC was not allowed. **Constraint Set** (4.31)–(4.34) declares the set of binary variables of the model. **Constraint Set** (4.35)–(4.37) declares the set of integer variables of the model. **Constraint Set** (4.38) declares the set of continuous variables of the model.

The model makes the decision of which ineffective store-products, from both company owned and franchise stores, to be chosen to participate in the RSC activity, where those chosen ineffective Store-Products will be allocated within the RSC network, and activating which RCs to be selected for the RSC activity considering important costs such as Plant Activation Costs, Transportation and Receiving and Handling Costs. The model considers the detailed profit and cost structure of a product. Namely; COG, deposit value of the product, liquidation price, landed cost of the product to the retail company, store purchase price and refund price (see *Figure 4.2*). In addition, the model considers important logistics and distribution costs such as transportation costs among plants/stores/vendors, and receiving and handling costs related to movement of the product within the RSC. Consideration of physical structure of the products such as weight - volume and their effects on the related costs such as transportation costs and receiving and handling costs are also given importance since it affects the distribution and allocation of the products, and therefore the costs, within the network significantly. The model also considers a multi-vendor, multi-distribution center, multi-warehouse, multi-store-type, and multi-store network structure. This realistic network structure lead to a RL process that very costly and therefore consideration of activation/de-activation of distribution centers (costs) and its effect on the overall RSC costs are also implemented in order to make not only operational selection-allocation decisions, but also tactical/strategic decisions. When certain returns centers are active, the network structure and hence network routing of the products changes drastically.

4.2.2 Assumptions

We assume transparency among store types and transparency among franchise stores where there is absolute transparency among all network members. All stores, including the retail company, know all other franchise stores' inventory levels compared to overall inventory level. Therefore, the reverse pull amount will be proportional to a certain ratio which would provide sufficient explanation to franchise stores by the retail company about their expected return potential. We assume that the physical (count) and value ($) based capacity issues at warehouses are known. Physical (volume) structure of products can also be considered in our model as a realistic constraint for the warehouses as an extension in order to deal with another aspect of physical capacity issues at the warehouses. Just to bring a little simplicity and develop our heuristic model accordingly without adding another layer of complexity, such consideration is eliminated in our model design. We assume we have a limited amount of vendor funds with a multi-layer penalty structure for each vendor when returning products back to their vendor. Such consideration affects the return costs significantly and identify clever ways to save money on returns.

4.3 Heuristic Algorithm

In *Figure 4.3* we summarize the major steps of the suggested heuristic algorithm.

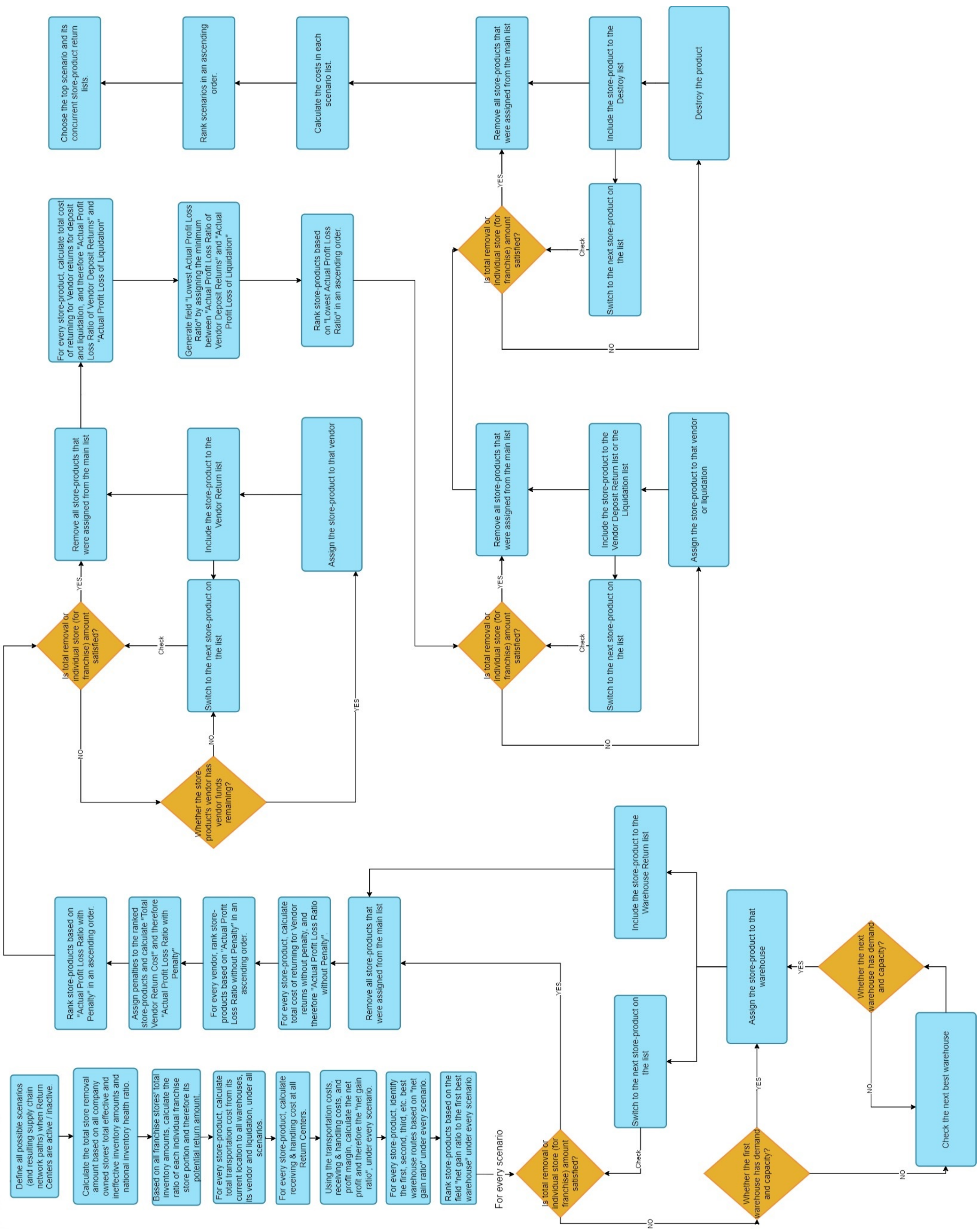

Figure 4.3: Multi-Stage Heuristic Algorithm for the State-of-the-Art Retail Reverse Supply Chain Model

The detailed steps of the heuristic are described in Appendix A2. We will provide more details about the heuristic logic later in this chapter.

4.4 Test Problems, Computations and Numerical Analysis

In this section, we present our findings about the test problems we have solved using CPLEX and compare the results and performance of our heuristic to the results of the exact solver.

4.4.1 Test Cases and Data Generation

For our test problems, we have generated a RRSC network of 500 Stores under 2 store-types, 250 franchise stores and 250 company owned stores, 4 distribution centers, 8 warehouses (2 warehouses under each RC), and 20 vendors. The test cases' problem sizes ranged from 10,000 store-products up to 100,000 store-product combinations. Each parameter of the product is generated with random data that fits a uniform distribution between certain bounds (lower and upper limit) that are consistent with values of the related parameters; COG, landed cost, store purchase-price cost, store refund rebate rate, deposit value of the product, liquidation rebate rate, weight and volume of the products. Also, other problem parameters such as unit costs of transportation, receiving & handling fees at distribution centers, warehouse capacities, product demand at warehouses, facility activation costs, available vendor funds, return thresholds within available vendor funds, penalty rate between these breaks, effective inventory amount at each store, franchise stores' total RSC amount and company owned stores national ineffective inventory ratio goal are generated in way that there is a wide range of variability for capacity, demand, and cost effects on profitability.

4.4.2 Test Case Problem RRSC Network Generation

For the test problems, a complex RRSC network layout is designed with a consistent cost structure (transportation and receiving&handling) that is traceable with the physical layout of the actual supply chain network. All store-product combinations are randomly generated where each product is assigned to a random vendor and since all of the above mentioned parameters of the products are randomly generated, each product's profit, cost, physical structure and other related parameters differ from each other. Therefore, we have a wide range of products in terms of profitability, costs, product dimensions. Since other problem parameters related to stores (effective inventory amounts, total RSC amount, goal ratio), warehouses (capacities and product demands), distribution centers (activation costs), vendors (funds, breaks, penalties) are generated that is consistent with physical and financial structure of the entities, we designed and created a realistic RRSC network for testing the performance of our test cases via CPLEX and our heuristic.

4.4.3 Test Case Problem Sizes

In total, we generated and solved 20 test problems. Small to large size problems are generated and solved both using CPLEX and our heuristic algorithm. The first 5 test problems consisted of 10,000 store-product combinations, the second 5 test problems consisted of 25,000 store-product combinations, the third 5 test problems consisted of 50,000 store-product combinations and fourth 5 test problems consisted of 100,000 store-product combinations. As can be observed from *Table 4.1*, smaller size problems, 10,000 store-product combination problems, were solved within 20 minutes to 1 hour and larger size problems, 100,000 store-products combination problems, were solved within 6-8 hours while asking for solutions within 1% optimality range. Solutions found within the given time and/or optimality range can be observed on *Table 4.1*.

The above RRSC inventory optimization and network planning model is a mixed integer linear program (MILP) and is designed and coded with an exact solver software, IBM ILOG CPLEX Optimization Studio 12.8, on a PC with an Intel Core i7-8550U CPU @ 1.8 GHz, 4 Cores, 8 Logical Processors and 20 GB of RAM.

4.4.4 Heuristic Data Processing and Deployment

Once we have generated our data as described in above section, the data is stored into database tables for the heuristic to run against. The heuristic is then coded in SQL Server as Stored Procedures for inventory relocation and each heuristic stage in these stored-procedures run sequentially for each scenario until desired amount of the store-products are removed from stores. Once the heuristic is run for a given problem, the resulting allocation is collected from resulting database tables and total solution time and model results are shared in *Table 4.1*.

4.4.5 Solver Solution Times

We can compare the results and solution times found by CPLEX and our heuristic algorithm using *Table 4.1*. In order to compare the solution times between CPLEX and our heuristic, we first solved the test problems via CPLEX by asking 1% optimality range and waited until a solution was found that is very close this optimality range. We then solved the test problems via our heuristic and calculated to the optimality range for the heuristic. We then solve the test problems via CPLEX again by asking the optimality range that is found by our heuristic. This way we can compare the results founds and, more importantly, solution times between CPLEX and our heuristic under the same given optimality range.

4.4.6 Comparison of Solver Solution Times and Heuristic

For smaller size problems, 10,000 store-product combination problems, the solutions found by our heuristic were calculated to be within the 1-3% of the optimality range and solution times range between 18-21 minutes. We then solved these test cases with CPLEX for the same optimality ranges and CPLEX was able to find a solution that is very close to this optimality range within 30-40 minutes for the test problems. Our heuristic solution algorithm finds 'good' solutions, 1-3% of the optimality range, within reasonable solution times. It also outperformed CPLEX in terms of solution times for about 10-20 minutes depending on the test case.

When problem size got larger and larger, for $25,000$, $50,000$ and $100,000$ store-product combinations, respectively, the heuristic was solid and was able to find solutions within reasonable solution times that are 1-5% of the optimality range. For $25,000$ store-product combination test problems, the solution times were about an hour, for $50,000$ the solution times were about 2 hours and for $100,000$ store-product combination problems the solution times were between 4-5 hours. Since our heuristic solution algorithm solves every potential scenario within a given test case and then chooses the best solution, the solution times increased with respective to this algorithm structure and its effect over the problem size.

If we take a deeper look into the computation performance of our heuristic, we can easily observe that our heuristic outperformed CPLEX in terms of solution times for large size problems. In some test cases, CPLEX was stuck on the given optimality range found by our heuristic with a sub-optimal solution and was not able to improve the result within the maximum given computation time. These test problems show that our heuristic solution algorithm is a stable solution algorithm and can solve very large size problem with reasonable solution times that are very close to optimality and with less computational resources even we solve all the potential scenarios for a problem.

4.4.7 Comparison when Return Centers are Stationary

We have also run test cases where we set the RRSC network to a stationary (where we know which RCs are activated) and the solution times of our heuristic dropped drastically (in these test cases, almost $(1/16)^{th}$ on the original computation time) because we were only solving one scenario. However, when the same scenario was solved with CPLEX, the computation time did not change significantly since CPLEX's internal solution algorithm uses a different methodology, branch-and-cut, to find the near optimal solutions that might not be consistent with the complexity of the solution space. Having a much smaller solution space without dealing with the complexity of selection of RCs did not affect the solution time for the CPLEX. However, since our heuristic solves each scenario, ranks them in terms of cost, and then chooses the scenario where the costs are minimized, our heuristic solution times grow linearly with respect to the scenarios in the problem it has to calculate. Nevertheless, our heuristic solution algorithm still solves large size problems within reasonable solution times and outperforms CPLEX solution times.

Test Case	Problem Size (# of Store - Products)	Solution Methodology Used	Optimality Range Found within the Given Time	Solution Time Given or Found (seconds)	Solution Time Given or Found (minutes)	Solution Time Given or Found (hours)	TOTAL COST of REVERSE SUPPLY CHAIN ACTIVITY
Test 01	10,000	CPLEX	2.31%	1,201	20	0.33	$ 779,029
		Heuristic	2.50%	1,163	19.23	0.32	$ 780,587
Test 02	10,000	CPLEX	2.76%	1,200	20	0.33	$ 706,012
		Heuristic	3.24%	1,106	18.26	0.31	$ 709,513
Test 03	10,000	CPLEX	3.84%	3,603	60	1.00	$ 583,865
		Heuristic	3.73%	1,084	18.04	0.31	$ 583,168
Test 04	10,000	CPLEX	1.26%	3,603	60	1.00	$ 870,638
		Heuristic	3.71%	1,237	20.37	0.34	$ 892,832
Test 05	10,000	CPLEX	1.12%	3,603	60	1.00	$ 1,141,319
		Heuristic	2.42%	1,313	21.53	0.36	$ 1,156,477
Test 06	25,000	CPLEX	3.78%	7,202	120	2.00	$ 853,607
		Heuristic	4.58%	3,375	56.15	0.94	$ 860,758
Test 07	25,000	CPLEX	1.84%	10,804	180	3.00	$ 917,947
		Heuristic	4.68%	3,429	57.09	0.95	$ 945,268
Test 08	25,000	CPLEX	2.16%	7,208	120	2.00	$ 1,042,709
		Heuristic	4.42%	3,485	58.05	0.96	$ 1,067,373
Test 09	25,000	CPLEX	4.05%	7,216	120	2.00	$ 1,029,828
		Heuristic	5.53%	3,481	58.01	0.96	$ 1,045,929
Test 10	25,000	CPLEX	5.98%	7,201	120	2.00	$ 1,293,328
		Heuristic	5.65%	3,626	60.26	1.00	$ 1,288,861
Test 11	50,000	CPLEX	1.19%	28,817	480.3	8.00	$ 2,617,198
		Heuristic	6.02%	8,280	138	2.18	$ 2,751,560
Test 12	50,000	CPLEX	0.99%	24,940	415.6	6.93	$ 2,324,932
		Heuristic	4.62%	9,000	150	2.30	$ 2,413,525
Test 13	50,000	CPLEX	2.02%	14,412	240.2	4.00	$ 2,337,156
		Heuristic	3.74%	8,100	135	2.15	$ 2,378,925
Test 14	50,000	CPLEX	1.80%	14,239	237	3.95	$ 2,109,302
		Heuristic	6.25%	7,740	129	2.09	$ 2,201,606
Test 15	50,000	CPLEX	2.86%	11,244	187.4	3.12	$ 1,906,949
		Heuristic	3.18%	7,560	126	2.06	$ 1,913,348
Test 16	100,000	CPLEX	1.13%	28,806	480	8.00	$ 4,149,720
		Heuristic	0.88%	16,260	271	4.31	$ 4,139,719
Test 17	100,000	CPLEX	1.09%	28,802	480	8.00	$ 5,745,484
		Heuristic	0.38%	18,900	315	5.15	$ 5,704,475
Test 18	100,000	CPLEX	0.88%	24,059	400.1	6.68	$ 6,108,926
		Heuristic	1.02%	19,920	332	5.32	$ 6,117,818
Test 19	100,000	CPLEX	1.41%	25,206	420	7.00	$ 4,502,148
		Heuristic	0.26%	16,620	277	4.37	$ 4,450,337
Test 20	100,000	CPLEX	1.62%	6,222	103	1.72	$ 6,587,367
		Heuristic	0.21%	20,820	347	5.47	$ 6,468,131

TABLE 4.1: Test Problem Results for State-of-the-Art Retail RSC Model, Solver vs Heuristic Results Performance Comparison

4.5 Heuristic Design, Insights and Results

In our RRSC inventory optimization and network planning model, we were dealing with a complex retail supply chain network where the network consisted of multi-RCs, -warehouses, -stores, -store-types, -vendors and -products. The optimization problem was concerned with minimizing all inventory costs while considering which RCs to be activated for the RSC activity. The network members consisted of two different entities, company owned stores and franchise stores. For company owned stores, the stores acted as one, a.k.a the retail company, and the decisions made by them included all the stores together where the overall inventory health status of the retail company was concerned. However, for the franchise stores, they had individual objectives in terms how much inventory they wanted to return and they wanted to optimize their inventory individually without the status of other stores. Therefore, there was conflict between retail company and individual franchise stores. The model has addressed this conflict by inventory transparency between these franchise stores. The transparency worked in a way where all franchise stores were able to share their inventory information and observe how much effective and ineffective inventory were held by each franchise store. This way franchise stores knew how much was going to be pulled from each store's inventory before the RSC activity since the total RSC pull amount was known by all the franchise stores and each franchise store knew how much inventory others franchise stores were carrying. The inventory removal amount from each store were directly proportional with their total amount of inventory they were carrying. If a franchise store held more inventory, that meant that more store-products (in terms of $ value) were going to be pulled from those stores, no matter how much ineffective inventory they were carrying. This idea of transparency and information sharing among franchise stores resolved the conflict between the retail company and its franchise stores, as well as resolving the conflict among franchise stores.

The optimization model's consideration of multi-RC and -warehouse structure resulted in a complex web of product RL paths within the huge number of network combinations 'potentially-generated' RSCs. When some of the RCs were activated / de-activated, the model considered all the new potential paths and therefore related costs such as transportation, receiving&handling and plant activation costs that is tied to the 'potential' new network plan.

The optimization model also optimized inventory based on the complex penalty structure of vendors where the penalty of returning products has several layers and different penalty percentages between different pre-determined thresholds. The more you returned, the more penalty percentage you paid to the vendor after some pre-determined value. This made the objective function of the optimization model much more complex than it already is and we had to develop a complex heuristic (a sub-routine) within our overall heuristic algorithm just to be able to address this issue.

The resulting model was a comprehensive optimization model that would minimize all the related inventory costs while considering physical structure of the supply chain network, and all the product, warehouse and vendor related parameters. Complexities of dealing with conflicting objectives among different store types (and also among franchise stores at the same time) was also a challenge from a modeling perspective.

We have then developed a very specific heuristic to solve this problem while considering all the interrelated parameters. The overall heuristic consisted of 5 main components with sub-routines in order to achieve the main goal of optimizing inventory: Warehouse Return Heuristic, Vendor Penalty Assignment Heuristic, Vendor Return Heuristic, Deposit&Liquidation Heuristic and Destroy Heuristic. These smaller but interrelated parts of the heuristic worked sequentially until all stores' inventories is optimized against national inventory goals are reached for company owned stores and individual inventory goals are reached for franchise stores.

As discussed in *Chapter 3*, the cost of using state-of-the art solvers to solve this problem is high, especially for smaller companies. This sheer cost might be one of the main reasons why retailers need to develop similar heuristics to solve such problems. When these solvers run realistic large size problems (most probably much larger than we have shown in *Table 4.1*), they will easily realize that there will be lots of CPU, memory and even hard-drive issues on the computers that will run such problems. For a relatively large size problem, we have run out of memory (20 GB) within the first hour of the solution time and used CPLEX options to write the majority of branches of the solution to local hard-drives. Therefore, developing a heuristic to solve this problem is much more of a necessity to solve realistic size RRSC inventory optimization problems.

Chapter 5

Contract Terms for Contract Re-negotiation in Retail Reverse Supply Chain Management

5.1 Problem Definition

In this chapter, we consider product-returns contracts between an independent retailer and all of its vendors (or suppliers/manufacturers/producers/distributors) in RRSC management. During contract renewal time for its product-returns, the retailer needs to re-negotiate existing product-returns contracts with all of its vendors and needs to come up with the best possible parameters-offer to convince its vendors to accept its new contract terms. The retailer is aware of the tough situation where the new offers that will be given to vendors should satisfy them in terms of the value it generates for them and hence, combined effect of the new contract parameters offered to vendors can not be worse than what current parameters generated in the existing contracts. The goal of the retailer is to identify optimal contract parameters, namely penalties (penalty percentages) and return thresholds, between the retailer and all of its vendors for contract re-negotiation for its RSC activities.

The retailer-vendor product-returns contracts are created for the retailer to return unwanted products back to their vendor via a multi-layered penalty structure deal. These kind of contracts are categorized as 'Buy-Back' contracts in the literature when there is no (reverse and/or forward) supply chain coordination between the retailer and its suppliers, such as profit or cost sharing, to share the burden of unsold/returned/damaged/faulty inventory. In these kinds of return contracts, the retailer purchases are subject to a buy-back by their suppliers with increasing levels of penalties. This buy-back contract between the retailer and vendor can be referred to as a multi-layered penalty structure deal. This means that the retailer does not pay the same amount of penalty ratio all the time, there are several and increasing layers of penalties. As the retailer returns more products back to its vendor, after certain thresholds, the percentage of the penalty the company pays increases. The more the retailer returns products back to its vendors, the more 'penalty percentage' they pay, from the original purchase price, over certain thresholds. This multi-layered penalty structure discourages the retailer to return more products back to its vendors since the penalty percentage the retailer pays increases or return-value percentage they get from every return decreases as the retailer returns more and more products back to its vendors.

There are several important terms in the contracts, namely; total available vendor funds, return thresholds and penalty fees/percentages. The total potential amount of returns to a vendor, a.k.a. 'Total Available Vendor Funds', is the maximum amount that a vendor can accept as returns from the retailer and depends on the purchases from a vendor that is done in the previous year/term and is generally calculated through a portion of the total purchase amount from a vendor in a certain time period. Total available vendor funds is calculated at the beginning of every RSC term via a portion of the potential purchases from the vendor at the beginning of the term and is embedded to the problem as a fixed parameter. Since this fixed amount is the maximum amount the vendor can accept as returns, it is the highest threshold and therefore a maximum limit to the vendor's acceptance level for product returns.

Since the retailer can return products back to its vendors up to a certain amount, 'Total Available Vendor Funds', there are also multiple thresholds up to this limit, which are called 'Return Thresholds', for each vendor. These thresholds are intermediary thresholds that warns retailer to expect higher penalty charges to be enforced in case retailer decides to return over these threshold. If a threshold is passed when returning products back to the vendor, the retailer starts paying a higher percentage of the product's purchase price back to the vendor, hence discouraging the retailer to return more products as a result. If all thresholds are passed while returning products to a specific vendor, the retailer can return up to the total available vendor funds to that vendor. For every vendor, its total available vendor funds and their intermediary thresholds differ significantly from one vendor to another and the retailer wants to negotiate some of these parameters with all of their vendors in order to pay lesser penalties for future returns.

Between each of the thresholds, a vendor charges certain penalty fees/percentages when receiving products from the retailer and these penalties increase according to increasing thresholds to discourage the retailer to return less products. This means that the percentage penalty fee increasingly changes up to the next return threshold. This phenomenon of increasing penalty structure can be referred as a multi-layer penalty structure. The increasing penalty structure stops when ultimate threshold, total available vendor funds, is reached.

An example of this multi-layer penalty structure illustration of a vendor is shown on the left hand side in *Figure 5.1*. This figure illustrates the relationship between a given product return amount and penalty percentages that would be charged between thresholds for a sample vendor. For any given return amount, the area under the function also correspond to the total penalty that would be charged to the retailer. If we assume that the retailer expects to make returns worth $500,000 to a specific vendor, and as a result uses all of the available vendor funds for that vendor, then the retailer would be charged equivalent to the area under the function, which can be calculated as $100,000 * 10% + ($250,000 - $100,000) * 25% + ($500,000 - $250,000) * 50% = $180,000. As a easy-to-observe summary, the relationship between a given product return amount to a specific vendor and the actual penalty that would be paid to that vendor can be illustrated as on the right hand side in *Figure 5.1* This figure summarizes the effects of the penalty structure of a vendor and visualize the penalty that would be charged in a concise and effective way.

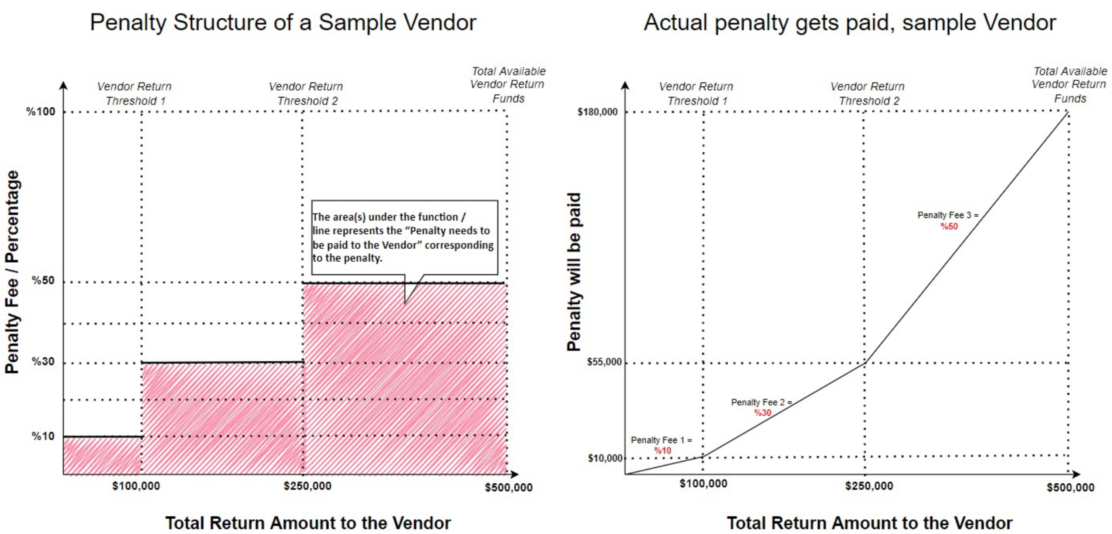

FIGURE 5.1: Penalty Structure of a Sample Vendor

Based on all of the above information, there are 2 important terms in the contracts that can be negotiated with the vendor, penalty fee/percentage and return thresholds. The objective of the retailer is to identify the best penalty fee structure and/or return thresholds that will minimize the penalty that would be incurred for a calculated future return compared to what would have been incurred if those same parameters were kept the same. The retailer and the vendors know the historical return amounts for every vendor, however, only the retailer knows, the future (expected) return amounts to every vendor. This phenomenon is resulting from the fact that the retailer have internal sales and inventory data (accompanied with other promotions, deals, clearance, etc. internal data) with the capability of forecasting future sales of its products, hence can calculate the future state of its inventory, which in return, the potential returnable inventory amount to its every vendor. Since vendors can not have this internal data of the retailer, they can only work with the historical data which both sides already have access to.

As a result, the retailer's objective is to find the optimal penalty fee structure and/or return thresholds that should be in place for each vendor, and therefore should be negotiated towards in contract re-negotiations. Since the retailer has leverage against its vendors via its internal data and information, this can benefit the retailer to better negotiate its future contracts for its RSC activities.

Since we have two distinct set of variables that will be negotiated with each vendor, we first design our model in order to identify best possible parameters, both vendor penalty fees and return thresholds, simultaneously in every vendor contract. The objective is to find the optimal vendor penalty fees and return thresholds which will minimize the penalty would be charged to retailer, by its vendors, for its future-expected returns. A better explanation of the retailer's goal would be to pay a lesser (or minimum) penalty as much as possible, based on the pre-existing conditions and the rules of the existing agreement, compared to what would have been paid if the old contract parameters were kept as is. We model the problem as a Mixed Integer Non-Linear Program (MINLP) and solve test case problems with the state-of-the-art MINLP solvers. We solve small size to large size problems and try many different MINLP solvers, such as AlphaECP, ANTIGONE, BARON, Bonmin, Knitro, DICOPT, LINDOGlobal and SBB using GAMS, to compare the performance of these solvers on the given problem.

In order to gain more insight into the penalty fee structure and return threshold behaviour of the problem, we model them separately as two distinct problems where we keep the return thresholds (for vendor penalty fee problem) and vendor penalty fees (for return thresholds problem) as known parameters. The goal for modelling them separately is to gain insight into the intertwined mechanism of the vendor penalty fees and return thresholds. Since considering one of them removes the non-linearity and the complexity of the problem, we hope to gain insights and extract a more generalized approach from the observed results. This methodology of our approach can be observed and tracked from *Figure 5.2*.

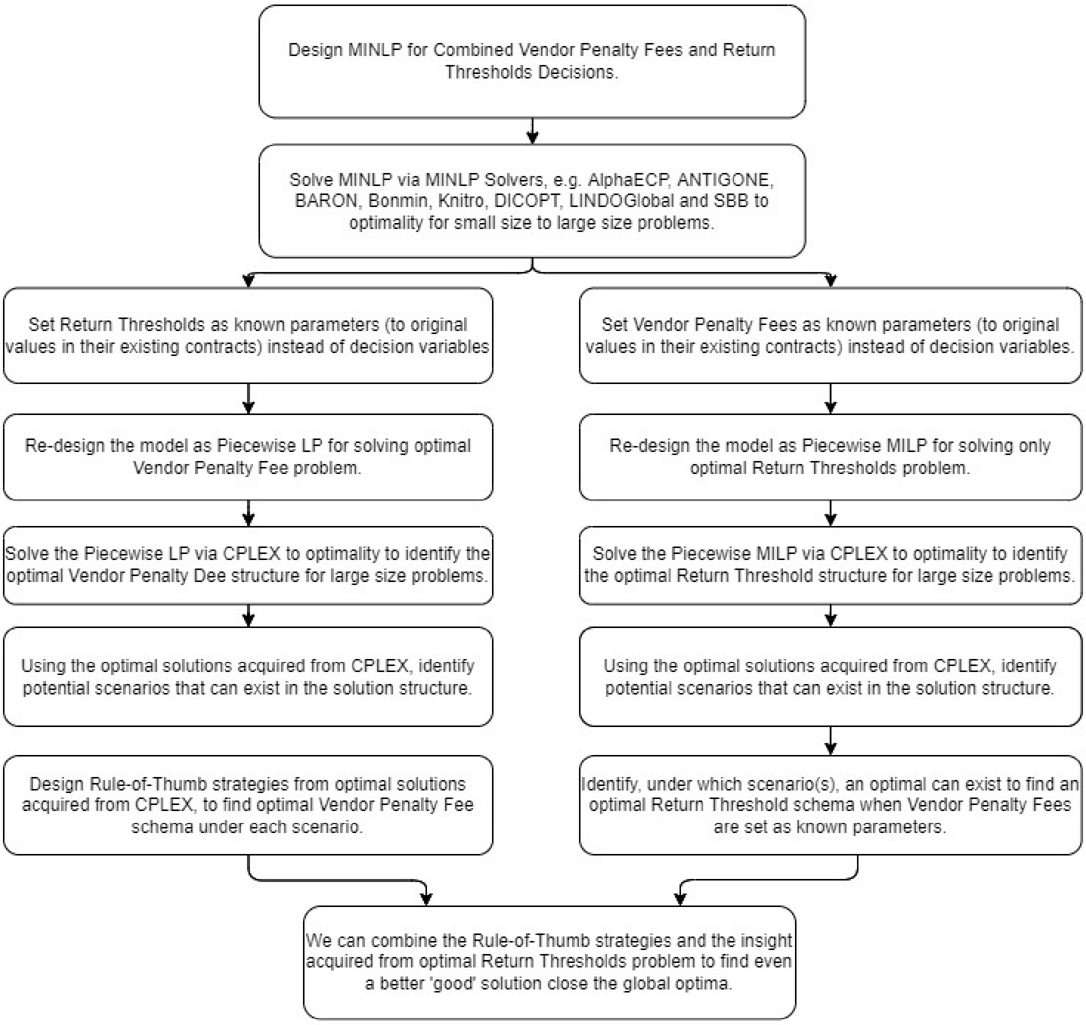

FIGURE 5.2: Methodology for Identifying Optimal Vendor Penalty Fees and Return Thresholds

When return thresholds are kept as existing parameters in the contract, the new problem can be modeled as a Linear Program (LP) and therefore response to the changes for given parameters, such as historical and expected returns, can be extracted. The objective of the second is to find the optimal vendor return penalty fees/percentages between each of return thresholds that are fixed. When the vendor penalty fees are kept as existing parameters in the contract, the new problem can also be modeled as a Mixed Integer Linear Program (MILP). The objective of the third problem is to find the optimal vendor return thresholds when penalty fees/percentages are kept the same as in the existing contract. We generate and solve test cases for the second and the third problem using a state-of-the-art solver, CPLEX. We then investigate the results for both of the test problems, generate rule-of-thumb strategies to solve the problems with structured mechanisms, extract important insights, and summarize our findings. Future research direction in this area are then provided about potential topics that might be of interest.

We illustrate the optimal vendor penalty fees / percentages in *Figure 5.3* and optimal vendor return thresholds in *Figure 5.4* for a sample vendor and show how the objective, historical and expected returns, and boundary parameters affect the problem structure by visually analyzing these graphics.

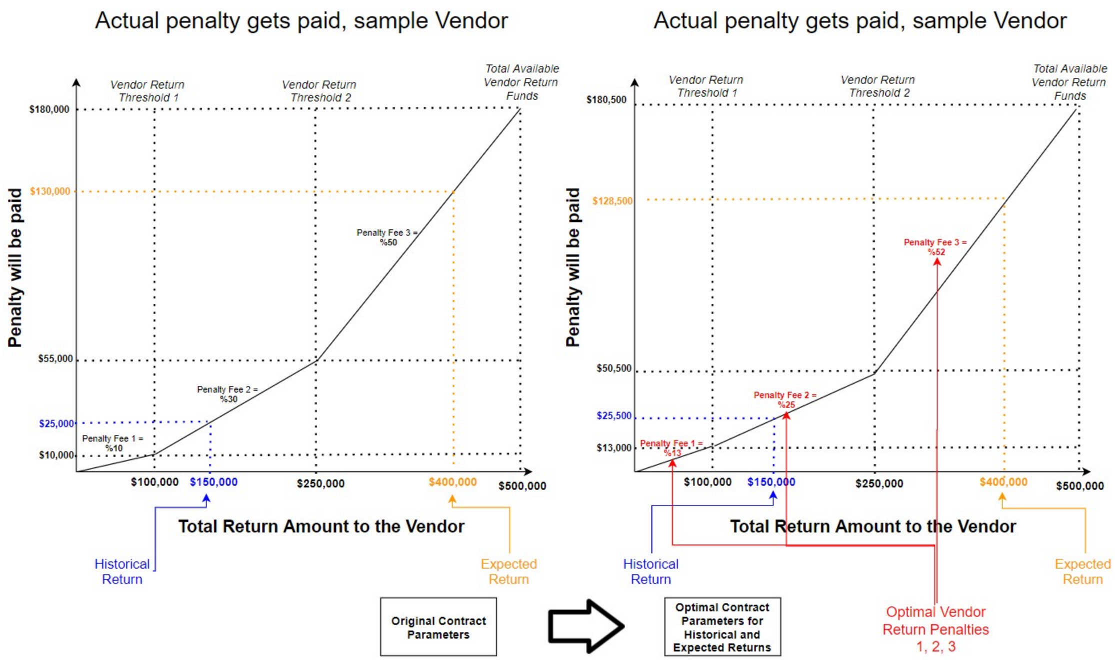

Figure 5.3: Identifying Optimal Vendor Penalties for a Sample Vendor, Before & After

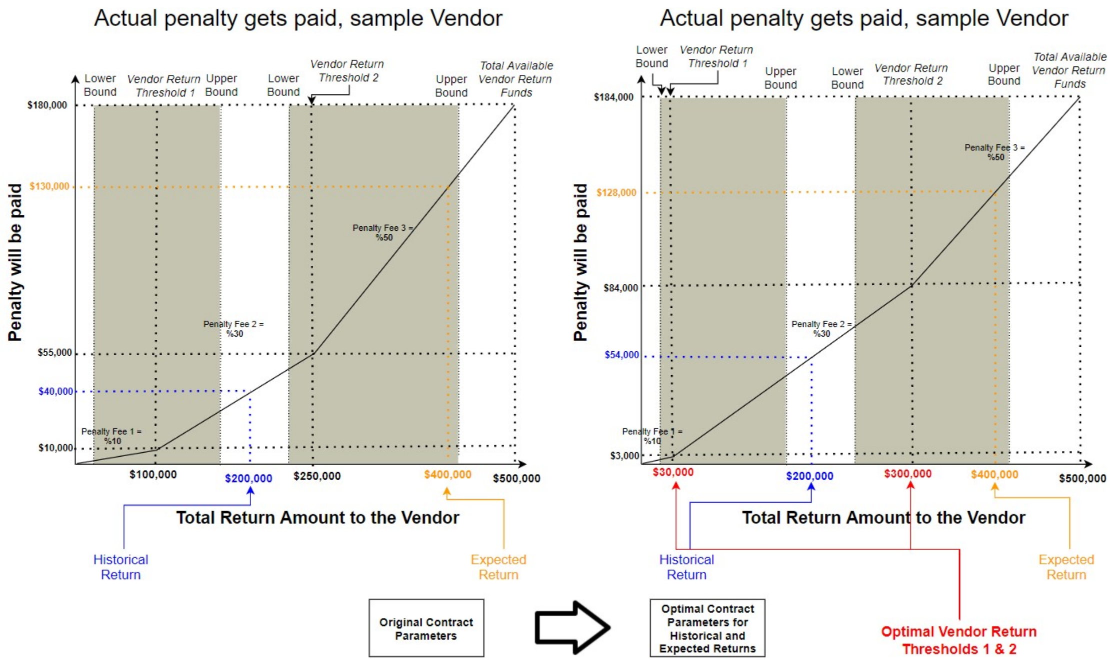

FIGURE 5.4: Identifying Optimal Vendor Return Thresholds for a Sample Vendor, Before & After

5.2 Model for Vendor Penalty Fee & Return Thresholds Decisions

Below we show the problem of jointly optimizing for the penalty and threshold. It results in a mixed integer nonlinear problem (MINLP).

$$\sum_{v=1}^{V} \begin{cases} vpf1_v \; EVR_v & \text{if } 0 < EVR_v \leq RT1Min_v \\ vpf1_v \; rt1_v \; bve1_v \\ + \, vpf2_v \; (EVR_v - rt1_v) \; bve1_v \\ + \, vpf1_v \; EVR_v \; (1 - bve1_v) & \text{if } RT1Min_v < EVR_v \leq RT1Max_v \\ vpf1_v \; rt1_v \\ + \, vpf2_v \; (EVR_v - rt1_v) & \text{if } RT1Max_v < EVR_v \leq RT2Min_v \\ vpf1_v \; rt1_v \\ + \, vpf2_v \; (EVR_v - rt1_v) \; (1 - bve2_v) \\ + \, vpf2_v \; (rt2_v - rt1_v) \; bve2_v \\ + \, vpf3_v \; (EVR_v - rt2_v) \; bve2_v & \text{if } RT2Min_v < EVR_v \leq RT2Max_v \\ vpf1_v \; rt1_v \\ + \, vpf2_v \; (rt2_v - rt1_v) \\ + \, vpf3_v \; (EVR_v - rt2_v) & \text{if } RT2Max_v < EVR_v \leq TVR_v \end{cases}$$

$$\text{s.t.} \quad RT1Min_v \leq rt1_v \qquad \forall\, v \in V \tag{5.1}$$

$$RT1Max_v \geq rt1_v \qquad \forall\, v \in V \tag{5.2}$$

$$RT2Min_v \leq rt2_v \qquad \forall\, v \in V \tag{5.3}$$

$$RT2Max_v \geq rt2_v \qquad \forall\, v \in V \tag{5.4}$$

$$rt1_v \leq rt2_v \qquad \forall\, v \in V \tag{5.5}$$

$$EVR_v - rt1_v \leq bve1_v \qquad \forall\, v \in V \tag{5.6}$$

$$rt1_v - EVR_v \leq (1 - bve1_v)\ M \qquad \forall\, v \in V \tag{5.7}$$

$$EVR_v - rt2_v \leq bve2_v\ M \qquad \forall\, v \in V \tag{5.8}$$

$$rt2_v - EVR_v \leq (1 - bve2_v)\ M \qquad \forall\, v \in V \tag{5.9}$$

$$bve2_v \leq bve1_v \qquad \forall\, v \in V \tag{5.10}$$

$$HVR_v - rt1_v \leq bvh1_v\ M \qquad \forall\, v \in V \tag{5.11}$$

$$rt1_v - HVR_v \leq (1 - bvh1_v)\ M \qquad \forall\, v \in V \tag{5.12}$$

$$HVR_v - rt2_v \leq bvh2_v\ M \qquad \forall\, v \in V \tag{5.13}$$

$$rt2_v - HVR_v \leq (1 - bvh2_v)\ M \qquad \forall\, v \in V \tag{5.14}$$

$$bvh2_v \leq bvh1_v \qquad \forall\, v \in V \tag{5.15}$$

$$VPF1Min_v \leq vpf1_v \qquad \forall\, v \in V \tag{5.16}$$

$$VPF1Max_v \geq vpf1_v \qquad \forall\, v \in V \tag{5.17}$$

$$VPF2Min_v \leq vpf2_v \qquad \forall\, v \in V \tag{5.18}$$

$$VPF2Max_v \geq vpf2_v \qquad \forall\, v \in V \tag{5.19}$$

$$VPF3Min_v \leq vpf3_v \qquad \forall\, v \in V \tag{5.20}$$

$$VPF3Max_v \geq vpf3_v \qquad \forall\, v \in V \tag{5.21}$$

$$vpf1_v \leq vpf2_v \qquad \forall\, v \in V \tag{5.22}$$

$$vpf2_v \leq vpf3_v \qquad \forall\, v \in V \tag{5.23}$$

$$
\begin{aligned}
&(TVR_v - rt2_v)\ vpf3_v + (rt2_v - rt1_v)\ vpf2_v + rt1_v\ vpf1_v \geq \\
&(TVR_v - ORT2_v)\ OVPF3_v + (ORT2_v - ORT1_v)\ OVPF2_v + \\
&ORT1_v\ OVPF1_v \qquad \forall\, v \in V
\end{aligned} \tag{5.24}
$$

$$
\begin{aligned}
&(HVR_v - rt2_v)\ vpf3_v\ bvh2_v + \\
&(rt2_v - rt1_v)\ vpf2_v\ bvh2_v + rt1_v\ vpf1_v\ bvh2_v + \\
&(HVR_v - rt1_v)\ vpf2_v\ (1 - bvh2_v)\ bvh1_v + \\
&rt1_v\ vpf1_v\ (1 - bvh2_v)\ bvh1_v + \\
&HVR_v\ vpf1_v\ (1 - bvh2_v)\ (1 - bvh1_v) \geq \\
&(HVR_v - ORT2_v)\ OVPF3_v\ BPH2_v + \\
&(ORT2_v - ORT1_v)\ OVPF2_v\ BPH2_v + ORT1_v\ OVPF1_v\ BPH2_v + \\
&(HVR_v - ORT1_v)\ OVPF2_v\ (1 - BPH2_v)\ BPH1_v + \\
&ORT1_v\ OVPF1_v\ (1 - BPH2_v)\ BPH1_v + \\
&HVR_v\ OVPF1_v\ (1 - BPH2_v)\ (1 - BPH1_v) \qquad \forall\, v \in V
\end{aligned} \tag{5.25}
$$

$$
\begin{aligned}
&(EVR_v - rt2_v)\ vpf3_v\ bve2_v + \\
&(rt2_v - rt1_v)\ vpf2_v\ bve2_v + rt1_v\ vpf1_v\ bve2_v + \\
&(EVR_v - rt1_v)\ vpf2_v\ (1 - bve2_v)\ bve1_v + \\
&rt1_v\ vpf1_v\ (1 - bve2_v)\ bve1_v + \\
&EVR_v\ vpf1_v\ (1 - bve2_v)\ (1 - bve1_v) \leq \\
&(EVR_v - ORT2_v)\ OVPF3_v\ BPE2_v + \\
&(ORT2_v - ORT1_v)\ OVPF2_v\ BPE2_v + ORT1_v\ OVPF1_v\ BPE2_v + \\
&(EVR_v - ORT1_v)\ OVPF2_v\ (1 - BPE2_v)\ BPE1_v + \\
&ORT1_v\ OVPF1_v\ (1 - BPE2_v)\ BPE1_v + \\
&EVR_v\ OVPF1_v\ (1 - BPE2_v)\ (1 - BPE1_v) \qquad \forall\, v \in V
\end{aligned} \tag{5.26}
$$

$$vpf1_v,\ vpf2_v,\ vpf3_v \in \mathbb{R} \qquad \forall\, v \in V \tag{5.27}$$

$$rt1_v,\ rt2_v \in \mathbb{N} \qquad \forall\, v \in V \tag{5.28}$$

$$bve1_v,\ bve2_v,\ bvh1_v,\ bvh2_v\ \in \{0, 1\} \qquad \forall\, v \in V \tag{5.29}$$

The objective of the retailer is to minimize the future expected cost of penalties that will be charged by all vendor. Since the penalty structure is multi-layered, increasing penalties between increasing return thresholds, the objective function is a piece-wise non-linear cost function where the retailer tries to minimize the penalties between these thresholds for the expected future returns. The MINLP involves combined vendor penalty fees and return threshold decisions simultaneously and therefore they are all considered as decision variables. The goal is to identify the optimal vendor penalty fee structure with the best possible return threshold schema.

Constraints (5.1) and 5.2 ensure that the vendor return threshold 1 should be between a given range by the vendor. **Constraints** (5.3) and 5.4 ensure that the vendor return threshold 2 should be between a given range by the vendor. **Constraint** (5.5) ensures that the vendor return threshold 2 is greater than the vendor return threshold 1 for every vendor. **Constraints** (5.6) and 5.7 ensure that the expected vendor return is greater than vendor return threshold 1 by using a dummy binary variable of the expected return for every vendor. **Constraints** (5.8) and 5.9 ensure that the expected vendor return is greater than vendor return threshold 2 by using a dummy binary variable of the expected return for every vendor. **Constraint** (5.10) ensures the logical correctness between dummy binary variables of the expected vendor return for every vendor. **Constraints** (5.11) and 5.12 ensure that the last vendor return is greater than the vendor return threshold 1 by using a dummy binary variable of the last vendor return for every vendor. **Constraints** (5.13) and 5.14 ensure that the last vendor return is greater than the vendor return threshold 2 by using a dummy binary variable of the last vendor return for every vendor. **Constraint** (5.15) ensures the logic between dummy binary variables of the last vendor return for every vendor. **Constraints** (5.16) and (5.17) ensure that vendor penalty fee / percentage 1 should be between a given range by the vendor. **Constraints** (5.18) and (5.19) ensure that vendor penalty fee / percentage 2 should be between a given range by the vendor. **Constraints** (5.20) and (5.21) ensure that vendor penalty fee / percentage 2 should be between a given range by the vendor. **Constraint** (5.22) ensures that ensures that vendor penalty fee / percentage 2 percentage is greater than vendor penalty fee / percentage 1 **Constraint** (5.23) ensure that vendor penalty fee / percentage 3 percentage is greater than vendor penalty fee / percentage 2. **Constraint** (5.24) ensure that if the retailer uses all of its available vendor funds for every vendor, the vendor will make more money with the new threshold structure compared to the old threshold structure. This constraint ensures that the vendor will make at least the same amount of money, if the retailer makes a huge return in the future and it exhausts all the available vendor funds. This constraint is one of the main convincing factors of the new penalty fee and return threshold structure for contract re-negotiation. **Constraint** (5.25) ensures that if the retailer makes the same return as the historical returns, the new penalty amount that will be paid to the vendor would be more than historical penalty amount. This constraint ensures the vendor to make at least the same amount of money historically in case similar returns happen in the future. **Constraint** (5.26) ensures that when the retailer makes the expected / forecasted return, the new penalty amount would be less than the penalty amount which would have been paid when the the original threshold structure was to be kept as is. This constraint is the main constraint that ensures the retailer to pay less penalty for future returns compared what could have been paid if existing penalty fee and return threshold schema exist as is. **Constraint** (5.27)–(5.29) declares the set of continuous, integer, and binary variables, respectively.

The model considers the multi-layered penalty structure of vendors when returning products back to their vendor. The retailer has to come up with an initial contract parameter schema in order to re-negotiate its existing product returns contracts with all of its vendors and try to convince them that the proposed penalty fees and return thresholds can be beneficial for them via (in the case of most likely scenario, historical returns, or extreme scenario, total consumption of available funds, happens) ensuring the potential outcomes of the newly proposed structure. The retailer proposes these new contract terms via considering its important internal data and information and vendors NOT knowing the expected future returns of the retailer. Since the expected vendor returns can only be forecasted on the retailer's end by using internal sales and inventory data, vendors have limited knowledge what to expect in the future except for most likely and extreme scenarios. Therefore, retail is leveraging its own source of internal data, information and analytics capabilities to get a head in the re-negotiation of its product returns contracts.

5.3 Test Problems, Computations and Overall Numerical Analysis

In this section, we present the results to the test problems we have solved via the existing MINLP solvers in GAMS Studio 1.8.

For our test problems, we have generated a set of vendors with related data. The test scenario problems consisted from small size problems, 5 vendors, to large size problems, 2500 vendors. The penalty schema of a vendor consisted of 2 return thresholds, a total available vendor fund amount and therefore a 3-layer increasing penalty fee structure for each vendor.

The the details of the problems we have generated can be observed in *Table 5.1*. The smallest size problem we have generated included 5 vendors which generated 45 decision variables (10 integer variables, 20 binary variables and 15 continuous variables) and 130 constraints (115 linear constraints and 15 nonlinear constraints). The largest size problem we have generated included 2500 vendors which generated 22500 decision variables (5000 integer variables, 10000 binary variables and 7500 continuous variables) and 65000 constraints (57500 linear constraints and 7500 nonlinear constraints).

Problem parameters such as total available vendor funds, minimum and maximum penalty fees, and minimum and maximum return thresholds of the vendor are generated with random data that fits a uniform distribution between certain bounds (lower and upper limit) that are consistent with values of the related parameters. Expected and historical return amounts are also generated randomly that fits a uniform distribution between 0 and total available vendor funds amount of the vendor. This allowed us generate all possible combinations of distinct test problems to test the performance of the MINLP solvers.

The combined vendor penalty fee and return thresholds parameter optimization problem defined in *Section 5.2* is a Mixed Integer Non-Linear Program (MINLP). The MINLP is coded using a commercial exact solver software, GAMS Studio 1.8, on a PC with an Intel(R) Xeon(R) CPU E5-2680 0 @ 2.70GHz, 2700 Mhz, 1 Core(s), 1 Logical Processor(s) and 32 GB of RAM.

5.3.1 Test Case Problem Sizes and MINLP Solver Solution Times

In total, we have generated 20 test problems and these problems included small size to large size problems. The details of the these test problems can be found in *Table 5.1*. In order to observe the performance of the available MINLP solvers on these test problems, these test problems were given to all of the available MINLP solvers (AlphaECP, ANTIGONE, BARON, Bonmin, Knitro, DICOPT, LINDOGlobal and SBB) separately to solve and compare the results using GAMS Studio 1.8. Problems were asked to give solutions within 0.01% optimality range. Unfortunately, AlphaECP, BARON, Bonmin, Knitro, LINDOGlobal and SBB were not able to solve even small size problems (problems greater than 10 vendors which generated at least 90 decision variables and 260 constraints) within reasonable solution times (more than 120 minutes, 2 hours, were given to all MINLP solvers). Therefore, the results of these MINLP solvers were not included in *Table 5.1* and only successful solver results were included in the summary report.

DICOPT and ANTIGONE MINLP solvers were the only 2 solvers that were able to solve small and larger size problems within reasonable solution times. ANTIGONE solver's performance also dropped significantly for mid-size problems (problems that have more than 250 vendors which generated at least 2250 decision variables and 6500 constraints). The results of the DICOPT and ANTIGONE solvers to given test problems can be observed in *Table 5.1*.

As can be observed from *Table 5.1*, DICOPT was able to solve all the problems, small to large size problems, within reasonable solution times. Small size problems, 5-100 vendors took less than a second to solve. Larger size problems, up to 2500 vendors, DICOPT was able to find a good solutions within 20 minutes. ANTIGONE also performed well for smaller size problems and was able to find good solution within reasonable solution times. However, when the problem size increased, problems with more than 250 vendors, the performance of the solver dropped significantly and was not able to find a solution within 2 hours given to the solver.

5.3.2 Results and Insights

In this section we present summaries of our numerical study. More detailed numerical results are included in Appendix A3. Based on the 25 test problems we have performed from small size to larger size problems, we can observe that the combined effect of optimal vendor penalty fees and return thresholds decisions can save the retailer 5-7 % (*Table 5.1*) in terms of penalties that would be paid to its vendors for returning products. The results suggest that re-negotiating the existing contract parameters, with the optimal parameters calculated through our model, can have significant savings for the retailer in terms of the penalty it pays to all of its vendors for its product returns in the upcoming return cycles. Since the optimal contract parameters that were identified on our model guarantees the vendors to charge at least the same amount of penalty for historical returns (most likely scenario) or in the case of the usage of all available vendor funds (worst case scenario), convincing the vendors to accept the retailer's offers is to be expected.

As can be observed from *Table 5.1*, even the best MINLP solver that was able to solve this problem, DICOPT, was not able to find an optimal solution within the given solution time (1 hour) for larger size problems (problems over 4500 Vendors). In order to find a 'good' solution within reasonable times and observe the inner workings of the linear models to develop heuristics and/or certain solution mechanisms, the original MINLP was decomposed into 2 separate problems. First problem is modelled as an LP to find an optimal vendor penalty fee schema while keeping the original return thresholds as is, and the second problem is modelled as a MILP to find an optimal return threshold schema while keeping the original penalty fee structure as is. The models and detailed analysis (rule-of-thumb mechanisms) of the solutions are presented in the following subsections.

The optimal penalty and threshold schema solutions of a sample problem that has 50 vendors can be observed in *Table A1.1*. These solutions are the optimal solutions that are identified by the DICOPT solver. As can be found from the results in *Table A1.1*, the optimal penalty fee and return threshold schema, penalty savings with an individual vendor and aggregate penalty savings from all vendors can be identified from this chart.

													MINLP SOLVER									
													DICOPT					ANTIGONE				
Test Case	Number of Vendors	Total Number of Decision Variables	Number of Integer Decision Variables	Number of Binary Decision Variables	Number of Continuous Decision Variables	Number of Constraints	Number of Linear Constraints	Number of Nonlinear Constraints	Optimality Criteria	Time Limit Given (minutes)	Optimal Solution Found within Given Time?	Total Vendor Penalty (Old / Original)	Total Vendor Penalty (New / Optimal)	Difference (Savings)	Difference Percentage (Savings)	Solution Time (minutes)	Solution Time (seconds)	Total Vendor Penalty (New / Optimal)	Difference (Savings)	Difference Percentage (Savings)	Solution Time (minutes)	Solution Time (seconds)
Test 01	5	45	10	20	15	130	115	15	0.00%	60	Yes	$ 771,136	$ 741,244	$ 29,892	3.88%	0.02	1.49	$ 742,020	$ 29,116	3.78%	0.02	1.46
Test 02	10	90	20	40	30	260	230	30	0.00%	60	Yes	$ 1,182,030	$ 1,139,953	$ 42,077	3.56%	0.01	0.59	$ 1,145,138	$ 36,892	3.12%	0.07	3.97
Test 03	20	180	40	80	60	520	460	60	0.00%	60	Yes	$ 2,382,545	$ 2,287,755	$ 94,790	3.98%	0.04	2.52	$ 2,287,295	$ 95,250	4.00%	0.09	5.25
Test 04	50	450	100	200	150	1,300	1,150	150	0.00%	60	Yes	$ 5,042,280	$ 4,790,255	$ 252,025	5.00%	0.06	3.52	$ 4,795,205	$ 247,075	4.90%	0.50	30.25
Test 05	75	675	150	300	225	1,950	1,725	225	0.00%	60	Yes	$ 6,519,842	$ 6,142,815	$ 377,027	5.78%	0.07	4.33	$ 6,123,569	$ 396,273	6.08%	0.70	41.92
Test 06	100	900	200	400	300	2,600	2,300	300	0.00%	60	Yes	$ 8,622,183	$ 8,115,544	$ 506,639	5.88%	0.08	4.64	$ 8,095,243	$ 526,939	6.11%	2.02	121.23
Test 07	250	2,250	500	1,000	750	6,500	5,750	750	0.00%	60	Yes	$ 21,749,494	$ 20,432,346	$ 1,317,148	6.06%	0.18	10.83	N/A	N/A	N/A	> 60	> 3600
Test 08	500	4,500	1,000	2,000	1,500	13,000	11,500	1,500	0.00%	60	Yes	$ 45,169,890	$ 42,343,432	$ 2,826,458	6.26%	0.51	30.70	N/A	N/A	N/A	> 60	> 3600
Test 09	750	6,750	1,500	3,000	2,250	19,500	17,250	2,250	0.00%	60	Yes	$ 64,007,620	$ 60,077,583	$ 3,930,037	6.14%	1.25	75.05	N/A	N/A	N/A	> 60	> 3600
Test 10	1,000	9,000	2,000	4,000	3,000	26,000	23,000	3,000	0.00%	60	Yes	$ 88,776,523	$ 83,697,013	$ 5,079,510	5.72%	2.07	124.47	N/A	N/A	N/A	> 60	> 3600
Test 11	1,250	11,250	2,500	5,000	3,750	32,500	28,750	3,750	0.00%	60	Yes	$ 110,890,193	$ 103,696,683	$ 7,193,510	6.49%	3.14	188.30	N/A	N/A	N/A	> 60	> 3600
Test 12	1,500	13,500	3,000	6,000	4,500	39,000	34,500	4,500	0.00%	60	Yes	$ 135,294,712	$ 126,750,051	$ 8,544,662	6.32%	5.67	340.13	N/A	N/A	N/A	> 60	> 3600
Test 13	1,750	15,750	3,500	7,000	5,250	45,500	40,250	5,250	0.00%	60	Yes	$ 158,940,340	$ 148,788,870	$ 10,151,470	6.39%	7.70	461.94	N/A	N/A	N/A	> 60	> 3600
Test 14	2,000	18,000	4,000	8,000	6,000	52,000	46,000	6,000	0.00%	60	Yes	$ 180,908,013	$ 169,386,063	$ 11,521,949	6.37%	10.68	640.56	N/A	N/A	N/A	> 60	> 3600
Test 15	2,000	18,000	4,000	8,000	6,000	52,000	46,000	6,000	0.00%	60	Yes	$ 182,452,766	$ 170,365,431	$ 12,087,335	6.62%	11.39	683.58	N/A	N/A	N/A	> 60	> 3600
Test 16	2,000	18,000	4,000	8,000	6,000	52,000	46,000	6,000	0.00%	60	Yes	$ 180,355,445	$ 168,539,152	$ 11,816,293	6.55%	10.07	604.13	N/A	N/A	N/A	> 60	> 3600
Test 17	2,000	18,000	4,000	8,000	6,000	52,000	46,000	6,000	0.00%	60	Yes	$ 178,000,931	$ 166,052,936	$ 11,947,994	6.71%	11.38	682.61	N/A	N/A	N/A	> 60	> 3600
Test 18	2,000	18,000	4,000	8,000	6,000	52,000	46,000	6,000	0.00%	60	Yes	$ 183,654,016	$ 171,446,211	$ 12,207,806	6.65%	11.39	683.64	N/A	N/A	N/A	> 60	> 3600
Test 19	2,250	20,250	4,500	9,000	6,750	58,500	51,750	6,750	0.00%	60	Yes	$ 205,363,423	$ 192,570,036	$ 12,793,387	6.23%	12.32	739.48	N/A	N/A	N/A	> 60	> 3600
Test 20	2,500	22,500	5,000	10,000	7,500	65,000	57,500	7,500	0.00%	60	Yes	$ 229,172,550	$ 214,937,415	$ 14,235,135	6.21%	17.82	1,068.95	N/A	N/A	N/A	> 60	> 3600
Test 21	3,000	27,000	6,000	12,000	9,000	78,000	69,000	9,000	0.00%	60	Yes	$ 275,932,028	$ 257,759,960	$ 18,172,068	6.59%	15.17	910.36	N/A	N/A	N/A	> 60	> 3600
Test 22	3,500	31,500	7,000	14,000	10,500	91,000	80,500	10,500	0.00%	60	Yes	$ 319,588,834	$ 299,097,852	$ 20,490,982	6.41%	19.55	1,173.09	N/A	N/A	N/A	> 60	> 3600
Test 23	4,000	36,000	8,000	16,000	12,000	104,000	92,000	12,000	0.00%	60	Yes	$ 353,180,389	$ 329,372,170	$ 23,808,219	6.74%	26.23	1,573.53	N/A	N/A	N/A	> 60	> 3600
Test 24	4,500	40,500	9,000	18,000	13,500	117,000	103,500	13,500	0.00%	60	No	$ 402,598,114	N/A	N/A	N/A	> 60	> 3600	N/A	N/A	N/A	> 60	> 3600
Test 25	5,000	45,000	10,000	20,000	15,000	130,000	115,000	15,000	0.00%	60	No	$ 446,641,688	N/A	N/A	N/A	> 60	> 3600	N/A	N/A	N/A	> 60	> 3600

TABLE 5.1: Results of Test Problems for the Vendor Penalty Schema and Return Threshold Model

5.4 LP for the Vendor Penalty Fees / Percentages

$$\sum_{v=1}^{V} \begin{cases} vpf1_v \, EVR_v & \text{if } 0 \leq EVR_v \leq ORT1_v \\ vpf1_v \, ORT1_v + vpf2_v \, (EVR_v - ORT1_v) & \text{if } ORT1_v < EVR_v \leq ORT2_v \\ vpf1_v \, ORT1_v + vpf2_v \, (ORT2_v - ORT1_v) & \\ + \, vpf3_v \, (EVR_v - ORT2_v) & \text{if } ORT2_v < EVR_v \leq TVR_v \end{cases}$$

s.t. *Constraints* 5.16 *to* 5.23,

Constraints 5.24 *to* 5.26 where $rt._v$ variables are replaced with $ORT._v$ parameters,

Constraint 5.27

The objective of the retailer is to minimize its future expected cost of penalties that will be charged to the retailer by all of its vendor. Since the penalty structure is multi-layered, increasing penalties between increasing return thresholds, the objective function is a piece-wise linear cost function where the retailer tries to minimize the penalties between these thresholds for the expected future returns. The cosntraints have already been defined in constraints in *Subsection 5.2*.

The model considers the multi-layer penalty structure of vendors when returning products back to their vendor. The retailer has to come up with a negotiation strategy with its vendors in order to convince them that the proposed penalty structure can be beneficial for them via (in the case of most likely scenario, historical returns, or extreme scenario, total consumption of available funds, happens) ensuring the potential outcomes of the newly proposed penalty schema. The retailer proposes these new contract terms via considering its important internal data and information and vendors NOT knowing the expected future returns of the retailer. Since expected vendor returns can only be forecasted on the retailer's end by using internal sales and inventory data, vendors have limited knowledge what to expect in the future except for most likely and extreme scenarios. Therefore, retail is leveraging its own source of internal data, information and analytics capabilities to get a head in the negotiation curve.

5.5 MILP for the Vendor Return Thresholds

$$\sum_{v=1}^{V} \begin{cases} OVPF1_v\, EVR_v & \text{if } 0 < EVR_v \leq RT1Min_v \\ OVPF1_v\, rt1_v\, bve1_v \\ + OVPF2_v\, (EVR_v - rt1_v)\, bve1_v \\ + OVPF1_v\, EVR_v\, (1 - bve1_v) & \text{if } RT1Min_v < EVR_v \leq RT1Max_v \\ OVPF1_v\, rt1_v \\ + OVPF2_v\, (EVR_v - rt1_v) & \text{if } RT1Max_v < EVR_v \leq RT2Min_v \\ OVPF1_v\, rt1_v \\ + OVPF2_v\, (EVR_v - rt1_v)\, (1 - bve2_v) \\ + OVPF2_v\, (rt2_v - rt1_v)\, bve2_v \\ + OVPF3_v\, (EVR_v - rt2_v)\, bve2_v & \text{if } RT2Min_v < EVR_v \leq RT2Max_v \\ OVPF1_v\, rt1_v \\ + OVPF2_v\, (rt2_v - rt1_v) \\ + OVPF3_v\, (EVR_v - rt2_v) & \text{if } RT2Max_v < EVR_v \leq TVR_v \end{cases}$$

s.t. *Constraints* 5.1 *to* 5.15,

Constraints 5.24 *to* 5.26 where $vpf._v$ variables are replaced with $OVPF._v$ parameters,

Constraints 5.28 *to* 5.29

Similar to the previous model in *Section 5.4*, the objective of the retailer is to minimize its future expected cost of penalties that will be charged to the retailer by all of its vendor. Since the penalty structure is multi-layered, increasing penalties between increasing return thresholds, the objective function is a piece-wise linear cost function where the retailer tries to minimize the penalties between these thresholds for the expected future returns. However, compared to the model described in *Section 5.4*, return thresholds are considered to be decision variables. The definitions of the related constraints in *Subsection 5.2*.

The model considers the multi-layer penalty structure of vendors when returning products back to their vendor. The retailer has to come up with a negotiation strategy with its vendors in order to convince them that the proposed return threshold structure can be beneficial for them via (in the case of most likely scenario, historical returns, or extreme scenario, total consumption of available funds, happens) ensuring the potential outcomes of the newly proposed return threshold schema. The retailer proposes these new contract terms via considering its important internal data and information and vendors NOT knowing the expected future returns of the retailer. Since expected vendor returns can only be forecasted on the retailer's end by using internal sales and inventory data, vendors have limited knowledge what to expect in the future except for most likely and extreme scenarios. Therefore, retail is leveraging its own source of internal data, information and analytics capabilities to get a head in the negotiation curve.

5.6 Test Problems, Computations and Overall Numerical Analysis

In this section, we present our findings about the test problems we have solved using CPLEX for both of the models in *Section 5.4 and 5.5*, deep dive into the results for cost savings and to identify insights for developing mechanisms to extract rule-of-thumb methodologies to derive exact or approximate solutions close to optimal penalty percentages and return thresholds via identifying all possible scenarios that can exist in the problem structures.

For our test problems, we have generated a set of vendors with all the related data and the test scenario problems consisted of 5000 vendors with 2 return thresholds within a total available vendor fund amount and therefore a 3-layer increasing penalty fee / percentage structure for each vendor. The above mentioned model size test scenarios resulted in a 15000 float decision variable and 55000 constraint size problem for the penalty fee structure model and a 10000 float decision variable, 20000 binary variables, a total of 30000 decision variables, and 90000 constraint size problem for the return threshold model. Problem parameters such as penalty fees / percentages, return thresholds, total available vendor funds, minimum and maximum penalty fees and minimum and maximum return thresholds of the vendor are generated with random data that fits a uniform distribution between certain bounds (lower and upper limit) that are consistent with values of the related parameters. Expected and historical return amounts are also generated randomly that fits a uniform distribution between 0 and total available vendor funds amount of the vendor. This allowed us generate all possible combinations of test problems that can happen in the real world in order to create all distinct scenarios and investigate the results in detail.

The above parameter optimization of the vendor contract terms models are a mixed integer linear program (MILP) and are designed and coded with an exact solver software, IBM ILOG CPLEX Optimization Studio 12.8, on a PC with an Intel Core i7-8550U CPU @ 1.8 GHz, 4 Cores, 8 Logical Processors and 20 GB of RAM.

In total, we generated 20 test problems for each model and solved using an exact solver, CPLEX. Optimal solutions to the linear problems were found in a very short amount of time, mostly within a couple of seconds. Problem size, vendor count, number of decision variables and constraints, solver solution times, historical penalties, expected penalties and hence expected savings in terms of value and percentage were observed and reported in *Table 5.2* for the optimal vendor penalty schema model and *Table 5.3* for the optimal vendor return threshold model.

Test Case	Number of Vendors	Number of Decision Variables	Number of Constraints	Problem Solution Time (seconds)	Total Solution Time (seconds)	Total Vendor Penalty (New / Optimal)	Total Vendor Penalty (Old / Original)	Difference (Savings)	Difference Percentage (Savings)
Test 01	5,000	15,000	55,000	0.05	25.50	$ 434,723,268	$ 445,777,616	$ 11,054,348	2.48%
Test 02	5,000	15,000	55,000	0.08	34.29	$ 435,110,238	$ 445,991,175	$ 10,880,936	2.44%
Test 03	5,000	15,000	55,000	0.06	33.65	$ 436,924,037	$ 447,478,247	$ 10,554,210	2.36%
Test 04	5,000	15,000	55,000	0.06	34.09	$ 434,786,741	$ 445,741,772	$ 10,955,031	2.46%
Test 05	5,000	15,000	55,000	0.03	30.91	$ 433,373,905	$ 444,395,448	$ 11,021,543	2.48%
Test 06	5,000	15,000	55,000	0.03	36.05	$ 429,516,707	$ 440,301,365	$ 10,784,658	2.45%
Test 07	5,000	15,000	55,000	0.03	35.86	$ 427,054,631	$ 438,211,082	$ 11,156,451	2.55%
Test 08	5,000	15,000	55,000	0.02	34.79	$ 439,768,099	$ 450,410,590	$ 10,642,491	2.36%
Test 09	5,000	15,000	55,000	0.05	35.61	$ 427,851,614	$ 438,452,743	$ 10,601,128	2.42%
Test 10	5,000	15,000	55,000	0.05	31.61	$ 438,871,557	$ 449,754,185	$ 10,882,628	2.42%
Test 11	5,000	15,000	55,000	0.06	33.53	$ 431,441,554	$ 442,603,467	$ 11,161,913	2.52%
Test 12	5,000	15,000	55,000	0.06	35.22	$ 433,454,465	$ 444,238,791	$ 10,784,326	2.43%
Test 13	5,000	15,000	55,000	0.06	38.16	$ 427,793,585	$ 438,549,904	$ 10,756,319	2.45%
Test 14	5,000	15,000	55,000	0.03	35.75	$ 435,897,152	$ 446,916,089	$ 11,018,937	2.47%
Test 15	5,000	15,000	55,000	0.06	36.60	$ 428,982,142	$ 439,612,379	$ 10,630,237	2.42%
Test 16	5,000	15,000	55,000	0.08	34.44	$ 440,043,127	$ 451,052,919	$ 11,009,792	2.44%
Test 17	5,000	15,000	55,000	0.06	35.58	$ 431,353,305	$ 442,261,281	$ 10,907,976	2.47%
Test 18	5,000	15,000	55,000	0.05	37.39	$ 425,558,569	$ 436,320,169	$ 10,761,600	2.47%
Test 19	5,000	15,000	55,000	0.05	32.70	$ 432,920,999	$ 443,725,110	$ 10,804,111	2.43%
Test 20	5,000	15,000	55,000	0.02	33.38	$ 437,915,688	$ 448,781,262	$ 10,865,574	2.42%

TABLE 5.2: Test Problem Results for the Vendor Penalty Fee Schema Model

Test Case	Number of Vendors	Total Number of Decision Variables	Number of Float Variables	Number of Binary Variables	Number of Constraints	Problem Solution Time (seconds)	Total Solution Time (seconds)	Total Vendor Penalty (New / Optimal)	Total Vendor Penalty (Old / Original)	Difference (Savings)	Difference Percentage (Savings)
Test 01	5,000	30,000	10,000	20,000	90,000	0.31	30.89	$ 446,706,178	$ 454,989,212	$ 8,283,034	1.82%
Test 02	5,000	30,000	10,000	20,000	90,000	0.58	44.66	$ 452,471,933	$ 460,334,062	$ 7,862,129	1.71%
Test 03	5,000	30,000	10,000	20,000	90,000	0.64	38.57	$ 432,878,340	$ 440,722,866	$ 7,844,527	1.78%
Test 04	5,000	30,000	10,000	20,000	90,000	0.64	44.05	$ 438,332,686	$ 446,213,423	$ 7,880,737	1.77%
Test 05	5,000	30,000	10,000	20,000	90,000	0.34	40.82	$ 438,838,673	$ 447,009,855	$ 8,171,182	1.83%
Test 06	5,000	30,000	10,000	20,000	90,000	0.64	39.73	$ 437,979,904	$ 446,294,671	$ 8,314,767	1.86%
Test 07	5,000	30,000	10,000	20,000	90,000	0.61	40.58	$ 433,989,004	$ 442,280,894	$ 8,291,891	1.87%
Test 08	5,000	30,000	10,000	20,000	90,000	0.30	36.16	$ 427,359,378	$ 435,958,248	$ 8,598,870	1.97%
Test 09	5,000	30,000	10,000	20,000	90,000	0.64	38.13	$ 438,938,185	$ 447,015,598	$ 8,077,413	1.81%
Test 10	5,000	30,000	10,000	20,000	90,000	0.44	38.12	$ 431,543,906	$ 440,235,149	$ 8,691,243	1.97%
Test 11	5,000	30,000	10,000	20,000	90,000	0.31	38.23	$ 435,567,252	$ 443,772,626	$ 8,205,374	1.85%
Test 12	5,000	30,000	10,000	20,000	90,000	0.33	37.56	$ 440,631,876	$ 448,763,116	$ 8,131,241	1.81%
Test 13	5,000	30,000	10,000	20,000	90,000	0.64	40.37	$ 436,821,262	$ 445,150,669	$ 8,329,407	1.87%
Test 14	5,000	30,000	10,000	20,000	90,000	0.33	38.13	$ 446,894,993	$ 455,065,046	$ 8,170,053	1.80%
Test 15	5,000	30,000	10,000	20,000	90,000	0.61	39.09	$ 434,520,364	$ 442,347,344	$ 7,826,980	1.77%
Test 16	5,000	30,000	10,000	20,000	90,000	0.30	39.54	$ 436,020,063	$ 444,211,869	$ 8,191,807	1.84%
Test 17	5,000	30,000	10,000	20,000	90,000	0.61	39.59	$ 427,239,304	$ 434,878,156	$ 7,638,853	1.76%
Test 18	5,000	30,000	10,000	20,000	90,000	0.31	39.42	$ 438,923,892	$ 447,126,789	$ 8,202,897	1.83%
Test 19	5,000	30,000	10,000	20,000	90,000	0.67	39.11	$ 434,528,761	$ 443,310,476	$ 8,781,715	1.98%
Test 20	5,000	30,000	10,000	20,000	90,000	0.70	46.93	$ 432,685,508	$ 440,747,910	$ 8,062,403	1.83%

TABLE 5.3: Test Problem Results for the Vendor Return Thresholds Model

5.6.1 Results and Insights

Using the 20 optimal penalty schema test problems that are generated and solved, on aggregate, we have observed that identifying the optimal vendor penalty schema strategy can save the retailer a minimum of 2.36%, a maximum of 2.55% and with an average of 2.45% on penalty costs (*Table 5.2*). Using the 20 optimal return threshold test problems that are generated and solved, on aggregate, we have observed that identifying the optimal return thresholds strategy can save the retailer a minimum of 1.71%, a maximum of 1.97%, with an average of 1.85% on penalty costs (*Table 5.3*). These savings could increase or decrease based on the bounds, minimum and maximum limit of the parameters that would be allowed by the vendor. Since we have generated the lower and upper bounds of the penalty and threshold limits for each vendor based on some randomly generated values between a certain range, this range affects the potential savings we would have on the solution. A looser range of penalties between return thresholds (or looser range of return threshold while keeping the same penalty structure) could allow more savings in penalties to be paid since better solutions could be identified via more possibilities. With a closer look at the savings of the individual vendors for the penalty schema problem, we observed a more detailed picture of the inner workings of the model. For the 1000 vendors that were generated on each problem, the savings for each vendor ranged from 0% up to 90% on some cases. In order to identify the discrepancy of the cases, we have partitioned the problem structure into 9 different scenarios (*Table 5.4*).

These scenarios can be described with historical returns and expected returns being in between different return threshold buckets such as; when historical return being less than return threshold 1 vs. expected return being greater than return threshold 2, historical return being greater than return threshold 2 vs. expected return being less than return threshold 1 and so on... These scenario definitions can also be observed in detail in *Table 5.4*. Since we have two mid-thresholds and a total available vendor funds being the highest and the third threshold in our test problems, this problem structure created 9 possible scenarios to exist in our results. We have observed the most significant savings realized on scenarios when historical returns was greater than return threshold 2 and expected returns were less than return threshold 1. However, even though there were some scenarios with huge savings potential (up to 90%), there were also scenarios with little savings (down to 4%) for the retailer. We have observed that the low savings were due to fact that random data generation generated tighter bounds and huge savings potential was due to fact that random data generation generated looser bounds for those vendor. If vendors have looser penalty boundaries, then expected future returns which belong to this scenario could generate huge savings for the retailer.

CASES	0 <= EVR <= ORT1	ORT1 < EVR <= ORT2	ORT2 < EVR <= TVR
0 <= HVR <= ORT1	Scenario 1	Scenario 2	Scenario 3
ORT1 < HVR <= ORT2	Scenario 4	Scenario 5	Scenario 6
ORT2 < HVR <= TVR	Scenario 7	Scenario 8	Scenario 9

TABLE 5.4: Distinct Scenarios that Exist in Vendor Penalty Schema Problem

We have also observed that (from scenarios shown in *Table 5.4*) where historical return being less than return threshold 1 vs. expected return being less than return threshold 1 (Scenario 1), the model could not identify any better penalty schema where the retailer pays less penalty. Even though, based on the constraints provided, the model identified a different penalty schema for most of the scenarios, it did not result in penalty savings on retailer's end. There were also similar non-cost-saving cases in scenario where historical return being greater than return threshold 2 vs. expected return being greater than return threshold 2 and expected return being greater then the historical return (in Scenario 9 where $EVR_v > HVR_v$). However, under this scenario, for only half of the cases this argument was true. This is due to the fact that expected returns were slightly higher than expected return under this scenario. We provide a snapshot of one of the sample problems' solution on *Table A1.2*. We can observe some of the intricacies of the model's optimal solution behaviour under different scenarios (*Table 5.4*) on this chart and generate a rule-of-thumb mechanisms to generate faster results. The main reason to generate these rule-of-thumb mechanisms is to gain insights from the solution structure that is extracted from the optimal solutions, generate a template of the possible outcomes as a rule set and eventually share these rules with the category managers who deal with their specific vendor portfolio on a daily basis for a fast and reliable decision making approach. Using such a mechanism, they would not need to the rely on other analysts/scientists to model this complex problem on their behalf and provide them with the optimal parameters for a specific vendor's contract re-negotiation. Therefore, such mechanisms are highly efficient and needed all around the retail organization for faster decision making process.

The chart in *Table 5.5* outlines the rule-of-thumbs for every possible scenario in the optimal vendor penalty schema problem.

	Rule-of-thumb to find optimal penalties based on identified scenarios
Scenario 1	Set both penalties to original (no better penalties exist)
Scenario 2	Set Penalty Fee 1 to original value, Set Penalty Fee 3 to the upper bound, Decrease the Penalty Fee 2 until historical penalty is minimized to offset the incremental gains of Penalty Fee 3 (vpf2 = OVPF2 - (VPF3Max - OVPF3) (TVR-ORT2) / (ORT2 - ORT1).
Scenario 3	If feasible solution can be found by, Set Penalty Fee 1 to original, Set Penalty Fee 2 to the lower bound, Increase the Penalty Fee 3 until historical penalty is minimized to offset the incremental losses of Penalty Fee 2 (vpf3 = OVPF3 + (OVPF2 - VPF2Min) (ORT2-ORT1) / (TVR - ORT2). If not, Set Penalty Fee 1 to the upper bound, Set Penalty Fee 3 to the upper bound, Decrease the Penalty Fee 2 until historical penalty is minimized to offset the incremental gains of Penalty Fee 1 and 3 (vpf2 = OVPF2 - ((VPF3Max - OVPF3) (TVR-ORT2) + (VPF1Max - OVPF1) (ORT1)) / (TVR - ORT2).
Scenario 4	Set the Penalty Fee 2 to the upper bound, Set the Penalty Fee 3 to the upper bound, Decrease the Penalty Fee 1 until historical penalty is minimized to offset the incremental gains of Penalty Fee 2 and 3 (vpf1 = OVPF1 - ((VPF3Max - OVPF3) (TVR-ORT2) + (VPF2Max - OVPF2) (ORT2 - ORT1)) / (ORT1).

Scenario 5	If Expected Return > Historical Return, then
	Set Penalty Fee 3 to the upper bound,
	Set Penalty Fee 1 to the upper bound,
	Decrease the Penalty Fee 2 until historical penalty is minimized to offset the incremental gains of Penalty Fee 1 and 3
	(vpf2 = OVPF2 - ((VPF3Max - OVPF3) (TVR-ORT2) + (VPF1Max - OVPF1) (ORT1)) / (ORT2 - ORT1).
	Or,
	Set Penalty Fee 3 to the upper bound,
	Set Penalty Fee 2 to the lower bound,
	Increase the Penalty Fee 1 until historical penalty is minimized to offset the incremental gains of Penalty Fee 3 and losses of Penalty Fee 2
	(vpf1 = OVPF1 + ((VPF3Max - OVPF3) (TVR-ORT2) + (OVPF2 - VPF2Min) (ORT2 - ORT1)) / (ORT1).
	If Expected Return < Historical Return, then
	Set Penalty Fee 3 to the upper bound,
	Set Penalty Fee 1 to the lower bound,
	Increase the Penalty Fee 2 until historical penalty is minimized to offset the incremental gains of Penalty Fee 3 and losses of Penalty Fee 1
	(vpf2 = OVPF2 + ((VPF3Max - OVPF3) (TVR-ORT2) + (OVPF1 - VPF1Min) (ORT2 - ORT1)) / (ORT1).

Scenario 6	There exists no rule-of-thumb for this scenario, an optimal solution can only be found by a solver.
Scenario 7	Set Penalty Fee 3 to the upper bound, Set Penalty Fee 2 to the upper bound, Decrease the Penalty Fee 1 until historical penalty is minimized to offset the incremental gains of Penalty Fee 2 and 3 (vpf1 = OVPF1 - ((VPF3Max - OVPF3) (TVR-ORT2) + (VPF2Max - OVPF2) (ORT2 - ORT1)) / (ORT1).
Scenario 8	If feasible solution can be found by, Set Penalty Fee 3 to original value, Set Penalty Fee 1 to the lower bound, Increase the Penalty 2 until historical penalty is minimized to offset the incremental gains of Penalty Fee 1 (vpf2 = OVPF2 + (OVPF1 - VPF1Min) (ORT1) / (ORT2 - ORT1). If not, Set Penalty Fee 3 to original value, Set Penalty Fee 2 to the upper bound, Increase the Penalty 1 until historical penalty is minimized to offset the incremental gains of Penalty Fee 1 (vpf1 = OVPF1 + (VPF2Max - OVPF2) (ORT2 - ORT1) / (ORT1).

Scenario 9	If Expected Return > Historical Return, then
	Set all penalties to original value (since there are no possible cost savings in this special case).
	If Expected Return < Historical Return, then
	Set Penalty Fee 3 to the upper bound,
	Set Penalty Fee 1 to the lower bound,
	Increase Penalty Fee 2 accordingly until historical penalty is minimized to offset the incremental gains or losses of Penalty Fee 1 and 3
	(vpf2 = OVPF2 + ((VPF3Max - OVPF3) (TVR-ORT2) + (OVPF1 - VPF1Min) (ORT1)) / (ORT2 - ORT1).
	if not,
	Set Penalty Fee 3 to the upper bound,
	Set Penalty Fee 2 to the lower bound,
	Increase Penalty Fee 1 accordingly until historical penalty is minimized to offset the incremental gains or losses of Penalty Fee 2 and 3
	(vpf1 = OVPF1 + ((VPF3Max - OVPF3) (TVR-ORT2) + (OVPF2 - VPF2Min) (ORT2 - ORT1)) / (ORT1).
	if not,
	Set Penalty Fee 3 to the upper bound,
	Set Penalty Fee 1 to the upper bound,
	Decrease Penalty Fee 2 accordingly until historical penalty is minimized to offset the incremental gains of Penalty Fee 1 and 3
	(vpf2= OVPF2 - ((VPF3Max - OVPF3) (TVR-ORT2) + (VPF1Max - OVPF1) (ORT1)) / (ORT2 - ORT1).

Table 5.5: Rule-of-Thumb Mechanism for the Vendor Penalty Schema Problem

When we take a closer look at the results and savings of the individual vendors for the optimal return threshold schema problem, we observe a more complex solution structure than the optimal penalty fee problem. The inner workings of the optimal solution gave us specific scenarios (*Table 5.6*) and we have identified the specific conditions (and *Table 5.7*), under these scenarios, when a better return return threshold schema can be found that would bring savings in terms of penalties paid to the retailer.

For the 1000 vendors that were generated on each problem, the savings for each vendor ranged from 0% up to 35% on some certain cases. In order to identify the discrepancy of the cases, 25 distinct scenarios are identified in the solution structure as can be observed in *Table 5.6*. The number of scenario increased significantly compared to the first model due to the fact that now we have moving return thresholds instead of stationary ones, with the consideration of historical returns and expected returns, and this fact made the solution structure more complex than the optimal penalty fee problem. Even though, the optimal return threshold solution structure was more complex than the optimal penalty fee solution structure, the potential savings of solving the threshold problem was not great. Almost 3/4 of the vendors in each test problem, model was unable to find better return thresholds to save future penalty costs. When we deep dived into the reasons why the model was unable find better solutions for most of the vendors was due to the fact that only specific arrangement of the historical return, expected return, original return thresholds, and minimum and maximum bounds for the vendor return thresholds allowed to obtain better return thresholds schema for penalty savings. For instance, when expected returns are less than the minimum bound for return threshold 1 or greater than maximum bound for return threshold 2 (Scenario {1,2,3,4,5}x1 and Scenario {1,2,3,4,5}x5 in *Table 5.6*) , there are no better solutions then the original threshold schema because any kind of rearrangement of the thresholds would not affect the penalty savings for the retailer. Therefore, for any vendor where specific conditions hold (*Table 5.7*), penalty savings would not occur. There are actually very small number of scenarios/arrangements where the model can find penalty savings. We have deep dived into the scenarios in *Table 5.6* and identified under which circumstances (*Table 5.7*) the model would be able to find better solutions for an optimal return threshold schema which would result in penalty savings for the retailer.

POTENTIAL SCENARIOS					
CASES	0 <= EVR <= ORT1 Min 1	ORT1 Min 1 < EVR <= ORT2 Max 1	ORT1 Max 1 < EVR <= ORT2 Min 2	ORT1 Min 2 < EVR <= ORT2 Max 2	ORT1 Max 2 < EVR <= TVR
0 <= HVR <= ORT1 Min 1	Scenario 1 X 1	Scenario 1 X 2	Scenario 1 X 3	Scenario 1 X 4	Scenario 1 X 5
ORT1 Min 1 < HVR <= ORT2 Max 1	Scenario 2 X 1	Scenario 2 X 2	Scenario 2 X 3	Scenario 2 X 4	Scenario 2 X 5
ORT1 Max 1 < HVR <= ORT2 Min 2	Scenario 3 X 1	Scenario 3 X 2	Scenario 3 X 3	Scenario 3 X 4	Scenario 3 X 5
ORT1 Min 2 < HVR <= ORT2 Max 2	Scenario 4 X 1	Scenario 4 X 2	Scenario 4 X 3	Scenario 4 X 4	Scenario 4 X 5
ORT1 Max 2 < HVR <= TVR	Scenario 5 X 1	Scenario 5 X 2	Scenario 5 X 3	Scenario 5 X 4	Scenario 5 X 5

Table 5.6: Distinct Scenarios that Exist in Vendor Return Thresholds Problem

BETTER RETURN THRESHOLDS AVAILABLE WHEN					
CASES	0 <= EVR <= ORT1 Min 1	ORT1 Min 1 < EVR <= ORT2 Max 1	ORT1 Max 1 < EVR <= ORT2 Min 2	ORT1 Min 2 < EVR <= ORT2 Max 2	ORT1 Max 2 < EVR <= TVR
0 <= HVR <= ORT1 Min 1	N/A	EVR > ORT1	available all the time	EVR < ORT2	N/A
ORT1 Min 1 < HVR <= ORT2 Max 1	N/A	only EVR > ORT1 > HVR	HVR < ORT1	only ORT1 > HVR and ORT2 > EVR	N/A
ORT1 Max 1 < HVR <= ORT2 Min 2	N/A	N/A	N/A	N/A	N/A
ORT1 Min 2 < HVR <= ORT2 Max 2	N/A	only HVR > ORT2 and EVR > ORT1	HVR > ORT2	only HVR > ORT2 > EVR	N/A
ORT1 Max 2 < HVR <= TVR	N/A	EVR > ORT1	available all the time	EVR < ORT2	N/A

BETTER RETURN THRESHOLDS NOT AVAILABLE WHEN					
CASES	0 <= EVR <= ORT1 Min 1	ORT1 Min 1 < EVR <= ORT2 Max 1	ORT1 Max 1 < EVR <= ORT2 Min 2	ORT1 Min 2 < EVR <= ORT2 Max 2	ORT1 Max 2 < EVR <= TVR
0 <= HVR <= ORT1 Min 1	N/A	EVR < ORT1	available all the time	EVR > ORT2	N/A
ORT1 Min 1 < HVR <= ORT2 Max 1	N/A	HVR > ORT1 > EVR HVR > EVR > ORT1 ORT1 > HVR > EVR ORT1 > EVR > HVR EVR > HVR > ORT1	HVR > ORT1	HVR > ORT1 and EVR > ORT2 HVR > ORT1 and ORT2 > EVR ORT1 > HVR and EVR > ORT2	N/A
ORT1 Max 1 < HVR <= ORT2 Min 2	N/A	N/A	N/A	N/A	N/A
ORT1 Min 2 < HVR <= ORT2 Max 2	N/A	HVR > ORT1 and ORT2 > EVR ORT2 > LVR and ORT1 > EVR ORT2 > LVR and EVR > ORT1	HVR < ORT2	EVR > ORT2 > HVR EVR > HVR > ORT2 ORT2 > HVR > EVR ORT2 > EVR > HVR HVR > EVR > ORT2	N/A
ORT1 Max 2 < HVR <= TVR	N/A	EVR < ORT1	available all the time	EVR > ORT2	N/A

Table 5.7: Conditions Necessary for Optimal Solutions under Different Scenarios

As can be observed from *Table 5.7*, only under a handful of scenarios and circumstances, the model is able to identify better return thresholds. For any given problem parameters, if the conditions do not hold, the parameters can be kept as is because there are no better thresholds to obtain. However, if the conditions hold then the optimal thresholds can be found by sliding the mid-piece of the piece-wise linear function upwards and downwards. This phenomenon is a very intuitive way to observe the solution structure because keeping the penalty fees as they are, the slopes of the each piece-wise function can not change. Therefore, to obtain a better solution, only the mid-piece of the function can move upwards or downwards. The upward movement of the mid-piece results in decreasing return threshold 1 and increasing return threshold 2, and downward movement of the mid-piece results in increasing return threshold 1 and decreasing return threshold 2. Using the movement of the mid-piece of the piece-wise linear function, it is possible to calculate the gains and losses in penalties for both historical and expected returns and identify an approximate solution.

Table 5.7 provides insights to potential cases under which circumstances the decision maker choose to negotiate parameters. If conditions on *Table 5.7* do not hold, there is no point to try to identify, calculate and negotiate the return thresholds but only to concentrate on optimal penalty fees. If these conditions hold, then the decision maker has a chance to calculate an approximate solutions via calculating the gain and losses in penalties for both historical and expected returns.

Only identifying optimal return thresholds does not provide a significant savings to the retailer compared to identifying optimal penalty fee structure. As can be observed from *Table 5.3*. If only optimal return thresholds are identified and re-negotiated to these levels with every vendor, the penalty savings for all vendors would be 1.5-2% (*Table 5.3*). If only optimal penalty fee schema are identified and re-negotiated with every vendor, the penalty savings for all vendors would be 2-3% (*Table 5.2*). Since penalty schema has a larger range to affect the penalty savings compared to identifying better return thresholds, readjusting the penalty percentages would shrink the penalties much more effectively than rearranging the return thresholds does. Using a combination of both charts in *Table 5.5* and *Table 5.7*, the decision maker can easily identify which path to pursue and calculate an exact or an approximate solution. However, if the retailer is able to re-negotiate all the potential parameters in the contract with all of its vendors, which is a combined solution that identifies optimal penalty fee and return threshold schema simultaneously, the retailer can save up 6-7 % as shown in *Table 5.1* in terms of penalties that would be incurred.

Chapter 6

Conclusion and Future Research Directions

In this chapter, we provide a summary of the major findings of the thesis and outline potential future research directions.

6.1 Thesis Summary

In this thesis, a RSC system is considered in a retailing environment. Based on literature reviewed in *Chapter 2*, we have observed a significant gap in operational, tactical and strategic models in RRSC. RSC literature is over supplied with conceptual frameworks, designs and operational models that consider almost all kinds of products, material recovery, and waste. Hopefully, this thesis and the following studies will start a new area of research in RSC literature where the focus is not only on product recovery and waste management of the RL activities, but also on designing and modelling other RL activities that may have RSC systems such as RSC of independent retailers and/or online retailers.

In the first problem in *Chapter 3*, we have discussed and modelled budgetary limitation, where the cost of doing business was restricted under budgetary limitations due to financial, accounting and taxation reasons. Similar budgetary limitations can be found in any industry where a certain amount of budget is set to be spent in order to action a process, and if this budget is not utilized, there would be certain consequences for over- or under- spending. The heuristic approach we have developed can be adjusted and reconfigured to solve similar problems in any industry that has similar budgetary limitations.

The sheer complexity of a realistic RSC of a retailer that is discussed and modelled in *Chapter 4*, and the number of network nodes, paths, rules, restrictions and elements, emphasizes the importance and challenges of modelling these networks and the usage of complex methodologies to solve these kinds of problems faster than available solution techniques. The multi-stage heuristic that is developed to solve this problem can help researchers develop similar heuristics for similar problems in nature, not just in RSC but also for forward supply chains, network design problems, production planning, inventory optimization, distribution, selection, allocation, location and location-allocation problems where the nature of the problems show a large number of choices, load allocation and node activation decisions.

Independent retailers and their suppliers, such as vendors or manufacturers, are always both collaborating and competing at the same time for different reasons. When it comes to selling products to consumers, both parties collaborate in a way to maximize their exposure to sales via providing internal and external discounts, promotions, and easy pay structures that would allow both parties to satisfy customers' demand. However, due to the overloaded incentives by the suppliers, retailers might end up with excessive inventory where the demand is not met by the end consumer for best selling products, or there might be no demand for some portion of the ordered products, because of customer desires. Therefore, there exists a buy-back and returns contract, discussed in *Chapter 5*, between the supplier and the retailer to share the cost of doing business. Our goal in the developed solution approach is to propose a method that is efficient but at the same time easy to understand and implement by manager.

6.2 Future Work and Open Areas for Research

We have found that there is a significant gap in the literature of RRSC, especially on the operational and tactical decision making problems that a retail has to deal with during its RL activities. One of the main gaps in the literature was the lack of the 'selection' aspect of the decision making process in RSC where the responsible parties had to identify which products (whether they are unsold / end-of-life / end-of-use /used / damaged / broken / faulty) should participate in the RL activities. This issue is also valid for the regular 'product recovery / re-manufacturing' RSC literature of manufacturers or 'recycling / waste management' RSC literature of green supply chains. Almost all of the existing RSC models consider all the collected products and/or material to be returned, except for few models, e.g., Eskandarpour et al. (2013). However, similar product selection issues are a valid concern for manufacturers due to product/material quality of the collected items and brings another complexity to the problem. Selection decisions might not be of concern for RSC in waste management or recycling of products / materials such as used tires, construction materials, glass, plastic or paper, however in many circumstances the collected and then 'potentially' returnable products should be evaluated and selected based on some sort of quality, content, environmental concerns or financial viability criteria to be reverse flowed in the RSC. We have observed that the RSC, as well as the CLSC, literature has significant gaps in this regard.

We have considered three major problems that were related to RRSC. For the problems that are discussed in *Chapter 3* and *Chapter 4*, the demand side of the returned products by the stores / warehouses is considered to be deterministic. This assumption was valid in our industrial partner's context where we formulate the warehouse demand component as the expectation of the forward supply chain forecasts. In some other environments, it is quite likely that the demand of the these products may be random. Therefore, a natural extension of our work is to consider cases where the demand for the returned items is stochastic.

For the two problems that are discussed in *Chapter 5*, assumed that future-expected return amounts are deterministic in order to identify optimal penalty and/or threshold parameters for contract re-negotiation with vendors. However, slight changes in these 'expected' return amounts can change the optimal penalty or threshold strategy significantly, especially if the expected figures changes the 'range' that are shown in the *Figure 5.3* and *Figure 5.4* (a different range will result in a different scenario). Since retailers can control the nature of their sales, such as through promotions, they can arrange, with the help of sales of regular forecasts, their potential future return in a very close range. However, there is still an uncertainty in the nature of the problem and considering the stochastic nature of this future-expected returns is worth considering in the future.

Another avenue for future research is to formally study the complexity of the models considered in this thesis. As well as develop more efficient computational procedures. For example, the decomposed LP and MILP models that are defined and solved in *Chapter 5* can also be further developed to a more complex heuristic that can be solved sequentially to find close-to-optimal solutions.

Another potential area of research is the consideration of multi-periods in RRSC literature. For the problems we have studied in *Chapter 3* and *Chapter 4*, we know from our industrial partner's that they conduct their RL activities twice in a year in order to optimize their inventory, either under a certain budget or purely to optimize inventory costs. Therefore a multi-period version of these models can be further developed in order to optimally use profit-loss budgets or optimize inventory holdings during several time-periods within a given time horizon. Additionally, the problem we have modelled in *Chapter 5* is also multi-period in nature since when a contract is re-negotiated, it will be on effect for several years, therefore modelling this framework in a multi-period is an area that is worthy of further investigations.

Vendors also provide counter offers to the retailer. These counter offers can be incorporated in the model structure using a multi-period setting and should be investigated further to develop a solution that is tailored to each vendor.

In time, more sophisticated vendors also learn from the new implemented contract parameters that they are not making as much money as previously from the penalties they charge to the retailer. As a result, these vendors might have concerns after several return cycles that they are not satisfied with the new contract parameters. Further revenue-sharing models can be investigated to address the concern of these vendors. Another possible extension is to incorporate the power relationship between a single retailer and a vendor using game theory.

In this thesis we study the RRSC from an independent retailer's perspective. Extending it to an 'online' retailer's perspective is a potential avenue for future research. As we have shown in the related RRSC literature in *Chapter 2*, it is clear that there is a significant gap in modelling the RRSC in this area of research. To the best of our knowledge, our research is the only research where an independent retailer's RRSC network is modelled for budget, inventory and contract parameter optimization. Online retailers bring the additional complexity that returns are often higher and they have different channels for returns.

Finally, when conducting our research with our industrial partner we noticed that the amount of information exchange between retailer-owned stores and franchisees, as well as the flexibility that is given to franchisees for procuring for alternative sources, create challenging and interesting areas for future research. The use of game theory and mechanism design may lead to insightful models.

Appendix A

A1 Heuristic Algorithm Detailed Steps for Chapter 3

1. Calculate the total amount of ineffective store-product returns - in terms of \$ value - that need to be pulled from all stores using constraint 2.6.
   ($TotalInventoryReturn = TSRA$)

2. Create a list of ineffective store-products, list 1, where there is any amount of demand for a product at the warehouse.

3. Rank list 1 by 'Cost of Returning to Warehouse Ratio' ($(SP_p - LC_p) \,/\, SP_p$) in a descending order.

4. Select the set of store-products from the ranked list 1, top-to-bottom, until either $TotalInventoryReturn$ is satisfied or warehouse capacity is full. If $TotalInventoryReturn$ is full, then STOP, else go to next step.

5. Calculate the inventory amount that is returned to warehouse
   ($WarehouseReturn = \sum_{p=1}^{P} wh_p \, SP_p$).

6. Calculate the remaining inventory return amount from the Total Inventory Return amount and ($RemainingInventoryReturn = TotalInventoryReturn$ - $WarehouseReturn$)

7. Remove the chosen selected store-products (that is identified in list 1) from the overall store-product list, and create a new list, list 2.

8. Rank list 2 by 'Total Cost of Returning (Profit Margin loss from COGs + Return Penalty) to Vendor Ratio' ($((SP_p - COG_p) + COG_p \, VPF_v) \,/\, SP_p$) in an ascending order.

9. Select the set of store-products from the ranked list 2 until $RemainingInventoryReturn$ is satisfied.

10. Observe whether the model is under-budget or not. If the model solution is under-budget $((\sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}\,(SP_p - LC_p)\,Q_{s,p}) + (\sum_{p=1}^{P} vr_p\,(LC_p - COG_p)) + (\sum_{v=1}^{V} VPF_v\, vf_v) \leq PLB)$ then it means that we can minimize all the costs while using all the available budget. We can directly proceed to Step 9 in order to use all the available budget, if applicable. If the model solution uses more than the given budget $((\sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}\,(SP_p - LC_p)\,Q_{s,p}) + (\sum_{p=1}^{IP} vr_p\,(LC_p - COG_p)) + (\sum_{v=1}^{V} VPF_v\, vf_v) > PLB)$ in the optimal solution, then we are over-budget and therefore the costs can NOT be minimized with the given budget. In this case, skip the next steps and directly proceed to the Step 16.

11. When the model is under-budget then it means that the minimized cost structure uses/spends less than the given budget and products that need to be sent to the vendors should have a higher 'Total Cost of Returning (Profit Margin loss from COGs + Return Penalty) to Vendor Ratio' so that the model can use more of the available profit-loss budget by sending more profitable products, instead of the less profitable products, back to vendors.

12. Calculate the total amount of inventory (in terms of Store Price) that needs to be pulled from store inventories and identify how much of the returns are vendor returns vs warehouse returns.
($WarehouseReturn = \sum_{p=1}^{P} wh_p\, SP_p$, $VendorReturn = \sum_{p=1}^{P} vr_p\, SP_p$, in return $VendorReturn = RemainingTotalInventoryReturn$ in this case.)

13. Calculate the 'Profit-loss Budget' used by the vendor returns ($BudgetUsedByVendorReturn = \sum_{p=1}^{P} vr_p\,(SP_p - COG_p) + \sum_{p=1}^{P}\sum_{v=1}^{V} vr_p\, COG_p\, VPF_v$), warehouse returns ($BudgetUsedByWarehouseReturn = \sum_{p=1}^{P} wh_p\,(SP_p - LC_p)$) and remaining budget ($RemainingBudgetNewVendorReturn = PLB - \sum_{p=1}^{P} wh_p\,(SP_p - LC_p)$) that needs to be used by new vendor return in order to use all the available budget.

14. Calculate the 'Total Cost of Returning (Profit Margin loss from COGs + Return Penalty) to Vendor Ratio' for the vendor returns and then calculate the 'Expected Total Cost of Returning to Vendor Ratio' ($DesiredProfitLossRatio = RemainingBudgetNewVendorReturn\, /\, (\sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}\, SP_p\, Q_{s,p} - \sum_{p=1}^{P} wh_p\, SP_p)$) that should satisfy the remaining budget after we remove the profit-loss budget that is used by warehouse returns.

15. After determining the 'Expected Total Cost of Returning to Vendor Ratio' for

the remaining budget, rank remaining ineffective store-products that could be returned to vendors by 'Cost of Returning to Vendor Ratio' ($((SP_p - COG_p) + COG_p VPF_v) \,/\, SP_p$) in a descending order and identify a lower and upper bound (by iteration) of 'Cost of Returning to Vendor Ratio' which will include a set of store-products that will be returned to vendors which would satisfy both the remaining returns ($VendorReturn = RemainingTotalInventoryReturn$) and profit-loss budget ($RemainingBudgetNewVendorReturn$), if applicable. If not, choose the set store-products from the top of this list which could only satisfy the remaining return amount. In the case of picking the highest 'Cost of Returning to Vendor Ratio' store-products not satisfying the profit-loss budget happens, then it means that we would never be able to use the given budget even we choose the best possible store-products to return to the warehouse and vendor. Even though this solution is not a feasible solution to the original problem, the solution is letting us know that the result is best result that is capable of using the maximum amount of profit-loss budget that is available within the given product parameters and it is still an acceptable solution to the general problem since we are using the given budget as much as possible.

16. When the model is over-budget then it means that the minimized cost structure uses all of the available profit-loss budget and products that need to be sent to the warehouse should have a lower 'profit margin (from the landed price)' so that the model can use less of the available profit-loss budget by sending less profitable products to warehouse. Also as a result, this would most likely to end up in a lower value of warehouse return and higher value of vendor returns, in terms of Store Price/Refund, since we are returning the same amount of value, in terms of COGs, with lower profit margins (from the landed price) to the warehouse because of the warehouse's capacity issue, in terms of COGs.

    Please note that depending on the profit margins available in the store-product data set and the amount of budget that the model is over, there might be products with lower profit margins (from the landed price) and higher vendor unrecoverable costs that might still end up with the same amount of warehouse and vendor returns, in terms of Store Price/Refund. However, if the gap between profit-loss budget available and profit-loss budget used is greater than a certain threshold, then the model inevitable has to identify products that have lower profit margin with higher COGs and therefore will not able to send products to the warehouse as valuable, in terms of Store Price/Refund, as it previously could. This will most likely end up in a lower value of warehouse return and higher value of vendor returns.

17. Calculate the total amount of inventory ($TotalInventoryReturn = \sum_{s=1}^{S}\sum_{p=1}^{P} rt_{s,p}\, SP_p Q_{s,p}$) that needs to be pulled from store inventories and identify how much of them is a vendor return vs. a warehouse return. ($WarehouseReturn = \sum_{p=1}^{P} wh_p\, SP_p$, $VendorReturn = \sum_{p=1}^{P} vr_p\, SP_p)$)

18. Calculate the 'profit-loss budget' used by the vendor returns ($BudgetUsedByVendorReturn = \sum_{p=1}^{P} vr_p\,(SP_p - COG_p) + \sum_{p=1}^{P}\sum_{v=1}^{V}\ vr_p\, COG_p\, VPF_v$) and warehouse return ($BudgetUsedByWarehouseReturn = \sum_{p=1}^{P} wh_p\,(SP_p - LC_p)$)

19. Remove existing vendor returned store-products from the list since they use the minimal profit-loss budget as possible.

20. Rank remaining store-products by 'Cost of Returning to Vendor Ratio' ($((SP_p - COG_p) + COG_p\, VPF_v/SP_p)$ in an ascending order, this is list 3.

21. Create a separate list, list 4, of store-products that are potentially warehouse returnable and rank these store-products by 'Cost of Returning to Warehouse Ratio' $(SP_p - LC_p)/SP_p$ in a descending order. Choose a set of store-products from the TOP of the list, which were potentially going back to the warehouse, and now some of them will stay in store inventories.

22. Calculate the Old Warehouse Return Amount ($\sum_{p=1}^{P} wh_p^{(original)} SP_p$) and the Old Warehouse Return COGs Amount ($\sum_{p=1}^{P} wh_p COG_p$) of the chosen list, which were potentially going back to the warehouse.

23. Now, identify a set of store-products (by iteration) from the BOTTOM of the list 4, that corresponds to the same total COGs value of the above list (which, in turn, should have a lower 'Total Warehouse Return Amount' than the above 'Old Warehouse Return Amount').

24. Assign the new set of store-products to be returned back to the warehouse and remove the tag of 'Old Warehouse Returns' to 'stay-in-store'.

25. Calculate the Inventory Amount of the New Warehouse Returns ($\sum_{p=1}^{P} wh_p^{(new)} SP_p$) and 2^{nd} Vendor Return ($\sum_{p=1}^{P} wh_p^{(original)} SP_p - \sum_{p=1}^{P} wh_p^{(new)} SP_p$)

26. Choose a set of store-products from the top of the ranked list 3 where 2^{nd} Vendor Return Amount is satisfied.

27. Calculate Used Profit-loss budget used by the new warehouse ($\sum_{p=1}^{P} wh_p^{(new)}(SP_p - LC_p)$ and; existing and new vendor returns ($\sum_{p=1}^{P} vr_p^{(original)}(SP_p - COG_p)$ $+\sum_{p=1}^{P}\sum_{v=1}^{V} \; vr_p^{(original)}COG_pVPF_v$ $+\sum_{p=1}^{P} vr_p^{(new)}(SP_p - COG_p)$ $+\sum_{p=1}^{P}\sum_{v=1}^{V} \; vr_p^{(new)}COG_pVPF_v$)

28. If New Profit-loss Budget is less than given Profit-loss Budget, then STOP, if it is greater than the given Profit-loss Budget then go to Step 23.

A2 Heuristic Algorithm Detailed Steps for Chapter 4

1. Define all the possible scenarios that can exist in a network where all the RCs are either activated or deactivated. Based on the given scenario where some of the RCs are active and some of the RCs are inactive, define the RSC network paths among the activated RCs.

2. Calculate the total store removal amount from all company owned stores as a whole using the total effective and ineffective inventory amount in each company owned store and the national inventory healthy ratio.

3. Calculate the individual store removal amount for all franchise stores using individual inventory levels and their relative ratios compared to total franchise inventory amount.

4. For every scenario, calculate the total transportation cost of a Store-Product that it would take all the items of that Store-Product from its current location to all the warehouses, their original vendor and liquidation in the RSC network using the volume of the products, unit transportation costs from stores to their RCs, unit transportation cost among RCs, unit transportation cost from RCs to Warehouses, unit transportation cost from RCs to Vendors and unit transportation cost from RCs to Liquidation.

5. For every Ineffective Store-Product, calculate the total Receiving&Handling cost of a Store-Product that it would take all the items of that Store-Product from its current location to all the warehouses, their original vendor and liquidation in the RSC network using the weight and volume of the products and Receiving&Handling Fee at RCs.

6. After calculating all the (total) transportation, receiving&handling costs for every Store-Product in their potential warehouse demand locations, calculate the 'Net Gain Returning' and therefore 'Actual Profit Gain Ratio' for every potential

Warehouse Return using all the calculated costs and Profit Margins (from Landed), Store Prices and Store Refund Rates under every scenario.

7. Identify the First-Best, Second-Best, etc. to Last-Best Warehouse for the Store-Products under every scenario.

8. Since we observe that the highest profit margin store-products need to be routed to the Warehouses, the Store-Product needs to be ranked using a calculated profit margin data point. Therefore, rank the Store-Products using 'Actual Profit Gain Ratio' using the First-Best Warehouse data field under every scenario. Ranking Store-Products based on this field allows us to send most profitable store-products to potential sources of demand in our RSC network and fill the warehouses with profitable items.

9. Route the items of the store-product to the First-Best warehouse where there is demand and capacity at the same time under every scenario. If the store-product can not be routed to the First-Best Warehouse because of demand or capacity issues at that warehouse, try to route it to the Second-Best Warehouse. If the store-product can not be routed to the Second-Best Warehouse because of demand or capacity issues at that warehouse, try the next warehouse, and so on. If the store-product can be routed to any warehouse, skip to the next store-product and search for potential demand and capacity for that to be potentially routed.

10. Route the store-products to the Warehouses until the total removal amounts from stores are satisfied both from the overall Company Store's perspective and individual Franchise Store's perspective under every scenario. If enough store-products are removed from stores, then STOP. If not, then go to next step.

11. Remove all the store-products that were returned to warehouses from the original dataset to get the remaining store-products that have not been returned to anywhere. This is the new list that you need to use in order to satisfy the store removal amounts.

12. Since Vendor Return option is the second-least costly and therefore next best option, we will use the new list to identify what the best potential store-products to return back to their vendor. However, we need to calculate another important cost structure to the 'Actual Profit Loss Ratio' for Vendor Return, which is called Vendor Penalty. This cost should also be included in the calculation of 'Actual Profit Loss Ratio' for Vendor Return with the use of Unrecoverable, Profit Margin (from Landed), Transportation and Receiving&Handling Costs. Since we have

a step-wise penalty function for Vendor Returns, we first need to identify which store-product needs to get which penalty ratio. It is logical to assume that the products with lower penalties need to be assigned to products with lower margins (actual profit loss ratio), since we only need to extract a portion of the products to a vendor and also vendors also will have limit of products they will accept to receive with given penalty structures.

13. Calculate the 'Total Cost of Returning for Vendor Returns without Penalty' and therefore 'Actual Profit Loss Ratio without Penalty' for Vendor Return using Unrecoverable, Profit Margin (from Landed), Transportation and Receiving&Handling Costs.

14. For every vendor, rank store-products based on 'Actual Profit Loss Ratio without Penalty' in an ascending order and assign the relevant penalties to those ranked products in order to calculate the correct Vendor Penalty Cost that should be identified and assigned for each store-product. Calculate the Vendor Penalty Cost for each store-product if the decision is to return them back to the vendor.

15. Since Vendor Penalty Cost is calculated for every store-product, we can now calculate the 'Total Cost of Returning for Vendor Returns with Penalty' and therefore 'Actual Profit Loss Ratio with Penalty'. Calculate the 'Actual Profit Loss Ratio with Penalty' in order to identify the lowest profit loss store-products.

16. Rank store-products based on 'Actual Profit Loss Ratio with Penalty' in an ascending order.

17. Since we observe that the lowest profit margin store-products need to be routed to Vendors, route the items of the store-product to its Vendor starting from the top of list and work our way down in the list until the total removal amounts from stores are satisfied both from the overall Company Store's perspective and individual Franchise Store's perspective. If enough store-products are removed from stores, then STOP. If not, then go to next step.

18. Remove all the store-products that were returned to vendors from the second dataset to get the remaining store-products that have not been returned to anywhere. This is the 3rd new list that you need to use in order to satisfy the store removal amounts.

19. Since Vendor Deposit Return or Liquidation option, interchangeably, is the third-least costly and therefore next best option, we will use the 3^{rd} new list to identify

what the best potential store-products to either send them for liquidation or return back to Vendor with a Vendor Deposit Return.

20. Calculate the 'Total Cost of Returning for Vendor Deposit Returns' and therefore 'Actual Profit Loss Ratio' for Vendor Deposit Return and the 'Total Cost of Returning for Liquidation' and therefore 'Actual Profit Loss Ratio' for Liquidation using Deposit Value of the product, Liquidation Rebate Rate of the product, COG, Profit Margin (from Landed), Transportation and Receiving&Handling Costs.

21. Generate a data field, called 'Lowest Actual Profit Loss Ratio After Vendor Return', that would take the lowest value between the 'Actual Profit Loss Ratio' for Vendor Deposit Return and 'Actual Profit Loss Ratio' for Liquidation. This value will help us to decide whether the store-product should be sent back to Vendor for deposit value extraction or liquidate the items of the product to a third party buyer. If 'Actual Profit Loss Ratio' for Vendor Deposit Return is lower, then store-product will be sent back to its Vendor for deposit value extraction, If 'Actual Profit Loss Ratio' for Liquidation value is lower, then the store-product will be sent for liquidation to a third party buyer from store's RC.

22. Rank store-products based on this newly calculated field, 'Lowest Actual Profit Loss Ratio After Vendor Return', in an ascending order.

23. Since we observe that the lowest profit margin store-products need to be routed to either Vendor for Deposit Value Extraction or Liquidation, route the items of the store-product to the one of those locations starting from the top of the list and work our way down in the list until the total removal amounts from stores are satisfied both from the overall Company Store's perspective and individual Franchise Store's perspective. If enough store-products are removed from stores, then STOP. If not, then go to the next step.

24. Remove all the store-products that were returned to either vendors for deposit return or sent for liquidation from the third dataset to get the remaining store-products that have not been returned to anywhere. This is the 4^{th} new list that you need to use in order to satisfy the store removal amounts.

25. Since destroying the store-product at source is the most costly option, we will use the 4^{th} new list to identify what other store-product can also destroyed in order to reach the total store removal amount goals. Destroy store-product from stores until the total removal amounts from stores are satisfied both from the overall

Company Store;s perspective and individual Franchise Store's perspective. When enough store-products are destroyed at source, STOP.

26. Once overall store removal amount is satisfied for company owned stores and individual removal amount is satisfied for every franchise store under every scenario, calculate the related costs in each scenario.

27. Rank the scenarios by the total cost of the process in an ascending order and pick the top scenario that is ranked the highest.

A3 Detailed Numerical Results for Chapter 5

Vendor	Original Vendor Penalty Fee 1	Original Vendor Return Threshold 1	Original Vendor Penalty Fee 2	Original Vendor Return Threshold 2	Original Vendor Penalty Fee 3	Optimal Vendor Penalty Fee 1	Optimal Vendor Return Threshold 1	Optimal Vendor Penalty Fee 2	Optimal Vendor Return Threshold 2	Optimal Vendor Penalty Fee 3	Total Vendor Funds Total
v1	20%	64843	40%	259372	70%	18%	96,675	31%	212,870	75%	389,058
v2	13%	106133	26%	424532	46%	19%	71,415	23%	461,552	52%	636,798
v3	23%	140406	46%	561624	81%	23%	97,477	40%	539,918	84%	842,436
v4	21%	92260	42%	369040	74%	15%	115,252	40%	326,605	76%	553,560
v5	19%	50458	38%	201832	67%	17%	73,390	44%	152,226	72%	302,748
v6	24%	51757	48%	207028	84%	22%	93,663	42%	170,867	89%	310,542
v7	20%	57090	40%	228360	70%	15%	113,540	48%	183,261	69%	342,540
v8	15%	107388	30%	429552	53%	15%	125,699	22%	387,282	58%	644,328
v9	14%	139680	28%	558720	49%	20%	95,764	23%	515,349	49%	838,080
v10	7%	81744	14%	326976	25%	8%	42,802	8%	282,854	28%	490,464
v11	22%	104838	44%	419352	77%	22%	159,490	34%	382,547	86%	629,028
v12	20%	148101	40%	592404	70%	20%	103,792	38%	636,777	76%	888,606
v13	14%	51997	28%	207988	49%	7%	37,387	29%	164,825	42%	311,982
v14	23%	82811	46%	331244	81%	24%	46,396	45%	359,440	84%	496,866
v15	15%	100808	30%	403232	53%	12%	96,588	31%	354,125	52%	604,848
v16	14%	141108	28%	564432	49%	8%	180,900	28%	523,426	52%	846,648
v17	10%	67528	20%	270112	35%	9%	38,058	18%	221,114	31%	405,168
v18	10%	129748	20%	518992	35%	10%	176,234	16%	471,807	38%	778,488
v19	11%	140300	22%	561200	39%	14%	97,652	19%	515,262	39%	841,800
v20	7%	127968	14%	511872	25%	9%	82,053	9%	543,144	34%	767,808
v21	23%	84551	46%	338204	81%	23%	129,669	45%	319,257	84%	507,306
v22	12%	87878	24%	351512	42%	7%	116,931	28%	310,041	48%	527,268
v23	22%	100713	44%	402852	77%	27%	121,873	43%	442,076	86%	604,278
v24	25%	55146	50%	220584	88%	25%	20,790	44%	226,856	94%	330,876
v25	25%	119199	50%	476796	88%	25%	85,078	43%	432,423	88%	715,194
v26	12%	51296	24%	205184	42%	6%	97,256	27%	160,223	43%	307,776
v27	18%	66274	36%	265096	63%	18%	122,903	34%	250,570	69%	397,644
v28	16%	57369	32%	229476	56%	17%	57,083	31%	184,189	50%	344,214
v29	7%	115930	14%	463720	25%	9%	75,081	11%	422,474	25%	695,580
v30	19%	93836	38%	375344	67%	21%	60,072	29%	371,338	77%	563,016
v31	25%	136227	50%	544908	88%	25%	164,157	43%	502,233	93%	817,362
v32	16%	116803	32%	467212	56%	16%	86,176	30%	418,643	52%	700,818
v33	14%	116508	28%	466032	49%	8%	74,320	27%	425,891	47%	699,048
v34	8%	138998	16%	555992	28%	11%	181,036	14%	617,176	35%	833,988
v35	9%	110072	18%	440288	32%	10%	68,025	15%	422,810	33%	660,432
v36	17%	57610	34%	230440	60%	13%	55,468	27%	181,725	59%	345,660
v37	20%	140472	40%	561888	70%	14%	137,148	42%	517,699	66%	842,832
v38	20%	90961	40%	363844	70%	15%	138,254	35%	314,550	79%	545,766
v39	24%	59208	48%	236832	84%	30%	19,627	40%	195,557	76%	355,248
v40	13%	86724	26%	346896	46%	8%	111,585	20%	296,992	52%	520,344
v41	23%	83656	46%	334624	81%	17%	34,061	44%	286,942	73%	501,936
v42	21%	70441	42%	281764	74%	15%	33,012	38%	240,364	70%	422,646
v43	11%	109311	22%	437244	39%	14%	71,795	16%	454,012	48%	655,866
v44	12%	143930	24%	575720	42%	10%	112,449	27%	534,796	36%	863,580
v45	9%	75847	18%	303388	32%	10%	26,187	10%	254,444	40%	455,082
v46	9%	76349	18%	305396	32%	13%	41,823	13%	260,666	32%	458,094
v47	15%	104428	30%	417712	53%	20%	55,362	25%	454,258	63%	626,568
v48	24%	102861	48%	411444	84%	22%	104,157	47%	369,054	80%	617,166
v49	14%	107687	28%	430748	49%	9%	127,074	37%	382,373	40%	646,122
v50	10%	72709	20%	290836	35%	13%	138,205	17%	249,914	35%	436,254

Vendor	Historical Vendor Return Amount	Expected Vendor Return Amount	Total Vendor Return (Old)	Total Vendor Return (Optimal)	Total Vendor Return Difference	Last Vendor Return (Old)	Last Vendor Return (Optimal)	Last Vendor Return Difference	Expected Vendor Return (Old)	Expected Vendor Return (Optimal)	Expected Vendor Return Difference
v1	330,545	181,306	$181,560	$185,563	4,003	140,601	141,678	1,077	59,554	43,637	15,917
v2	359,647	523,543	$193,162	$194,428	1,266	79,711	79,862	151	141,631	135,536	6,095
v3	73,204	768,260	$452,107	$453,511	1,404	16,837	16,837	-	392,396	391,204	1,192
v4	446,170	382,280	$271,244	$274,315	3,070	192,313	192,698	386	145,354	144,142	1,212
v5	155,446	105,775	$134,218	$155,540	21,322	49,482	49,482	(0)	30,607	26,726	3,882
v6	241,766	212,073	$173,904	$177,342	3,439	116,132	116,132	(0)	91,190	89,705	1,485
v7	232,852	20,044	$159,852	$160,400	548	83,070	84,715	1,644	4,009	3,007	1,002
v8	49,980	125,699	$225,515	$225,490	(25)	7,497	7,497	-	21,602	18,855	2,747
v9	729,105	760,859	$273,773	$273,796	23	220,375	220,398	23	235,935	235,957	(23)
v10	474,232	399,396	$ 80,109	$ 80,759	650	76,132	76,214	82	57,797	55,260	2,537
v11	95,619	428,238	$322,901	$322,901	(0)	21,036	21,036	-	168,293	150,222	18,071
v12	53,682	868,143	$414,683	$414,683	0	10,736	10,736	-	400,359	399,131	1,228
v13	94,550	59,759	$101,914	$101,380	(534)	19,194	19,194	(0)	9,453	9,105	348
v14	168,687	494,299	$266,651	$267,443	791	58,549	66,166	7,617	264,585	265,286	(701)
v15	323,054	170,288	$211,697	$221,803	10,106	81,795	81,795	(0)	35,965	34,438	1,528
v16	824,025	519,685	$276,572	$278,455	1,883	265,486	266,691	1,204	125,757	109,332	16,425
v17	191,750	38,058	$ 94,539	$ 93,432	(1,107)	31,597	31,090	(507)	3,806	3,425	381
v18	47,136	336,574	$181,647	$181,454	(193)	4,714	4,714	-	54,340	43,278	11,062
v19	315,868	316,425	$216,062	$220,367	4,305	54,058	55,132	1,074	54,181	55,238	(1,058)
v20	181,355	605,870	$125,409	$125,269	(140)	16,432	16,322	(110)	85,734	70,210	15,524
v21	56,832	358,669	$272,254	$273,100	846	13,071	13,071	-	152,601	148,245	4,357
v22	316,743	195,771	$147,635	$166,525	18,890	65,473	65,473	(0)	36,440	30,260	6,179
v23	276,667	448,134	$310,196	$310,087	(109)	99,577	99,467	(109)	189,965	175,803	14,162
v24	54,133	279,015	$193,011	$193,645	634	13,533	19,868	6,335	147,633	144,896	2,737
v25	204,514	385,957	$417,197	$419,466	2,270	72,457	72,627	170	163,179	150,647	12,531
v26	297,075	137,469	$ 86,177	$ 86,284	107	81,683	81,683	0	26,837	16,693	10,144
v27	38,176	293,742	$167,010	$167,010	0	6,872	6,872	-	101,552	95,318	6,234
v28	118,748	142,504	$128,507	$129,120	613	28,820	28,820	0	36,422	36,185	238
v29	629,008	690,421	$113,611	$113,247	(364)	97,301	96,604	(697)	112,347	111,957	390
v30	106,589	464,825	$249,604	$250,474	870	22,675	26,105	3,430	184,307	174,867	9,440
v31	88,116	164,157	$476,795	$479,482	2,687	22,029	22,029	-	48,022	41,039	6,983
v32	341,073	404,362	$261,639	$260,259	(1,379)	90,455	90,257	(198)	110,707	109,244	1,463
v33	240,954	400,068	$228,356	$229,254	898	51,156	50,937	(219)	95,708	93,898	1,810
v34	265,662	733,385	$155,678	$156,858	1,180	31,386	31,762	376	127,509	121,647	5,862
v35	185,480	637,147	$138,691	$138,436	(255)	23,480	24,421	941	131,356	130,751	604
v36	252,857	55,468	$137,112	$138,022	910	81,894	83,268	1,374	9,430	7,211	2,219
v37	515,358	408,217	$393,322	$393,620	298	178,049	178,049	(0)	135,192	133,050	2,143
v38	430,082	361,221	$254,691	$265,102	10,412	173,712	173,712	-	126,296	119,312	6,984
v39	204,598	299,438	$198,939	$197,625	(1,314)	83,997	83,131	(866)	152,058	155,210	(3,151)
v40	462,905	380,642	$157,838	$162,151	4,314	131,703	132,283	580	94,273	89,506	4,767
v41	312,820	32,788	$269,372	$274,004	4,631	124,656	135,949	11,293	7,541	5,574	1,967
v42	283,787	32,418	$207,097	$211,343	4,246	105,035	114,142	9,106	6,808	4,863	1,945
v43	182,948	461,543	$168,339	$168,096	(243)	28,224	27,836	(389)	93,525	74,821	18,704
v44	745,080	94,158	$241,802	$243,641	1,838	192,032	200,981	8,948	11,299	9,416	1,883
v45	331,491	285,887	$ 95,567	$105,700	10,132	56,636	56,263	(373)	44,633	38,022	6,612
v46	334,373	400,663	$ 96,200	$ 97,064	864	57,228	57,473	245	78,109	78,686	(577)
v47	268,518	485,976	$219,299	$219,352	53	64,891	64,361	(530)	145,488	130,779	14,709
v48	456,667	104,157	$345,613	$345,906	293	210,794	217,507	6,713	25,309	22,915	2,394
v49	227,827	170,112	$211,067	$211,397	330	48,715	48,715	(0)	32,555	27,361	5,194
v50	98,857	199,520	$101,793	$102,176	384	12,501	12,851	351	32,633	28,390	4,243
									5,042,280	**4,790,255**	**5.00%**

Table A1.1: Sample Problem (For 50 Vendors) Results for the Vendor Penalty Schema and Return Threshold Model

Vendor	VendorPenalty FeeMinFor1	OrginalVendor PenaltyFee1	NewVendorPe naltyFee1	VendorPenalty FeeMaxFor1	VendorPenalty FeeMinFor2	OrginalVendor PenaltyFee2	NewVendorPe naltyFee2	VendorPenalty FeeMaxFor2	VendorPenalty FeeMinFor3	OrginalVendor PenaltyFee3	NewVendorPe naltyFee3	VendorPenalty FeeMaxFor3	Scenari	Total Vendor Return (Old)	Total Vendor Return (Optima	Total Vendor Return Differenc	Last Vendor Return (Old)	Last Vendor Return (Optimal)	Last Vendor Return Difference	Expected Vendor Return (Old)	Expected Vendor Return (Optimal)	Expected Vendor Return Differenc
1	10.00%	11.00%	10.00%	16.00%	17.00%	22.00%	22.34%	31.00%	37.50%	38.50%	39.50%	39.50%	5	$ 81,153	$ 82,215	$ 1,062	$ 40,066	$ 40,066	$ -	$ 29,094	$ 28,925	$ 169
2	7.00%	13.00%	13.90%	15.00%	20.00%	26.00%	20.37%	35.00%	44.50%	45.50%	53.50%	53.50%	6	$212,299	$212,299	$ -	$ 20,016	$ 20,016	$ -	$140,999	$ 128,462	$ 12,536
3	18.00%	24.00%	24.00%	26.00%	39.00%	48.00%	45.33%	52.00%	80.00%	84.00%	88.00%	88.00%	2	$175,725	$175,725	$ -	$ 12,294	$ 12,294	$ -	$ 36,046	$ 34,741	$ 1,305
4	14.00%	20.00%	23.00%	23.00%	35.00%	40.00%	35.95%	47.00%	68.00%	70.00%	75.00%	75.00%	5	$251,639	$252,399	$ 760	$ 44,593	$ 44,593	$ -	$ 51,846	$ 51,111	$ 735
5	10.00%	16.00%	17.00%	17.00%	28.00%	32.00%	31.67%	40.00%	48.00%	56.00%	56.00%	62.00%	6	$187,976	$187,976	$ -	$ 93,731	$ 93,731	$ -	$109,648	$ 109,646	$ 2
6	5.00%	9.00%	13.00%	13.00%	8.00%	18.00%	15.33%	20.00%	30.50%	31.50%	33.50%	33.50%	3	$168,337	$168,337	$ -	$ 4,460	$ 6,442	$ 1,982	$158,950	$ 158,354	$ 596
7	10.00%	13.00%	10.00%	18.00%	25.00%	26.00%	27.00%	34.00%	38.50%	45.50%	45.50%	49.50%	9	$130,014	$130,014	$ -	$ 79,681	$ 79,681	$ -	$117,036	$ 117,036	$ -
8	7.00%	11.00%	11.00%	12.00%	12.00%	22.00%	18.00%	26.00%	29.50%	38.50%	44.50%	44.50%	2	$210,624	$210,624	$ -	$ 14,191	$ 14,191	$ -	$ 78,028	$ 66,577	$ 11,452
9	4.00%	9.00%	4.00%	16.00%	17.00%	18.00%	25.00%	25.00%	25.50%	31.50%	36.50%	36.50%	4	$131,379	$158,489	$ 27,110	$ 62,105	$ 77,394	$ 15,289	$ 7,685	$ 3,415	$ 4,269
10	9.00%	13.00%	9.00%	18.00%	16.00%	26.00%	33.00%	33.00%	41.50%	45.50%	52.50%	52.50%	7	$243,172	$284,591	$ 41,419	$ 227,249	$ 266,219	$ 38,970	$ 17,137	$ 11,864	$ 5,273
11	14.00%	15.00%	16.00%	16.00%	20.00%	30.00%	27.45%	34.00%	48.50%	52.50%	56.50%	56.50%	5	$281,770	$283,565	$ 1,795	$ 35,887	$ 35,887	$ -	$ 83,809	$ 79,729	$ 4,080
12	4.00%	10.00%	11.08%	15.00%	19.00%	20.00%	19.00%	30.00%	34.00%	35.00%	36.00%	36.00%	5	$132,122	$132,198	$ 76	$ 29,838	$ 29,838	$ -	$ 32,970	$ 32,813	$ 157
13	10.00%	13.00%	10.00%	18.00%	21.00%	26.00%	27.00%	33.00%	35.50%	45.50%	45.50%	53.50%	9	$194,698	$194,698	$ -	$ 118,407	$ 118,407	$ -	$178,667	$ 178,667	$ -
14	8.00%	9.00%	8.00%	16.00%	8.00%	18.00%	14.45%	26.00%	25.50%	31.50%	39.50%	39.50%	9	$170,639	$176,526	$ 5,887	$ 147,460	$ 147,460	$ -	$ 95,912	$ 82,820	$ 13,092
15	16.00%	19.00%	16.00%	26.00%	32.00%	38.00%	39.00%	48.00%	63.50%	66.50%	66.50%	76.50%	8	$207,887	$207,887	$ -	$ 169,647	$ 169,647	$ -	$ 30,990	$ 29,070	$ 1,920
16	3.00%	9.00%	9.13%	10.00%	17.00%	18.00%	17.00%	23.00%	25.50%	31.50%	39.50%	39.50%	5	$ 72,330	$ 79,869	$ 7,539	$ 6,543	$ 6,543	$ -	$ 10,535	$ 10,313	$ 222
17	7.00%	12.00%	7.00%	14.00%	16.00%	24.00%	23.05%	29.00%	40.00%	42.00%	48.00%	48.00%	9	$151,499	$155,231	$ 3,732	$ 125,372	$ 125,372	$ -	$ 82,711	$ 76,617	$ 6,094
18	22.00%	23.00%	22.00%	26.00%	37.00%	46.00%	46.33%	50.00%	76.50%	80.50%	80.50%	83.50%	8	$478,621	$478,621	$ -	$ 279,122	$ 279,122	$ -	$ 76,935	$ 75,758	$ 1,177
19	8.00%	12.00%	11.78%	14.00%	20.00%	24.00%	25.00%	25.00%	36.00%	42.00%	43.00%	43.00%	4	$159,076	$163,598	$ 4,522	$ 16,467	$ 16,467	$ -	$ 2,036	$ 1,998	$ 38
20	8.00%	15.00%	18.00%	20.00%	29.00%	30.00%	29.00%	35.00%	45.50%	52.50%	52.50%	58.50%	9	$148,397	$148,397	$ -	$ 108,577	$ 108,577	$ -	$116,223	$ 116,223	$ -
21	17.00%	19.00%	24.00%	24.00%	35.00%	38.00%	36.33%	44.00%	64.50%	66.50%	66.50%	68.50%	9	$256,235	$256,235	$ -	$ 158,215	$ 158,215	$ -	$177,427	$ 177,427	$ -
22	4.00%	7.00%	7.83%	12.00%	13.00%	14.00%	13.00%	23.00%	23.50%	24.50%	25.59%	28.50%	6	$124,109	$124,109	$ -	$ 23,514	$ 23,514	$ -	$117,257	$ 116,954	$ 304
23	4.00%	8.00%	4.00%	11.00%	9.00%	16.00%	17.33%	18.00%	27.00%	28.00%	28.00%	37.00%	8	$128,286	$128,286	$ -	$ 103,167	$ 103,167	$ -	$ 31,234	$ 28,492	$ 2,742
24	14.00%	21.00%	15.00%	24.00%	35.00%	42.00%	44.00%	44.00%	69.50%	73.50%	73.50%	76.50%	8	$371,898	$371,898	$ -	$ 246,509	$ 246,509	$ -	$168,614	$ 167,788	$ 826
25	19.00%	21.00%	19.00%	24.00%	40.00%	42.00%	42.68%	44.00%	69.50%	73.50%	79.50%	79.50%	5	$400,019	$416,406	$ 16,386	$ 196,375	$ 196,375	$ -	$124,585	$ 123,421	$ 1,164
26	8.00%	12.00%	8.00%	17.00%	22.00%	24.00%	25.33%	34.00%	36.00%	42.00%	42.00%	44.00%	8	$248,838	$248,838	$ -	$ 192,709	$ 192,709	$ -	$ 46,213	$ 41,868	$ 4,345
27	5.00%	10.00%	5.00%	17.00%	16.00%	20.00%	22.58%	29.00%	26.00%	35.00%	42.00%	42.00%	5	$209,661	$234,744	$ 25,083	$ 72,954	$ 72,954	$ -	$ 51,029	$ 48,197	$ 2,832
28	15.00%	20.00%	20.00%	24.00%	36.00%	40.00%	36.00%	43.00%	67.00%	70.00%	76.00%	76.00%	2	$322,039	$322,039	$ -	$ 7,183	$ 7,183	$ -	$143,982	$ 131,884	$ 12,098
29	6.00%	9.00%	6.00%	12.00%	13.00%	18.00%	24.00%	24.00%	24.50%	31.50%	36.50%	36.50%	4	$ 65,128	$ 78,050	$ 12,922	$ 9,655	$ 9,772	$ 117	$ 3,372	$ 2,248	$ 1,124
30	15.00%	18.00%	18.64%	25.00%	26.00%	36.00%	29.79%	38.00%	58.00%	63.00%	72.00%	72.00%	5	$187,790	$187,790	$ -	$ 16,186	$ 16,186	$ -	$ 54,545	$ 47,924	$ 6,621
31	17.00%	18.00%	25.00%	25.00%	33.00%	36.00%	33.00%	44.00%	53.00%	63.00%	64.00%	64.00%	3	$211,471	$211,471	$ -	$ 6,152	$ 8,545	$ 2,392	$119,262	$ 117,798	$ 1,464
32	13.00%	20.00%	25.00%	25.00%	31.00%	40.00%	31.67%	50.00%	63.00%	70.00%	80.00%	80.00%	3	$160,686	$160,686	$ -	$ 9,905	$ 12,381	$ 2,476	$148,360	$ 146,599	$ 1,761
33	23.00%	24.00%	23.00%	30.00%	47.00%	48.00%	48.33%	52.00%	78.00%	84.00%	84.00%	88.00%	8	$374,502	$374,502	$ -	$ 243,095	$ 243,095	$ -	$ 59,396	$ 58,508	$ 888
34	20.00%	23.00%	24.00%	24.00%	41.00%	46.00%	42.51%	49.00%	74.50%	80.50%	86.50%	86.50%	5	$243,229	$245,132	$ 1,903	$ 27,319	$ 27,319	$ -	$ 69,737	$ 66,515	$ 3,222
35	7.00%	9.00%	7.00%	13.00%	11.00%	18.00%	18.67%	20.00%	23.50%	31.50%	31.50%	36.50%	8	$115,140	$115,140	$ -	$ 103,689	$ 103,689	$ -	$ 30,600	$ 29,602	$ 999
36	19.00%	23.00%	19.00%	24.00%	43.00%	46.00%	47.33%	52.00%	75.50%	80.50%	80.50%	84.50%	8	$421,015	$421,015	$ -	$ 352,384	$ 352,384	$ -	$182,961	$ 182,163	$ 798
37	21.00%	23.00%	23.00%	26.00%	39.00%	46.00%	44.67%	47.00%	77.50%	80.50%	82.50%	82.50%	5	$216,104	$216,104	$ -	$ 15,535	$ 15,535	$ -	$ 33,455	$ 32,935	$ 520
38	16.00%	17.00%	17.00%	23.00%	27.00%	34.00%	30.00%	40.00%	57.50%	59.50%	65.50%	65.50%	1	$156,587	$156,587	$ -	$ 5,490	$ 5,490	$ -	$ 3,030	$ 3,030	$ -
39	10.00%	15.00%	21.00%	21.00%	20.00%	30.00%	22.33%	38.00%	47.50%	52.50%	61.00%	62.50%	6	$182,032	$182,032	$ -	$ 33,352	$ 33,352	$ -	$168,902	$ 166,776	$ 2,126
40	20.00%	23.00%	20.00%	30.00%	43.00%	46.00%	47.00%	56.00%	72.50%	80.50%	80.50%	87.50%	8	$214,046	$214,046	$ -	$ 174,349	$ 174,349	$ -	$ 68,516	$ 67,679	$ 837
41	17.00%	24.00%	24.00%	27.00%	40.00%	48.00%	46.00%	50.00%	76.00%	84.00%	87.00%	87.00%	2	$208,525	$208,525	$ -	$ 13,562	$ 13,562	$ -	$ 24,510	$ 24,109	$ 401
42	8.00%	12.00%	8.00%	15.00%	18.00%	24.00%	27.85%	34.00%	41.00%	42.00%	52.00%	52.00%	5	$106,818	$124,330	$ 17,512	$ 23,494	$ 23,494	$ -	$ 20,954	$ 20,547	$ 407
43	5.00%	10.00%	13.84%	17.00%	17.00%	20.00%	17.00%	25.00%	27.00%	35.00%	45.00%	45.00%	5	$143,214	$158,397	$ 15,183	$ 36,432	$ 36,432	$ -	$ 65,315	$ 60,983	$ 4,333
44	18.00%	24.00%	18.00%	31.00%	44.00%	48.00%	45.95%	52.00%	79.00%	84.00%	92.00%	92.00%	9	$306,425	$309,946	$ 3,520	$ 269,461	$ 269,461	$ -	$231,517	$ 227,904	$ 3,614
45	12.00%	17.00%	12.00%	20.00%	27.00%	34.00%	35.73%	39.00%	55.50%	59.50%	67.50%	67.50%	5	$307,134	$328,029	$ 20,894	$ 148,719	$ 148,719	$ -	$ 88,890	$ 85,845	$ 3,045
46	13.00%	20.00%	17.00%	21.00%	37.00%	40.00%	41.00%	41.00%	65.00%	70.00%	70.00%	79.00%	8	$375,626	$375,626	$ -	$ 296,653	$ 296,653	$ -	$ 75,579	$ 72,773	$ 2,806
47	16.00%	21.00%	22.00%	22.00%	37.00%	42.00%	40.68%	44.00%	66.50%	73.50%	80.50%	80.50%	5	$229,549	$238,179	$ 8,630	$ 41,318	$ 41,318	$ -	$ 90,578	$ 89,034	$ 1,543
48	6.00%	8.00%	8.00%	12.00%	9.00%	16.00%	12.00%	25.00%	24.00%	28.00%	34.00%	34.00%	2	$ 91,097	$ 91,097	$ -	$ 4,114	$ 4,114	$ -	$ 31,701	$ 25,402	$ 6,298
49	5.00%	7.00%	7.00%	12.00%	5.00%	14.00%	8.67%	24.00%	15.50%	24.50%	32.50%	32.50%	2	$ 62,137	$ 62,137	$ -	$ 3,873	$ 3,873	$ -	$ 5,736	$ 5,241	$ 494
50	7.00%	9.00%	7.00%	14.00%	11.00%	18.00%	18.67%	23.00%	24.50%	31.50%	31.50%	34.50%	8	$150,957	$150,957	$ -	$ 83,477	$ 83,477	$ -	$ 31,993	$ 30,382	$ 1,611

Table A1.2: Optimal Solutions Found for the Vendor Penalty Schema Problem

Vendor	VendorReturnBreakMinFor1	OrginalVendorReturnBreak1	NewVendorReturnBreak1	VendorReturnBreakMaxFo	VendorPenaltyFeeMinFor2	OrginalVendorReturnBreak	NewVendorReturnBreak2	VendorReturnBreakMaxFo	Scenario (L x E)	Total Vendor Return (Old)	Total Vendor Return (Optima	Total Vendor Return Differenc	Last Vendor Return (Old)	Last Vendor Return (Optimal)	Last Vendor Return Difference	Expected Vendor Return (Old)	Expected Vendor Return (Optimal)	Expected Vendor Return Differenc
1	28,957	64,843	96,675	96,675	212,870	259,372	212,870	326,476	5 x 3	$181,560	$189,145	$ 7,584	$ 140,601	$ 148,186	$ 7,584	$ 59,554	$ 53,187	$ 6,366
2	71,415	106,133	71,415	152,420	381,146	424,532	447,677	483,123	3 x 5	$193,162	$193,162	$ -	$ 79,711	$ 84,224	$ 4,513	$141,631	$ 141,631	$ -
3	97,477	140,406	97,477	191,524	520,799	561,624	590,243	592,584	1 x 5	$452,107	$452,107	$ -	$ 16,837	$ 16,837	$ -	$392,396	$ 392,396	$ -
4	61,901	92,260	72,400	126,214	326,605	369,040	382,280	412,952	5 x 4	$271,244	$271,244	$ -	$ 192,313	$ 192,313	$ -	$145,354	$ 145,354	$ -
5	1,863	50,458	50,458	88,747	152,226	201,832	155,446	261,181	4 x 3	$134,218	$147,438	$ 13,220	$ 49,482	$ 49,482	$ -	$ 30,607	$ 30,607	$ -
6	14,714	51,757	44,190	93,663	158,770	207,028	212,073	246,795	4 x 4	$173,904	$173,904	$ -	$ 116,132	$ 116,132	$ -	$ 91,190	$ 91,190	$ -
7	18,979	57,090	20,044	113,540	183,261	228,360	190,727	283,081	4 x 2	$159,852	$178,551	$ 18,699	$ 83,070	$ 101,770	$ 18,699	$ 4,009	$ 4,009	-$ 0
8	64,519	107,388	125,699	176,403	387,282	429,552	387,282	464,914	1 x 2	$225,515	$232,279	$ 6,764	$ 7,497	$ 7,497	$ -	$ 21,602	$ 18,855	$ 2,747
9	95,764	139,680	95,764	188,919	515,349	558,720	587,997	605,616	5 x 5	$273,773	$273,773	$ -	$ 220,375	$ 220,375	$ -	$235,935	$ 235,935	$ -
10	42,802	81,744	131,512	131,512	282,854	326,976	293,797	372,140	5 x 5	$ 80,109	$ 80,109	$ -	$ 76,132	$ 76,132	$ -	$ 57,797	$ 57,797	$ -
11	65,299	104,838	91,509	159,490	374,798	419,352	428,238	475,701	2 x 4	$322,901	$322,901	$ -	$ 21,036	$ 21,940	$ 904	$168,293	$ 168,293	$ -
12	103,792	148,101	103,792	203,512	543,149	592,404	621,943	638,344	1 x 5	$414,683	$414,683	$ -	$ 10,736	$ 10,736	$ -	$400,359	$ 400,359	$ -
13	11,274	51,997	51,997	96,701	164,825	207,988	164,825	246,777	2 x 2	$101,914	$110,978	$ 9,064	$ 19,194	$ 19,194	$ -	$ 9,453	$ 9,453	$ -
14	46,396	82,811	46,396	134,957	286,867	331,244	355,521	389,804	3 x 5	$266,651	$266,651	$ -	$ 58,549	$ 66,925	$ 8,375	$264,585	$ 264,585	$ -
15	68,177	100,808	100,808	132,395	354,125	403,232	354,125	471,670	3 x 3	$211,697	$222,746	$ 11,049	$ 81,795	$ 81,795	$ -	$ 35,965	$ 35,965	$ -
16	102,508	141,108	180,900	180,900	523,426	564,432	523,426	611,970	5 x 3	$276,572	$279,612	$ 3,040	$ 265,486	$ 268,527	$ 3,040	$125,757	$ 120,186	$ 5,571
17	28,541	67,528	38,058	106,812	221,114	270,112	221,114	336,702	3 x 2	$ 94,539	$104,836	$ 10,297	$ 31,597	$ 34,544	$ 2,947	$ 3,806	$ 3,806	$ -
18	81,705	129,748	176,234	176,234	471,807	518,992	471,807	590,641	1 x 3	$181,647	$184,076	$ 2,429	$ 4,714	$ 4,714	$ -	$ 54,340	$ 49,691	$ 4,649
19	97,652	140,300	140,300	195,693	515,262	561,200	515,262	627,045	3 x 3	$216,062	$223,642	$ 7,580	$ 54,058	$ 54,058	$ -	$ 54,181	$ 54,181	$ -
20	82,053	127,968	82,053	171,222	466,448	511,872	542,482	556,298	3 x 5	$125,409	$125,409	$ -	$ 16,432	$ 19,646	$ 3,214	$ 85,734	$ 85,734	$ -
21	49,478	84,551	129,669	129,669	295,210	338,204	308,125	399,059	2 x 4	$272,254	$272,254	$ -	$ 13,071	$ 13,071	$ -	$152,601	$ 152,601	$ -
22	51,477	87,878	87,878	128,348	310,041	351,512	316,743	420,735	4 x 3	$147,635	$153,893	$ 6,258	$ 65,473	$ 65,473	$ -	$ 36,440	$ 36,440	$ -
23	66,705	100,713	100,713	174,046	356,925	402,852	402,852	452,131	3 x 4	$310,196	$310,196	$ -	$ 99,577	$ 99,577	$ -	$189,965	$ 189,965	$ -
24	20,790	55,146	54,133	103,918	172,997	220,584	221,259	264,729	2 x 5	$193,011	$193,011	$ -	$ 13,533	$ 13,533	$ -	$147,633	$ 147,633	$ -
25	85,078	119,199	119,199	180,595	432,423	476,796	432,423	547,240	3 x 3	$417,197	$433,836	$ 16,640	$ 72,457	$ 72,457	$ -	$163,179	$ 163,179	$ -
26	11,318	51,296	118,738	120,616	160,223	205,184	160,223	239,150	5 x 3	$ 86,177	$ 86,177	$ -	$ 81,683	$ 81,683	$ -	$ 26,837	$ 18,744	$ 8,093
27	28,470	66,274	122,903	122,903	217,343	265,096	227,343	312,447	2 x 4	$167,010	$167,010	$ -	$ 6,872	$ 6,872	$ -	$101,552	$ 101,552	$ -
28	18,870	57,369	57,369	111,716	184,189	229,476	184,189	273,271	3 x 3	$128,507	$139,375	$ 10,869	$ 28,820	$ 28,820	$ -	$ 36,422	$ 36,422	$ -
29	75,081	115,930	75,081	184,865	422,474	463,720	490,953	494,319	5 x 5	$113,611	$113,611	$ -	$ 97,301	$ 97,301	$ -	$112,347	$ 112,347	$ -
30	60,072	93,836	93,836	153,871	325,529	375,344	375,344	426,529	2 x 5	$249,604	$249,604	$ -	$ 22,675	$ 22,675	$ -	$184,307	$ 184,307	$ -
31	88,435	136,227	164,157	169,835	502,233	544,908	502,233	588,880	1 x 2	$476,795	$485,815	$ 9,021	$ 22,029	$ 22,029	$ -	$ 48,022	$ 41,039	$ 6,983
32	86,176	116,803	116,803	157,509	418,643	467,212	418,643	505,245	3 x 3	$261,639	$273,295	$ 11,657	$ 90,455	$ 90,455	$ -	$110,707	$ 110,707	$ -
33	74,320	116,508	116,508	184,105	425,512	466,032	425,512	515,842	3 x 3	$228,356	$236,865	$ 8,509	$ 51,156	$ 51,156	$ -	$ 95,708	$ 95,708	$ -
34	100,565	138,998	100,565	181,036	515,863	555,992	581,614	617,176	3 x 5	$155,678	$155,678	$ -	$ 31,386	$ 34,461	$ 3,075	$127,509	$ 127,509	$ -
35	68,025	110,072	68,025	179,505	393,844	440,288	468,319	476,060	3 x 5	$138,691	$138,691	$ -	$ 23,480	$ 27,264	$ 3,784	$131,356	$ 131,356	$ -
36	26,721	57,610	55,468	93,072	181,725	230,440	181,725	276,883	4 x 2	$137,112	$149,898	$ 12,786	$ 81,894	$ 94,680	$ 12,786	$ 9,430	$ 9,430	$ -
37	94,658	140,472	140,472	185,552	517,699	561,888	517,699	634,898	3 x 3	$393,322	$406,578	$ 13,257	$ 178,049	$ 178,049	$ -	$135,192	$ 135,192	$ -
38	47,722	90,961	94,896	150,025	314,550	363,844	361,221	425,839	5 x 4	$254,691	$254,691	$ -	$ 173,712	$ 173,712	$ -	$126,296	$ 125,509	$ 787
39	19,627	59,208	19,627	103,320	195,557	236,832	263,219	285,253	4 x 5	$198,939	$198,939	$ -	$ 83,997	$ 93,497	$ 9,499	$152,058	$ 152,058	$ -
40	45,415	86,724	157,409	157,409	296,992	346,896	299,773	391,859	5 x 4	$157,838	$157,838	$ -	$ 131,703	$ 131,703	$ -	$ 94,273	$ 94,273	$ -
41	34,061	83,656	34,061	151,467	286,942	334,624	312,820	404,281	4 x 1	$269,372	$288,302	$ 18,929	$ 124,656	$ 136,063	$ 11,407	$ 7,541	$ 7,541	$ -
42	33,012	70,441	33,012	104,570	240,364	281,764	283,787	347,766	4 x 1	$207,097	$214,319	$ 7,223	$ 105,035	$ 112,258	$ 7,223	$ 6,808	$ 6,808	$ -
43	71,795	109,311	72,863	170,919	394,754	437,244	461,543	479,765	3 x 4	$168,339	$168,339	$ -	$ 28,224	$ 32,234	$ 4,009	$ 93,525	$ 93,525	$ -
44	112,449	143,930	112,449	208,348	534,796	575,720	534,796	618,532	5 x 1	$241,802	$252,946	$ 11,144	$ 192,032	$ 203,176	$ 11,144	$ 11,299	$ 11,299	$ -
45	26,187	75,847	102,099	126,171	254,444	303,388	285,887	358,076	4 x 4	$ 95,567	$ 95,567	$ -	$ 56,636	$ 56,636	$ -	$ 44,633	$ 42,271	$ 2,363
46	41,823	76,349	116,763	116,763	260,666	305,396	278,453	343,805	4 x 5	$ 96,200	$ 96,200	$ -	$ 57,228	$ 57,228	$ -	$ 78,109	$ 78,109	$ -
47	55,362	104,428	104,428	159,348	370,486	417,712	417,712	483,008	3 x 5	$219,299	$219,299	$ -	$ 64,891	$ 64,891	$ -	$145,488	$ 145,488	$ -
48	61,166	102,861	104,157	156,850	369,054	411,444	369,054	447,827	5 x 2	$345,613	$360,562	$ 14,949	$ 210,794	$ 225,743	$ 14,949	$ 25,309	$ 24,998	$ 311
49	62,699	107,687	107,687	141,144	382,373	430,748	382,373	494,535	3 x 3	$211,067	$221,225	$ 10,159	$ 48,715	$ 48,715	$ -	$ 32,555	$ 32,555	$ -
50	34,071	72,709	72,709	138,205	249,914	290,836	249,914	352,150	2 x 3	$101,793	$107,931	$ 6,138	$ 12,501	$ 12,501	$ -	$ 32,633	$ 32,633	$ -

TABLE A1.3: Optimal Solutions Found for the Vendor Return Thresholds Problem

Bibliography

Alumur, S. A., Nickel, S., Saldanha-da-Gama, F., and Verter, V. (2012). Multi-period reverse logistics network design. *European Journal of Operational Research* 220(1), 67–78.

Alumur, S. A. and Tari, I. (2014). Collection center location with equity considerations in reverse logistics networks. *INFOR: Information Systems and Operational Research* 52(4), 157–173.

Ambilkar, P., Dohale, V., Gunasekaran, A., and Bilolikar, V. (2021). Product returns management: a comprehensive review and future research agenda. *International Journal of Production Research*, 1–25.

Ashayeri, J. and Tuzkaya, G. (2011). Design of demand driven return supply chain for high-tech products. *Journal of Industrial Engineering and Management* 4(3), 481–503.

Bajor, I. and Babic, D. (2014). Reverse logistics retail level return. *International Journal for Traffic and Transport Engineering* 4(2), 161–170.

Barros, A. I., Dekker, R., and Scholten, V. (1998). A two-level network for recycling sand: a case study. *European Journal of Operational Research* 110(2), 199–214.

Beh, L.-S., Ghobadian, A., He, Q., Gallear, D., and O'Regan, N. (2016). Second-life retailing: a reverse supply chain perspective. *Supply Chain Management: An International Journal*.

Beitelspacher, L. S., Baker, T. L., Rapp, A., and Grewal, D. (2018). Understanding the long-term implications of retailer returns in business-to-business relationships. *Journal of the Academy of Marketing Science* 46(2), 252–272.

Bernon, M. and Cullen, J. (2007). An integrated approach to managing reverse logistics. *International Journal of Logistics: Research and Applications* 10(1), 41–56.

Bernon, M., Cullen, J., and Gorst, J. (2016). Online retail returns management: Integration within an omni-channel distribution context. *International Journal of Physical Distribution & Logistics Management*.

Bernon, M., Rossi, S., and Cullen, J. (2011). Retail reverse logistics: a call and grounding framework for research. *International Journal of Physical Distribution & Logistics Management* 41(5), 484–510.

Bernon, M., Tjahjono, B., and Ripanti, E. F. (2018). Aligning retail reverse logistics practice with circular economy values: an exploratory framework. *Production Planning & Control* 29(6), 483–497.

Bernon, M., Upperton, J., Bastl, M., and Cullen, J. (2013). An exploration of supply chain integration in the retail product returns process. *International Journal of Physical Distribution & Logistics Management.*

Borba, J. L. G. de, Magalhaes, M. R. de, Filgueiras, R. S., and Bouzon, M. (2020). Barriers in omnichannel retailing returns: a conceptual framework. *International Journal of Retail & Distribution Management.*

Caruso, C., Colorni, A., and Paruccini, M. (1993). The regional urban solid waste management system: A modelling approach. *European Journal of Operational Research* 70(1), 16–30.

Chaves, G. d. L. D. and Alcantara, R. L. C. (2006). Reverse logistics and the relation between industry and retail in the after-sale reverse flow management. In: *Third International Conference on Production Research-Americas' Region*, 1–14.

Das, K. (2012). Integrating reverse logistics into the strategic planning of a supply chain. *International Journal of Production Research* 50(5), 1438–1456.

Das, K. and Chowdhury, A. H. (2012). Designing a reverse logistics network for optimal collection, recovery and quality-based product-mix planning. *International Journal of Production Economics* 135(1), 209–221.

Dat, L. Q., Linh, D. T. T., Chou, S.-Y., and Vincent, F. Y. (2012). Optimizing reverse logistic costs for recycling end-of-life electrical and electronic products. *Expert Systems with Applications* 39(7), 6380–6387.

Demajorovic, J., Santos, J. B., and Silva Oliveira, L. da (2019). Reverse logistics in retail: barriers and motivation to products and packaging return. *Revista de Administracao da Universidade Federal de Santa Maria* 12(5), 911–930.

Dias, K. T. and Braga Junior, S. S. (2016). The use of reverse logistics for waste management in a Brazilian grocery retailer. *Waste Management & Research* 34(1), 22–29.

Dias, K., Braga, S., Silva, D., and Satolo, E. (2019). Reverse Logistics for Return Management in Retail: A Systematic Literature Review from 2007 to 2016. *New Global Perspectives on Industrial Engineering and Management*, 145–153.

Eskandarpour, M., Zegordi, S. H., and Nikbakhsh, E. (2013). A parallel variable neighborhood search for the multi-objective sustainable post-sales network design problem. *International Journal of Production Economics* 145(1), 117–131.

Ferguson, M. E., Fleischmann, M., and Souza, G. C. (2011). A Profit-Maximizing Approach to Disposition Decisions for Product Returns. *Decision Sciences* 42(3), 773–798.

Fleischmann, M., Bloemhof-Ruwaard, J. M., Dekker, R., Van der Laan, E., Van Nunen, J. A., and Van Wassenhove, L. N. (1997). Quantitative models for reverse logistics: A review. *European journal of operational research* 103(1), 1–17.

Frei, R., Jack, L., and Krzyzaniak, S.-A. (2020). Sustainable reverse supply chains and circular economy in multichannel retail returns. *Business Strategy and the Environment* 29(5), 1925–1940.

Godichaud, M. and Amodeo, L. (2015). Efficient multi-objective optimization of supply chain with returned products. *J Manuf Syst* 37, 683–691.

Guo, P., Jia, Y., Gan, J., and Li, X. (2021). Optimal Pricing and Ordering Strategies with a Flexible Return Strategy under Uncertainty. *Mathematics* 9(17), 2097.

Guo, S., Shen, B., Choi, T.-M., and Jung, S. (2017). A review on supply chain contracts in reverse logistics: Supply chain structures and channel leaderships. *Journal of Cleaner Production* 144, 387–402.

Gustafsson, E., Jonsson, P., and Holmstrom, J. (2021). Reducing retail supply chain costs of product returns using digital product fitting. *International Journal of Physical Distribution & Logistics Management*.

Horvath, P. A., Autry, C. W., and Wilcox, W. E. (2005). Liquidity implications of reverse logistics for retailers: A Markov chain approach. *Journal of retailing* 81(3), 191–203.

Hosseinzadeh, M. and Roghanian, E. (2012). An optimization model for reverse logistics network under stochastic environment using genetic algorithm. *International Journal of Business and Social Science* 3(12), 1–17.

Jack, E. P., Powers, T. L., and Skinner, L. (2010). Reverse logistics capabilities: antecedents and cost savings. *International Journal of Physical Distribution & Logistics Management*.

Jayaraman, V., Guide Jr, V., and Srivastava, R. (1999). A closed-loop logistics model for remanufacturing. *Journal of the operational research society* 50(5), 497–508.

Jeong, I.-J. (2012). A centralized/decentralized design of a full return contract for a risk-free manufacturer and a risk-neutral retailer under partial information sharing. *International Journal of Production Economics* 136(1), 110–115.

John, S. T., Sridharan, R., Sachdeva, A., Sharma, V., and Garg, R. K. (2015). Modelling and analysis of network design for a reverse supply chain. *Journal of Manufacturing Technology Management* 26(6), 853–867.

Junior, B., Junqueira, K. T. S. D., Silva, D. da, and Oliveira, S. C. (2020). Analise sobre a disposicao para implementacao da logistica reversa no varejo supermercadista. *International Journal of Innovation* 8(2), 204–222.

Kaboudani, Y., Ghodsypour, S. H., Kia, H., and Shahmardan, A. (2020). Vehicle routing and scheduling in cross docks with forward and reverse logistics. *Operational Research* 20(3), 1589–1622.

Kopicki, R., Berg, M. J., and Legg, L. (1993). Reuse and recycling-reverse logistics opportunities.

Krikke, H. R., Kooi, E., and Schuur, P. C. (1999). Network design in reverse logistics: a quantitative model. In: *New trends in distribution logistics*. Springer, 45–61.

Kroon, L. and Vrijens, G. (1995). Returnable containers: an example of reverse logistics. *International Journal of Physical Distribution & Logistics Management* 25(2), 56–68.

Li, S., Wang, N., Jia, T., He, Z., and Liang, H. (2016). Multiobjective Optimization for Multiperiod Reverse Logistics Network Design. *IEEE Transactions on Engineering Management* 63(2), 223–236.

Louwers, D., Kip, B. J., Peters, E., Souren, F., and Flapper, S. D. P. (1999). A facility location allocation model for reusing carpet materials. *Computers & Industrial Engineering* 36(4), 855–869.

Marin, A. and Pelegrin, B. (1998). The return plant location problem: Modelling and resolution. *European journal of operational research* 104(2), 375–392.

Matsui, K. (2010). Returns policy, new model introduction, and consumer welfare. *International Journal of Production Economics* 124(2), 299–309.

Melachrinoudis, E., Min, H., and Wu, X. (1995). A multiobjective model for the dynamic location of landfills. *Location Science* 3(3), 143–166.

Min, H. (1989). A bicriterion reverse distribution model for product recall. *Omega* 17(5), 483–490.

Moise, M. et al. (2008). The importance of reverse logistics for retail activity. *The Amfiteatru Economic journal* 10(24), 192–209.

Mostert, W., Niemann, W., and Kotze, T. (2017). Supply chain integration in the product return process: A study of consumer electronics retailers. *Acta Commercii* 17(1), 1–16.

Nenes, G. and Nikolaidis, Y. (2012). A multi-period model for managing used product returns. *International Journal of Production Research* 50(5), 1360–1376.

Niknejad, A. and Petrovic, D. (2014). Optimisation of integrated reverse logistics networks with different product recovery routes. *European Journal of Operational Research* 238(1), 143–154.

Olariu, I. (2014). An Overview On Retail Reverse Logistics. *Studies and Scientific Researches. Economics Edition* (19).

Panigrahi, S. K., Kar, F. W., Fen, T. A., Hoe, L. K., and Wong, M. (2018). A strategic initiative for successful reverse logistics management in retail industry. *Global Business Review* 19(3_suppl), S151–S175.

Rogers, D. S. and Tibben-Lembke, R. (1999). Going backwards. *Reverse Logistics Trends and Practice. Pittsburgh, PA: Council of Logistics Management.*

Roghanian, E. and Pazhoheshfar, P. (2014). An optimization model for reverse logistics network under stochastic environment by using genetic algorithm. *Journal of Manufacturing Systems* 33(3), 348–356.

Soleimani, H. and Govindan, K. (2014). Reverse logistics network design and planning utilizing conditional value at risk. *European Journal of Operational Research* 237(2), 487–497.

Sonya Hsu, H., Alexander, C. A., and Zhu, Z. (2009). Understanding the reverse logistics operations of a retailer: a pilot study. *Industrial Management & Data Systems* 109(4), 515–531.

Spengler, T., Puchert, H., Penkuhn, T., and Rentz, O. (1997). Environmental integrated production and recycling management. In: *Produktion und Umwelt.* Springer, 239–257.

Stock, J. R. (1992). *Reverse logistics: White paper.* Council of Logistics Management.

Thierry, M. (1997). An analysis of the impact of product recovery management on manufacturing companies. *Ph. D. Dissertation, Erasmus University.*

Tibben-Lembke, R. S. (2002). Life after death: reverse logistics and the product life cycle. *International Journal of Physical Distribution & Logistics Management.*

Tibben-Lembke, R. S. and Rogers, D. S. (2002). Differences between forward and reverse logistics in a retail environment. *Supply Chain Management: An International Journal* 7(5), 271–282.

Tuzkaya, G., Gulsun, B., and Onsel, S. (2011). A methodology for the strategic design of reverse logistics networks and its application in the Turkish white goods industry. *International Journal of Production Research* 49(15), 4543–4571.

Van Engeland, J., Belien, J., De Boeck, L., and De Jaeger, S. (2020). Literature review: Strategic network optimization models in waste reverse supply chains. *Omega* 91, 102012.

Vidovic, M., Dimitrijevic, B., Ratkovic, B., and Simic, V. (2011). A novel covering approach to positioning ELV collection points. *Resources, Conservation and Recycling* 57, 1–9.

Wojanowski, R., Verter, V., and Boyaci, T. (2007). Retail–collection network design under deposit–refund. *Computers & operations research* 34(2), 324–345.

Xu, D. et al. (2019). Research On Strategic Mode And Development Of Reverse Logistics In Retail Industry. *Malaysian E Commerce Journal (MECJ)* 3(1), 26–29.

Yi, P., Huang, M., Guo, L., and Shi, T. (2016). Dual recycling channel decision in retailer oriented closed-loop supply chain for construction machinery remanufacturing. *Journal of Cleaner Production* 137, 1393–1405.

Yuliawati, E., Pratikto, P., Sugiono, S., and Novareza, O. (2021). Development of Cores Acquisition Model With Two Switching Mechanisms in a Retailer-Oriented Closed-Loop Supply Chain System. *Eastern-European Journal of Enterprise Technologies* 2(3), 110.

Yun, Y. (2015). Evaluating Reverse Logistics Networks with Centralized Centers, an adaptive genetic algorithm approach based on fuzzy logic controller. *International Journal of Intelligent Information Systems*.