Assortment Planning for Multiple Quality Levels Using Locational Choice

Mark Mcelreath
Clemson University, markmcelreath@gmail.com

Follow this and additional works at: http://tigerprints.clemson.edu/all_dissertations
Part of the Operations Research, Systems Engineering and Industrial Engineering Commons

Recommended Citation
ASSORTMENT PLANNING FOR MULTIPLE QUALITY LEVELS USING LOCATIONAL CHOICE

A Dissertation
Presented to
the Graduate School of
Clemson University

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

by
Mark H. McElreath
December 2010

Accepted by:
Dr. Maria E. Mayorga, Committee Chair
Dr. William G. Ferrell
Dr. Mary E. Kurz
Dr. Kevin M. Taaffe
Abstract

Assortment planning is the process in which a retailer selects a product line to offer to customers and is a key determinant of a retailer’s profit. We consider the assortment planning problem using a locational choice model for customer product selection and allow for both horizontal and vertical product differentiation. When the distribution of customer preference is unimodal, the optimal solutions for this problem are unknown. We propose two solution philosophies for generating product assortments. First, we introduce a metaheuristic representation for the problem and test the performance of three metaheuristic techniques. We suggest that a tabu search or genetic algorithm may be the best technique for the problem depending on the parameters. Next, we introduce a combined dynamic programming and line search approach for generating optimal solutions. We use this technique to explore the properties of the optimal solution and suggest instances where this technique is preferable to the metaheuristic methods. We then propose a new model which allows for heterogeneous quality preferences among the customer population. This model allows for more realistic customer product selection but also increases the complexity of the problem. We give mathematical properties of optimal solutions to the heterogeneous model and propose a new metaheuristic representation and a genetic algorithm for solving the problem.
Dedication

To my parents, Robert and Susan McElreath, for making this possible. Thank you.
Acknowledgments

I would like to thank my advisor, Dr. Maria Mayorga, for everything she has taught me and for the countless hours she has spent helping me throughout these years. I believe that she is truly a model for what anyone can hope for out of an advisor and a teacher.

I would like to thank my committee members, Dr. William Ferrell, Dr. Mary Beth Kurz, and Dr. Kevin Taaffe. Each one has helped me tremendously – not only with this document but with knowledge, inspiration, advice, friendship and more.

I would also like to thank the entire faculty and staff of the Clemson Industrial Engineering department. It has been a pleasure to be around you all for many years.
# Table of Contents

Title Page ........................................................................................................... i
Abstract ............................................................................................................... ii
Dedication .......................................................................................................... iii
Acknowledgments ............................................................................................. iv
List of Tables ...................................................................................................... vii
List of Figures .................................................................................................... viii

1 Introduction ..................................................................................................... 1
   1.1 Customer Choice ..................................................................................... 2
   1.2 Literature Review ................................................................................... 5
   1.3 Assortment Planning with Locational Choice ........................................ 7

2 Model Definition and Properties of the Optimal Solution for Homogeneous Quality Preferences ........................................................................................................... 9
   2.1 Model Description .................................................................................. 10
   2.2 Properties of the Optimal Solution ....................................................... 13

3 A Metaheuristic Study for Homogeneous Quality Preferences ............. 16
   3.1 Common Elements ............................................................................... 17
   3.2 Genetic Algorithms .............................................................................. 21
   3.3 Simulated Annealing ............................................................................ 23
   3.4 Tabu Search ......................................................................................... 24
   3.5 Computational Experiments ............................................................... 24
   3.6 Results ................................................................................................. 25
   3.7 Conclusions ......................................................................................... 27

4 Assortment Planning with Multiple Quality Levels: A Dynamic Programming Approach ........................................................................................................... 32
   4.1 Dynamic Program ................................................................................ 32
   4.2 Line Search .......................................................................................... 40
   4.3 Computational Studies ......................................................................... 44
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Assortment Planning with Heterogeneous Customer Quality Preferences</td>
<td>52</td>
</tr>
<tr>
<td>5.1</td>
<td>Model</td>
<td>53</td>
</tr>
<tr>
<td>5.2</td>
<td>Analysis</td>
<td>58</td>
</tr>
<tr>
<td>5.3</td>
<td>Genetic Algorithm Solutions</td>
<td>67</td>
</tr>
<tr>
<td>5.4</td>
<td>Computational Studies</td>
<td>69</td>
</tr>
<tr>
<td>6</td>
<td>Conclusions and Future Work</td>
<td>76</td>
</tr>
<tr>
<td>6.1</td>
<td>Contributions</td>
<td>76</td>
</tr>
<tr>
<td>6.2</td>
<td>Future Work</td>
<td>77</td>
</tr>
<tr>
<td>Appendices</td>
<td></td>
<td>78</td>
</tr>
<tr>
<td>A</td>
<td>Test Cases for Metaheuristic Computational Experiments</td>
<td>79</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>82</td>
</tr>
</tbody>
</table>
List of Tables

3.1  Aggregated Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
3.2  Case Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26
3.3  Statistical Ranking . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

4.1  Three Iterations of Search Example . . . . . . . . . . . . . . . . . . . . . . 44
4.2  Percentage Profit Deviation (rounded to nearest hundredth percent) . . . 50

5.1  An Example of a GA Experiment . . . . . . . . . . . . . . . . . . . . . . . . 73
5.2  Variance of Test Cases . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 74
List of Figures

2.1 Illustration of Homogeneous Model Definitions ......................... 11
3.1 Sample Solution Representation. ........................................ 19
3.2 Illustration of the Crossover Operator. ................................ 22
3.3 Boxplot of Percent Deviation from Bound ............................... 27
3.4 Pre-classified problem instances ........................................ 31
4.1 Two Examples of the $J(b)$ Function .................................. 41
4.2 An Example of Solution Mapping ....................................... 46
4.3 $\Pi$ as a Function of $\sigma$ and $K$ .................................... 48
4.4 $n$ as a Function of $\sigma$ and $K$ ...................................... 48
4.5 $n_H$ as a Function of $\sigma$ and $K$ .................................... 49
4.6 $n_L$ as a Function of $\sigma$ and $K$ .................................... 49
5.1 Illustration of Heterogeneous Model Definitions ....................... 54
5.2 A Necessary Overlap Example .......................................... 59
5.3 A Domination Example .................................................. 61
5.4 Illustration of $\Pi(b,y)$ Increasing in $\alpha$ ............................ 66
5.5 Illustration of new representation ...................................... 69
5.6 Objective Value as a Function of $\alpha$ for a GA Experiment ....... 75
Chapter 1

Introduction

An important decision that most retailers face is the question of what products to offer consumers. As most firms have limited resources, they are only able to offer some subset of available products and must choose this subset carefully with the goal of maximizing the profitability of the firm. This decision is greatly aided when the firm has information about its customer base. If the firm can characterize the preferences of its customers, then she can match that information with the characteristics of the potential products to formulate an assortment. To choose an assortment to offer, the firm first starts by considering some set of possible products to offer. The firm then seeks to choose the optimal subset of these products, which is the subset that will maximize expected profit. This subset is the group of products that the firm offers for sale to its customers.

Consider an example of a retailer who begins selling some product. Her supplier gives her a list of all the available sizes she can offer. Each unit sold offers some profit margin to her, so to maximize sales she could potentially offer every available size. But suppose that she pays some cost for every size she offers – i.e. shelf space, warehouse space, organizational costs, etc. In this case, she might find that some sizes are more profitable than others – in fact, some may not be profitable at all. To maximize her profit, she must pick the subset of sizes which will provide her the highest profit. To properly solve this problem, she must
have information about her customers. How many customers will she have? How will the customers make decisions about which size to purchase? How many of each size can she expect to sell? What is the distribution of size preferences in the general population? Are customers willing to substitute if they can not buy their preferred size?

In this dissertation, we will examine the ways in which a retailer can choose an optimal assortment of products. In this chapter we will discuss the assortment planning problem at a high level, giving the history of the problem as well as discussing the greater questions that must be answered before finding a solution. We will discuss a specific case of the assortment planning problem that will serve as the basis for the rest of the document. In Chapter 2 we formalize this model mathematically and Chapters 3 and 4 discuss new solution methodologies for this model. Finally, in Chapter 5, we extend this model further and discuss ways in which this new model can be solved.

1.1 Customer Choice

The first key component in modeling an assortment planning problem is determining how customers will make purchasing decisions. Historically, there are two main ways in which customer choice is modeled – discrete and locational choice. We will first give a brief overview of each of these methods.

1.1.1 Discrete Choice

Discrete choice is the most popular of the two methods and is usually implemented as a multinomial logit (MNL) model. In discrete choice, the retailer has some finite number of mutually exclusive products that she can potentially offer. Each product can be thought of as a bundle of one or more attributes and each customer is a bundle of one or more characteristics. Using these attributes and characteristics, we can then find a probability that each item is selected by a random customer.
While discrete choice is an acceptable model for many cases, problems can arise when considering substitution behavior. In the MNL model, when the first choice of a consumer is not available, he will pick the next available product which shares the most with his preferred product. This choice, however, may not be rational because we do not have a measure of adjacency of substitutes – that is, the model may not be able to tell which is an appropriate substitute by figuring out which alternative is “closest” to the original preference.

For example, consider a man booking one-way airline travel. His preferred product is a 8:30am Delta flight to San Francisco. He finds that this flight is not available but that he has three alternatives to choose from: an 8:45am USAir flight to San Francisco, an 8:30am Delta flight to Chicago, and a 6:30pm Delta flight to San Francisco. Most likely, the man should purchase the 8:45am USAir flight, as it is his intended destination and leaves around the time of his preference. Perhaps the man would choose the 6:30pm flight because he has some strong preference for Delta. It is unlikely, however, that he would choose the 8:30am Delta flight to Chicago, although it shares the most characteristics (time and airline) with his original preference.

To avoid the problem above, in the MNL model one must ensure that all products are equal substitutes. This is because the MNL model implies a property known as the Independence of Irrelevant Alternatives (IIA). An often cited example of the IIA property is the “Red Bus/Blue Bus” example (McFadden [19]). Suppose a person is choosing between a red bus and a car for transportation and he chooses each of these choices with equal probability (he has a 0.50 probability of picking either). If we introduce a blue bus as a third choice to the person, the IIA property implies he will still pick any of the three choices with equal probability (0.33 probability of each). This is counter-intuitive because the color of the bus should be irrelevant to the traveler. The IIA property states that the odds ratios between choices should not be affected by the presence of a new choice. This may create substantial
problems with substitution behavior as in the above example as the demand for the buses would be artificially and unrealistically inflated.

An alternative discrete choice demand model is the exogenous demand model which is sometimes applied. As opposed to being a utility based model, in the exogenous demand model there is a fixed and known demand for each product. The focus of this model is then tied to the substitution behavior of the consumers. Each product is given a probability of being a substitute for another product. The exogenous demand model has two key disadvantages. First, without the presence of a utility model, we can not differentiate between the customer’s original preference and his substitution choices. Second, we have a fixed pool of products and introducing new alternatives requires starting over with a new model. This choice model is typically more applicable to the inventory management step than the assortment planning step.

1.1.2 Locational Choice

A third model, known as a locational choice model, attempts to correct some of the problems with the discrete choice models. In a locational choice model, alternatives can be considered as locations on one or more axes and customer preferences are given as points on these axes. Because a spatial relationship is defined, alternatives have clearly defined distances from each other and from customer preferences. If a customer finds that his preference is not available, he can find the alternatives that are closest to his original preference. He may also find that no alternative exists within an acceptable distance from his preference and may not choose anything. The utility a consumer gets from a product is a function of the distance between the product’s location and his ideal point. Thus, the locational choice model is a utility based model where the firm can control the rate of substitution between products by choosing their locations relative to each other. In the MNL model, customers may substitute between any two products as opposed to just the products which
are located “close” to their preference in the locational choice model. In the exogenous model, substitution is more limited than the MNL model, but must be explicitly defined upon creation of the model.

1.2 Literature Review

A recent survey of assortment planning literature is provided by Kök et al. [9]. The authors first introduce the concepts of product variety (how one product is differentiated from another) and also discuss how inventory is related to assortment planning as well related issues, such as how shelf space planning can impact assortment planning and how consumers perceive variety. They then cover the most popular demand models, including the MNL, locational choice and exogenous demand models as previously described. The authors then review the major assortment planning models available using each of these demand models.

Historically, a common approach is to consider assortment planning and inventory decision together. Many authors have studied the inventory problem for a given and known assortment, including Mussa and Rosen [21], Moorthy [20], Economides [3], and Rajaram and Tang [22]. Further work has been done by authors who combine the assortment problem with the inventory problem. The majority of this work uses a discrete choice model. Mahajan and van Ryzin [24] consider the optimal stocking levels and assortment of a group of products with identical price and cost using multinomial logit in the absence of substitution. Smith and Agrawal [23] consider the case of dynamic substitution and suggest finding an optimal assortment through enumeration. Mahajan and van Ryzin also extend their earlier work by allowing for substitution within a retailer [14] and substitution from a competitor [13]. The discrete choice assortment planning problem has also been studied by Maddah and Bish [12].

A locational model for consumer choice was first proposed by Lancaster [11], based on the
work of Hotelling [7]. Lancaster’s model presented products as a bundle of characteristics and each customer had some preference for each characteristic. He then proposed a utility function where the amount of utility gained by a consumer from a purchase was directly related to the distance between that customer’s preferences and the characteristics of the product he purchased. Lancaster’s model was significant as instead of allowing substitution between any two products, it limited substitution to products that were similar to the customer’s original preference. Gaur and Honhon [4] were the first to study the assortment problem under a locational choice model while incorporating inventory decisions. They show that in the make to order and static substitution problems, the assortment planning problem is separable from the inventory problem. They then solve the static substitution problem, using a locational choice model with horizontal differentiation. Under horizontal differentiation, consumer preferences for an attribute can be modeled as a location on a line (e.g. shoe size) and the distribution of these preferences is known. They show that the optimal assortment consists of products distributed on the attribute space such that there is no substitution among them and market coverage is continuous. When the distribution of consumer preferences for the horizontal attribute is uniform, the exact assortment (location of products on the variety space and number of products) is known. On the other hand, when the distribution of consumer preferences is unimodal, the optimal solution can be found by conducting a single variable line search. Thus, we see that when consumer choice is based only on horizontal differentiation, the optimal assortments can be easily found.

Computational tractability is lost when vertical differentiation among products is allowed. Mayorga ([15],[16]) and Mayorga [17] consider an assortment planning problem in which products are both horizontally and vertically differentiated. With only horizontal differentiation, consumer choice is based only on an individual’s idiosyncratic preferences (e.g., black vs. white earphones). The addition of vertical differentiation allows another type of consumer choice, based on consumers’ taste for quality (e.g., regular vs. noise canceling earphones). The assortment and inventory management problem under different substitution
environments is investigated by Mayorga ([15],[16]) for the cases where consumer preferences for the horizontal attribute are distributed uniformly. This work is also extended by Mayorga et al. [17] to consider more general distribution assumptions. In the case of static substitution, the inventory and assortment problems completely decouple as was the case with Gaur and Honhon [4], such that for a given assortment the optimal inventory levels can be obtained analytically. The assortment problem, however, proves more difficult. Mayorga et al. [17] show that some properties of the optimal horizontal attributes are the same as in Gaur and Honhon [4]; that is, products are spaced such that there is no substitution between them and market coverage is continuous. However, to find the optimal assortment using these properties, one still needs to choose the number of products and the quality level of each product. In general, solving the problem to optimality requires an enumerative search over all possible assortment sizes and quality combinations. Mayorga et al. [17] show that the problem can be greatly simplified in some cases. In particular, they give conditions under which the optimal assortment is composed of products that all have the same quality level, reducing the problem to the previously solved problem with only horizontal differentiation. While they show that over a large region of the parameter space these conditions are indeed satisfied, we are left with an open problem when the conditions are not satisfied. In this case, the optimal assortment can be of either a single quality level or mixed quality levels and neither the optimal number of products to carry or attributes are known.

1.3 Assortment Planning with Locational Choice

For the work that follows, we examine the assortment planning problem with locational choice. Consumers will make purchasing decisions based on a single preference and we will also allow for vertical differentiation. We will consider only the make to order case and, as a result, inventory decisions will not be required. Note that in the static substitution case the inventory decisions are separable and our results will still apply. While many previous authors consider the assortment and inventory problems together, we believe this causes
two problems. First, for the cases in which the inventory and assortment problems are separable, the inventory decisions are not needed to find the optimal assortment. Second, for the cases where the problems are not separable, the authors may only approximate results because of problem complexity.

As an example of our problem, consider a retailer who is selling baseball caps to customers. Customers have some size preference which is determined by the circumference of their heads. Each customer also has some tolerance around this value for which they can still wear the cap (an exact fit is not required). The distribution of size preference can be represented by some continuous probability distribution (e.g. normal). Assume that the retailer may order hats to be manufactured in any circumference (that is, there are no pre-existing sizes). She must choose which (if any) of sizes she will order to maximize her profit, knowing that she will incur a cost for each size that she orders.

In the next chapter, we will define mathematically a model for assortment planning with locational choice. This model allows for horizontal and vertical differentiation between products and assumes that customers have a homogeneous preference for quality attributes.
Chapter 2

Model Definition and Properties of the Optimal Solution for Homogeneous Quality Preferences

In Chapters 3 and 4 we will examine techniques for generating solutions to the assortment problem with a locational choice for customer choice and two quality levels of products. In this chapter, we introduce the model which we will use to generate those solutions. Customer preference for some attribute is distributed along a horizontal axis and products are chosen as locations on this axis (horizontal differentiation). Horizontal differentiation is based on consumer’s idiosyncratic preferences – things like color which have no affect on the innate value of the item. Products are also selected as being one of two quality levels (vertical differentiation). Vertical differentiation is based on quality differences – all other things being equal a higher quality item, provides more utility to a customer. The model we use assumes that customers have homogeneous quality preferences – specifically, all customers recognize a quality premium associated with the high quality product. In this chapter, we define the specific model that is used and provide background information on already established properties of optimal solutions.
2.1 Model Description

A retailer wishes to maximize expected profit by choosing an assortment \((b, y)\). Let \(b\) be a vector representing the locations of chosen products along the horizontal attribute space \((b_j \in \mathbb{R})\) and let \(y\) be a vector representing the quality levels of the products in the vertical space \((y_j \in \{L, H\})\). \(n(b, y)\) gives the number of products in the assortment. For notation purposes, the products are numbered \(1, \ldots, n(b, y)\). The location of product \(j\) is given as \(b_j\) and \(b_j < b_{j+1}\). The distribution of customer preference along the horizontal attribute is given by \(F(z), z \in \mathcal{B} \subset \mathbb{R}\). Each product has a selling price, \(p_j \in \{p_L, p_H\}\), which depends upon the chosen quality level. The unit cost to the retailer of all products is given by \(c\) and the retailer must also pay a fixed cost, \(K\), for each type of product in the assortment. The intensity of demand is given as \(\lambda\). Each product has a base value to customers, \(v\), and high quality products offered have an additional premium value, \(q\). Customers pay a per-distance travel cost, \(t\), for purchasing a product which is at a location other than that customer’s preference.

We now define a number of terms which will be useful in further analysis of the problem. An illustration of these terms is provided in Figure 2.1.

**Base utility**, \(u_j(y_j)\): The base utility of a product is the utility gained by a customer when purchasing a product at his preferred location. A high quality product has a base utility of \(v + q\) and the low quality product has a base utility of \(v\). As such, \(u_j(y_j)\) is given by:

\[
u_j(y_j) = \begin{cases} 
v + q & \text{if } y_j = H \\
v & \text{Otherwise.}
\end{cases}
\]

**Purchasing Utility**, \(U_j(z, b_j, y_j)\): The utility gained by a customer with preference \(z\) from purchasing a product at location \(b_j\) with quality level \(y_j\) is given by \(U_j\). This utility is a surplus derived from subtracting the purchase price and travel cost from the base utility.
Purchasing utility is defined as:

\[ U_j(z, b_j, y_j) = u_j(y_j) - p_j - t|z - b_j|. \]

Each customer will purchase the product which gives him the highest purchasing utility, provided that utility is positive.

**Coverage distance**, \(l_j\): For ease of analysis it is useful to compute the distance \(|z - b_j|\) where \(U_j = 0\). This distance represents the maximum distance a customer’s preference can be from a product’s horizontal location where the customer receives nonnegative utility. This is known as coverage distance and it depends only on the product quality and is independent of the actual location of a product. \(l_j\) is given by:

\[ l_j = \frac{u_j(y_j) - p_j}{t}. \]

As a shorthand, we use the notation \(l_j \in \{l_L, l_H\}\) where \(l_L\) is the coverage distance for a
low quality product and $l_H$ is the coverage distance of a high quality product.

**First choice interval, $[b_j^-, b_j^+]$:** The first choice interval of a product $j$ gives the horizontal locations of every customer who will choose that product to purchase. The endpoints of the interval are given as:

$$ b_j^- = \min \{ z : U_j(z, b_j, u_j(y_j)) > U_i(z, b_i, u_i(y_i)) \ \forall \ i \neq j \} $$
$$ b_j^+ = \max \{ z : U_j(z, b_j, u_j(y_j)) > U_i(z, b_i, u_i(y_i)) \ \forall \ i \neq j \}. $$

This definition assumes unique products – there will never be two products of the same quality at the same location (this would be an irrational decision for obvious reasons). For ease of computation, the endpoints may also be given as:

$$ b_j^- = \max \left\{ b_j - l_j, \frac{(p_j - u(y_j)) - (p_{j-1} - u(y_{j-1})) + b_j t + b_{j-1} t}{2t} \right\} \quad (2.1) $$
$$ b_j^+ = \min \left\{ b_j + l_j, \frac{(p_{j+1} - u(y_{j+1})) - (p_j - u(y_j)) + b_j t + b_{j+1} t}{2t} \right\}. \quad (2.2) $$

This definition assumes no dominated products. A dominated product is a product which is not the first choice product for any customer. Mayorga [15] has shown dominated products are never optimal in the cases described by this model.

**First choice probability, $d_j(b, y)$:** From the first choice interval we can compute the probability that a given customer will choose a product $j$. This probability is given by:

$$ d_j(b, y) = \int_{b_j^-}^{b_j^+} f(z) dz = F(b_j^+) - F(b_j^-). \quad (2.3) $$

**Product profit, $\Pi_j(b, y)$:** The profit for an individual product $j$ in an assortment $(b, y)$ is given by:

$$ \Pi_j(b, y) = (p_j - c_j) \lambda d_j(b, y) - K. $$
The profit consists of the marginal revenue for a product, \( p_j - c_j \), multiplied by the expected proportion of customers who choose that product, \( \lambda d_j(b, y) \), minus the cost to the retailer of adding that product to the assortment, \( K \).

**Assortment profit, \( \Pi(b, y) \):** The profit for an assortment \((b, y)\) is given by:

\[
\Pi(b, y) = \sum_{j=1}^{n(b, y)} \Pi_j(b, y).
\]

**Objective function:** The objective of the retailer is then to:

\[
\max_{(b, y)} \Pi(b, y)
\]

subject to (2.1), (2.2) and (2.3).

### 2.2 Properties of the Optimal Solution

Our work follows the works of Gaur and Honhon [4] in locational choice with a single quality level and Mayorga [15] in locational choice with multiple quality levels and homogeneous quality preferences. For the unimodal and uniform cases, Gaur and Honhon reduce the single quality level problem to a maximization problem of a single variable over a bounded region. This is accomplished by showing:

[P2.1] There is a bounded region which contains the only locations where products can be profitable.

[P2.2] Coverage of this region is continuous – every location on the interval \([b_1^-, b_{n(b, y)}^+]\) is contained within the first choice interval of some product. Alternatively, \( b_j^+ \geq b_{j+1}^- \forall j = 1...n(b, y) - 1 \).

[P2.3] Coverage is non-overlapping – every customer sees positive utility for at most one product. Alternatively, \( b_j^+ \leq b_{j+1}^- \forall j = 1...n(b, y) - 1 \).
[P2.4] The location of the first product is no more than twice the coverage distance of the product beyond the lower bound of the bounded region (and the location of the last product is no more than twice the coverage distance of the product before the upper bound).

Without these properties, the problem is quite difficult as it contains many variables and non-linear terms. With these properties, the authors show that the once difficult problem can now be solved with a single variable line search. Once the location of the first product is found, the remaining assortment is implied by the properties above.

Mayorga adds a second quality level of product and reexamines the problem, again for the uniform and unimodal cases. She shows that [P2.1], [P2.2] and [P2.3] all hold from above. She also provides a range of parameters for which the two quality problem reduces to the single quality problem – if the optimal solution is known to consist of a single quality (all high or all low quality products), the optimal solution can be found using the methodology of Gaur and Honhon. However, for a large number of cases the optimal solution is unknown. These assortments may be of a single quality or a mixture of different qualities. Because of [P2.2] and [P2.3], the locations of all products can be found using the location of the first product. However, unlike Gaur and Honhon, we must now find a sequence of quality levels for the assortment. These two problems (the location problem and the sequence problem) are inherently linked together and must be solved simultaneously. This makes the multiple quality problem much more difficult than the single quality problem.

For the uniform case, McElreath et al. [18] show that the optimal assortment can still be found by efficiently enumerating the solution space. The problem is formulated as a modified knapsack problem which allows for a single partial product. This approach works because in the uniform case the demand associated with a full product is constant and independent of position. The solution space can be enumerated because many solutions are equivalent and we do not need to find a unique optimal solution. In the unimodal case, the demand
associated with a product is a function of its location. As each solution is unique, we need to find a different method for generating solutions.

In the next chapter, we present and examine heuristic methods for finding solutions to the assortment problem with multiple quality levels in the unimodal case. Specifically, we evaluate three metaheuristic techniques and examine their effectiveness in generating quality approximate solutions.
Chapter 3

A Metaheuristic Study for
Homogeneous Quality Preferences

With the unimodal case presenting a difficult problem to solve optimally, we employ a series of metaheuristic techniques to attempt to provide high quality solutions in a relatively small amount of time. Three methods are investigated: a Genetic Algorithm (GA), Simulated Annealing (SA), and a Tabu Search (TS). The goal of this exercise is not only to provide an efficient method of generating solutions to this class of problems, but also to study the effectiveness of each of these techniques for solving these types of problems. Since there is no prior knowledge of which type of approach will be best, we must try a range of different techniques. While the GA is a population-based metaheuristic, the TS and SA are path based metaheuristics. Finding the best method of these three will not only give us a solution method for the homogenous quality preference problem, but also can give insight back into the metaheuristic literature for which method is most appropriate for this combination continuous/combinatorial problem. We will begin by describing the implementation and design decisions for the study and for each of the metaheuristics.
3.1 Common Elements

The metaheuristic approaches share several common elements. These include the solution representation, the fitness function, and the upper bound on profit. Before defining these elements, we first present a number of bounds which will be useful for our approach:

**Bounds on product location:** Products should only be located so that they attain enough profit to cover the fixed cost, that is \( b_1 \geq \min\{b : (p_1 - c)\lambda d_1(b, y) \geq K\} \). This depends on the quality level of product 1, which is an unknown variable. Thus, the bounds on the location of products of type \( y \in \{L, H\} \) are given by

\[
\begin{align*}
\bar{b}_y &= \min\{b : (p_y - c)\lambda(F(b + l_y) - F(b - l_y)) = K\} \\
\tilde{b}_y &= \max\{b : (p_y - c)\lambda(F(b + l_y) - F(b - l_y)) = K\}.
\end{align*}
\]

It is useful to develop an absolute minimum location, before which no product can be located. We will call this location \( \bar{b} \) and it represents the first variety attribute (location) where we may place a product that will return a non-negative expected profit. Similarly, we can develop an absolute maximum location, \( \tilde{b} \), representing the last location where we can place a product that will return non-negative expected profit. These bounds are defined as:

\[
\begin{align*}
\bar{b} &= \min\{\bar{b}_L, \bar{b}_H\} \\
\tilde{b} &= \max\{\bar{b}_L, \tilde{b}_H\}
\end{align*}
\]

**Bound on the number of products:** Using the bounds on product location, we can develop upper bounds on the number of products of each type. Given the coverage distance of products of each quality level, and the minimum and maximum locations, we define \( \bar{n}_L \) and \( \bar{n}_H \) to be the upper bound on the number of low and high quality products in the
assortment where:

\[
\bar{n}_L = \left\lceil \frac{b_l - b_L}{2l_L} \right\rceil, \quad (3.3)
\]

\[
\bar{n}_H = \left\lceil \frac{b_H - b_H}{2l_H} \right\rceil. \quad (3.4)
\]

**Low and high quality regions:** From Mayorga [15], we know that when \( l_H \geq l_L \), the high quality product cannibalizes, thereby reducing the problem to a single variable line search. We restrict our attention to cases where \( l_L > l_H \) (note that the methodology could be slightly modified to cover all cases). In these cases the region over which high quality products are profitable is smaller than the region over which low quality products are profitable, since the high quality products have smaller coverage distance and therefore smaller resulting demand. There exists (at least) two locations in each distribution where the more profitable product will switch from low to high (or vice versa). We define these two locations, \( b_{LH} \) and \( b_{HL} \) below.

\[
b_{LH} = \min \{ b : (p_L - c_L)[F(b + l_L) - F(b - l_L)] = (p_H - c_H)[F(b + l_H) - F(b - l_H)] \},
\]

\[
b_{HL} = \max \{ b : (p_L - c_L)[F(b + l_L) - F(b - l_L)] = (p_H - c_H)[F(b + l_H) - F(b - l_H)] \}.
\]

It is also of use to define \( \tilde{d} \) and \( \hat{d} \) as densities given by:

\[
\tilde{d} = [F(\bar{b}) - F(b)], \quad \hat{d} = [F(b_{HL}) - F(b_{LH})].
\]

With these bounds defined, we present the common elements of the metaheuristic representations:

**Solution representation:** The solution representation is illustrated in Figure 3.1. Each representation consists of \( \bar{n}_H + \bar{n}_L \) elements and a real number on \([0, 1]\). This number, known as \( \Delta \), is used to compute \( b_1 = b + \Delta \).
Figure 3.1: Sample Solution Representation.

The remaining $n_H + n_L$ elements each represent a potential product in the solution, with the total number of elements equal to the sum of the upper bound on low quality products in the assortment, $n_L$, and the upper bound on high quality products in the assortment, $n_H$. Each element designates a product as high or low quality and contains a random number on the interval (0, 1) which serves as a random key for sorting purposes. Random keys have been used frequently in genetic algorithms and were first proposed by Bean in 1994 [2].

We will now discuss the decoding of the representation in the following, as it is inherently related to computing the objective function.

**Objective value:** Our goal to maximize expected profit. Each solution is decoded by a function which calculates the expected profit for the given solution, as follows:

1. The $n_H + n_L$ elements are sorted by the random key field (from smallest to largest).
   
   Let the $[i]^{th}$ element be the element with the $i^{th}$ smallest random key.

2. $Current\ Location = (b + \Delta - l_{[i]})$ and $Count = 0$. 


3. For $i = 1$ to $\bar{n}_H + \bar{n}_L$:

(a) Consider the $[i]^{th}$ element.

(b) Determine the quality level of the product and calculate $\Pi_{[i]}$, the expected profit for the product at $\text{Current Location} + l_{[i]}$.

(c) If $\Pi_{[i]} \geq 0$ the product is added to the assortment and the total expected profit is updated. $b_{\text{Count}} = \text{Current Location} + l_{[i]}$, $\text{Current Location} = \text{Current Location} + 2l_{[i]}$, and $\text{Count} = \text{Count} + 1$

(d) Else, If $\text{Count} = 0$, then: $\text{Current Location} = \text{Current Location} + 2l_{[i]}$.

**Upper bound:** As an optimal solution is not available in the general unimodal case, we employ the use of an upper bound on expected profit for evaluation of solution quality. To compute the upper bound, we divide the customer preference region into up to two areas: an area in which high quality products will be more profitable, and the area in which low quality products will be more profitable. We can then derive an upper bound on expected profit for each region and the sum will provide the upper bound for our problem. For each region, we assume that the entire density can be captured by placing products, so the gross revenue is equal to the density multiplied by the profit margin for the respective item. We pay the fixed cost for each full product that may be placed in that region. Because this method allows for partial products that will not result in a fixed cost, the bound will not be tight to the optimal solution. The bounds will also depend on the problem instance. Consider the following three cases:

**Case unmixed 1:** $b = b_H$. This implies that a high quality product is more profitable than a low quality product at the same location, anywhere in the feasible location region. Thus the optimal assortment will consist of only high quality products. An upper bound on profit is given as follows:

$$UB_1 = (p_H - c) \lambda \tilde{d} - \left[ \frac{b - b}{\sqrt{n}} \right] K.$$
Case unmixed 2: $b = b_L$ and $b_{LH} < b$. In this case the optimal assortment will consist of only low quality products. An upper bound is given by:

$$UB_2 = (p_L - c)\lambda\tilde{d} - \left\lfloor \frac{\bar{b} - b}{2L} \right\rfloor K.$$ 

Case mixed: otherwise. In this case the optimal assortment may contain both high and low quality products, but high quality products are only profitable in the region bounded by $[b_{LH}, b_{HL}]$. Thus the bound is given by:

$$UB_3 = (p_H - c)\lambda\hat{d} - \left\lfloor \frac{b_{HL} - b_{LH}}{2H} \right\rfloor K + (p_H - c)\lambda|d - \hat{d}| - \left(\left\lfloor \frac{b_{HL} - b_{LH}}{2L} \right\rfloor + \frac{\bar{b} - b_{HL}}{2L} \right) K.$$ 

3.2 Genetic Algorithms

Genetic Algorithms (GA) are population based evolutionary metaheuristics, whose use for combinatorial problems is described in Goldberg [6]. A population consists of chromosomes, each of which maps to a point in the solution space. We use the solution representation discussed in Section 4.1 for our chromosomes. Once an initial population is established, future generations are produced through the use of the following operators:

- **Elite reproduction**: Chromosomes may be passed untouched to the next generation through the use of elite reproduction. Typically, the “most fit” chromosomes (as judged by a fitness function) will be passed on, with the number determined by the implementation. In elite reproduction, the best X% of the current generation is copied, unaltered, to the next generation. The fixed percentage X is often set at 20%, as in Bean [2].

- **Descendants**: Descendant chromosomes are produced through the “mating” of two parent chromosomes. Our implementation uses a single point crossover operator to produce descendants. Single point crossover produces two offspring from two parents. One descendant will contain the genetic information from one parent up to a crossover
point and the genetic information from the other parent after the crossover point. The second descendant will contain the complementary genetic information from each of the parents. Both offspring are retained for the future generation. The crossover operation is illustrated in Figure 3.2.

- **Immigration**: Immigration is the process of introducing entirely new chromosomes to the population. In our case, these chromosomes are generated randomly.

![Figure 3.2: Illustration of the Crossover Operator.](image)

Our genetic algorithm contains a population of 100, which is initialized with a random starting population. Following populations are built using 20 percent elite reproduction, 70 percent crossover, and 10 percent immigration. Parents are chosen randomly from the entire previous population and reproduction is performed with a single point crossover with a random crossover point. A total of 1000 generations are evaluated. All parameters were chosen empirically based on a subset of initial test cases. For example, to enable a fair comparison between methods, we needed to establish a fixed number of solution evaluations over all three methods. To establish this limit we found the number of solutions needed to
converge over a subset of test cases for all thee methods and increased this number by some factor to err on the side of caution.

### 3.3 Simulated Annealing

Simulated annealing (SA) is a path-based metaheuristic first applied to optimization by Kirkpatrick in 1983 [8]. The method is intended to approximate the natural process in which metal is annealed, and employs the Metropolis algorithm. The metaheuristic begins with an initial solution and a “temperature” value. New solutions are generated by performing a small modification to the previous solution. New solutions that show an improvement in objective value are always accepted, and solutions which produce a inferior objective value are accepted with probability $e^{\Delta \varepsilon / T}$, where $\Delta \varepsilon$ is the change in objective value and $T$ is the current temperature of the system. The temperature is lowered once “stability” is reached at a temperature level (determined empirically) and the algorithm is finished after the solution becomes stable over multiple temperature levels. The cooling scheme used to lower the temperature is geometric, with the updated temperature set to 90% of the current temperature.

Our simulated annealing process begins with a random starting solution. The initial temperature is set according to the characteristics of the individual problem instance. New solutions are generated by either randomly modifying the offset (0.10 probability) or changing the quality level of a randomly selected product (0.90 probability). The offset modification is performed by generating a $(-1, 1)$ random number which is multiplied by $\sigma$ (the standard deviation of customer preference) and added to the previous offset. For comparison purposes to the other methods, a limit of 100,000 solutions was enforced.
3.4 Tabu Search

Tabu search (TS) was proposed and applied to optimization problems by Glover in 1986 [5]. Like simulated annealing, tabu search is a path based metaheuristic. An initial solution is generated, as well as a list of “neighbor” solutions. Neighbor solutions are defined as a group of solutions that border the original solution and usually involve a single perturbation of the original solution. A tabu list is established, which is a listing of recently visited solutions (the length of the list is determined by the implementer). For each “move” in the algorithm, the tabu search generates the list of neighbor solutions and moves to the neighbor with the most desirable objective value that is not present on the tabu list.

Our tabu search begins with a random starting solution. The tabu list consists of 10 elements, and any solution with the same objective function is considered identical. In other words, our tabu list consists of a set of objective values rather than a set of solutions. We do this for two reasons. First, because of the representation used, many solutions could yield the same objective function; second, our solution representation contains a continuous variable so that it is unlikely that the exact same solution be visited again; thus, a tabu list consisting of full solution representations would be very inefficient. The search consists of 10,000 updates, with \( \bar{n}_H + \bar{n}_L + 1 \) neighbors generated by randomly modifying the offset and changing the quality level of each product individually. The offset modification is performed by generating a \((-1, 1)\) random number which is multiplied by \(\sigma\) and added to the previous offset. Both tabu search and simulated annealing use the same perturbations. Simulated annealing allows only one solution to be considered per move, and tabu search considers the entire neighborhood of solutions and selects the best non-tabu move.

3.5 Computational Experiments

The three methods were each evaluated computationally over a series of 656 test cases. The cases are divided into three \( p_L \) values, which each contain two \( K \) values, and then a
range of $p_H$ and $q$ values are presented which result in $l_H$ values of interest (recall we are interested in cases such that $l_H < l_L$), all other parameters are fixed. For example, for $p_L=1.5$, parameters are such that $l_L=.25$; thus, we vary $p_H$ and $q$ such that $p_H - p_L < q$, resulting in $l_H < .25$. A detailed list of test cases is in Appendix A. An upper bound for each case was computed off-line. Each method was implemented in C++ and compiled with Microsoft Visual Studio 2008. 50 replications of each test case were evaluated using Condor, a high throughput grid computing solution. The Mersenne Twister pseudo-random number generator was used and each replication of each run used a pre-generated seed that ensured no overlapping of random number streams. Results were computed by measuring deviation from the bound for each replication, and averaging the results for each case over the 50 replications. Computational time for a single case on a P4 3.20 GHz PC with 2 GB of RAM was <1 second.

### 3.6 Results

The aggregated results are presented in Table 3.1. **Bold** entries show the best value in each row. Each row represents a single test case or an aggregation of a class of test cases. These results show similar performance by the GA and TS, and both are well ahead of SA. Each method shows low variance (on the order of $10^{-6}$) and as a result we are satisfied with the convergence of each method. Subdividing the tests cases gives the results as shown in Table 3.1: Aggregated Results

<table>
<thead>
<tr>
<th>% Deviation from Bound [Avg (Min, Max)]</th>
<th>GA</th>
<th>TS</th>
<th>SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>2.590%(0.018%, 26.070%)</td>
<td>2.629%(0.022%, 26.616%)</td>
<td>3.487%(0.135%, 26.160%)</td>
</tr>
<tr>
<td>Mixed</td>
<td>3.471%(0.018%, 26.070%)</td>
<td>3.510%(0.022%, 26.616%)</td>
<td>4.667%(0.197%, 26.160%)</td>
</tr>
<tr>
<td>Unmixed</td>
<td>1.099%(0.059%, 4.456%)</td>
<td><strong>1.040%(0.043%, 4.280%)</strong></td>
<td>1.357%(0.135%, 4.694%)</td>
</tr>
</tbody>
</table>

3.2. Here we can observe the relative strengths of both the GA and the TS. The GA offers increased performance in mixed cases (Cases 1 and 2) and generally in the class of problems
that we consider to be “hard” (the mixed cases are a member of this class). Harder problems
tend to occur in the situations where there is a small price premium or quality premium
between products. Hard problems are also those in which the K value is relatively high
(problems in which it is more difficult to obtain positive profit from each product). The
TS method, by contrast, excels at problems in which the resulting assortment is unmixed.
These are problems in which the assortment is relatively easy to find and the difficulty is
in finding the correct offset.

Box plots of the results are shown in Figure 3.3. Inspection reveals that these results

<table>
<thead>
<tr>
<th>Table 3.2: Case Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Deviation from Bound [Avg (Min, Max)]</td>
</tr>
<tr>
<td>( P_L )</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>1.5</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>1.0</td>
</tr>
<tr>
<td>Mixed</td>
</tr>
<tr>
<td>Unmixed</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>Mixed</td>
</tr>
<tr>
<td>Unmixed</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>Mixed</td>
</tr>
<tr>
<td>Unmixed</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Mixed</td>
</tr>
<tr>
<td>Unmixed</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>Mixed</td>
</tr>
<tr>
<td>Unmixed</td>
</tr>
</tbody>
</table>

are not normally distributed. Non-parametric statistical testing confirms the observational
results above. A statistical ranking of the three methods is presented in Table 3.3. Non-
parametric techniques are used, as the results are skewed towards the upper bound, thereby
eliminating the normality assumption. Friedman’s test revealed significant statistical dif-
fences between the methods, and then a multiple comparison test was used to rank the
methods for each case. We observe that the results in Tables 3.2 and 3.3 are consistent.
3.7 Conclusions

Simulated Annealing can be eliminated as an effective tool for solving our problem cases. SA performed at an inferior level in each of the cases tested, suggesting a poor fit for these problem cases and for the multiple quality assortment problem in general. We speculate that the performance of the SA suffers from only considering a single solution at each iteration, but we have not tested this specific assertion.

The genetic algorithm and tabu search perform better than simulated annealing in general in these problems, as demonstrated by our statistical analysis. Relative to the upper bound, GA and TS can perform within 0.018\% and 0.021\% respectively, on some problems. While the GA performs slightly better, averaging over all cases, statistical ranking shows that the difference is not significant. On the other hand, the TS does out-perform the GA in more case-by-case comparisons. Therefore, without pre-classifying the problem (to know if the assortment will be mixed or unmixed), both GA and TS methods are recommended for solving the assortment planning problem.
Previous work with these problems by Mayorga [15] has suggested that they may be pre-classified by instance type and then solved. For example, as shown in Figure 3.4, problems in the gray region (such as point (a)) will contain a single quality type (unmixed case), while problems in the white region (such as point (b)) may contain multiple quality types (mixed case). The unmixed case can be solved using several single variable line searches.

The computational work necessary to pre-classify problems (as mixed or unmixed) is non-trivial. Fortunately, the use of these metaheuristic methods to solve assortment problems eliminates the need to pre-classify the problems. Thus, an interesting side effect of our work shows that since these methods work so quickly over all problem types, there is no need to pre-classify problems when the only end goal is to find a solution to the problem. In this case both GA and TS methods are highly competitive.
For mixed problem instances, the Genetic Algorithm is the preferred method for solving assortment problems. The GA offers increased performance for these “hard” problems. On the other hand, for the unmixed cases, the TS method was the dominant method. This advantage seems to stem from the TS method’s ability to modulate an existing offset, as opposed to generating entirely new random offsets, as the GA does. Also, TS considers changing the quality level of any product and only changes the offset when it will provide the largest gain in the objective value. It seems reasonable to suggest that adding an offset mutation to the GA method would allow for better performance against the TS in these cases.

Additionally, it is worth making a note on the quality of the bounds used in this evaluation. The bounds, by nature, will underpay the fixed cost of the generated assortment. As such, the gap between the optimal solution and the bound on profit grows greatly with $K$, and becomes even larger as the marginal profit to fixed cost ratio grows. This relationship is illustrated in our results and should be considered more of a statement on the quality of the bound than a statement on the quality of the results. However, no tighter bound is known at this time for these cases.

We have shown that metaheuristic techniques can be efficiently and effectively used to approximate solutions to difficult assortment planning problems. Thus researchers may look to such methods as more pragmatic retail models are developed. One possible extension is generalizing the assumption of consumer preferences beyond the unimodal distribution. For example, if consumer preferences are bi-modal the optimal assortment may contain overlapping products, thus the location of each product needs to be determined and the solution representation would have to be altered accordingly. Another possible extension is to consider the dynamic substitution environment, where consumers will attempt to substitute for their most preferred product if it is out of stock. In this case the assortment and inventory problem do not decouple, making the problem analytically intractable. Not only
is a new solution representation necessary, but also a more complex objective function.

Having established the viability of metaheuristics as a solution approach, we now consider a different approach. Metaheuristics can produce quality solutions quickly but are inherently random in nature. There is no guarantee of a good solution on a given run and they often need to be run multiple times which takes away from their quick nature. As previously mentioned the uniform case has been solved to optimality by using a modified knapsack formulation. In the next chapter we use a similar approach to find an optimal solution using a dynamic program combined with a line search.
Figure 3.4: Pre-classified problem instances
Chapter 4

Assortment Planning with Multiple Quality Levels: A Dynamic Programming Approach

The metaheuristic approach to the assortment planning problem with multiple quality levels yielded relatively good approximate solutions in a small amount of time. However, there may be another approach which can provide optimal solutions by using a deterministic algorithm. In this chapter, we draw parallels between the assortment planning problem with multiple quality levels and the knapsack problem and then exploit that relationship by developing a combined dynamic programming/line search approach to find an optimal solution.

4.1 Dynamic Program

The assortment planning problem is related to the classical knapsack problem in that the firm must choose a subset of items (products) to carry in order to maximize total profit and the total number of items which can be carried is constrained. Each item has a weight (first
choice interval) and a value (profit). The difference between the classical knapsack and the assortment planning problem is that the size of our knapsack is unknown ahead of time and is embedded into the optimization problem. The range of the size of our knapsack is instead constrained by the maximum and minimum locations on the attribute space for which a product may be placed and produce a non-negative profit. Furthermore, when customer preference is unimodal, the classical knapsack formulation is insufficient as the value of each item is determined by its location in the assortment. For the uniform case, the profit for each item is proportional to its first choice interval and is constant for products which are contained within the bounds of the distribution. McElreath, Mayorga, and Kurz [18] provide optimal solutions for the uniform case by allowing for a single partial product with proportional demand. The unimodal case is more complex, as the demand for a product is directly related to its position. To solve this problem we formulate a dynamic program approach inspired by the knapsack problem which is then combined with a line search to find an optimal solution.

While McElreath et al. [18] formulate the uniform case as an integer program, it could alternatively be formulated as a dynamic program. The optimal location of the first product is fixed as a function of the quality level of that product and products are continually added until a truncated product is added, at which point the assortment is finished. At each stage of the problem there are only four possible actions: add a high or low quality product or add a truncated high or low quality product. The state needs to consider only the horizontal location of the last product added to the assortment. This simple formulation produces an optimal assortment very quickly. The unimodal case, however, requires additional complexity to find a solution. The optimal location of the first product is a function of both the quality level of that product and the quality levels of the remaining assortment. Partial products may no longer exist in the assortment because the distribution may not be truncated and they may not be optimal even if they do exist. We define a dynamic programming formulation that, when paired with a line search, will find the optimal solution.
to the general unimodal case.

There are several properties of a problem that can be determined \textit{a priori} that are useful to a dynamic programming approach. Recall from Chapter 3 that we may bound possible positions of products in the assortment. For each type of product (L and H) there must be a minimum and maximum position at which that type of product may be placed and still produce non-negative profit. These positions are given as \((b_L, \bar{b}_L), (b_H, \bar{b}_H)\) and are defined as in equations (3.1) and (3.2). We then have the absolute minimum and maximum positions, \(\underline{b} = \min\{b_L, b_H\}\) and \(\bar{b} = \max\{b_L, b_H\}\). If the mode of \(F(z)\) is given by \(m\), then \(b < m\) and \(\bar{b} > m\). Additionally, it follows that as each product has some first choice interval and, as there is no substitution between products, that there is some maximum number of products that can be placed in these bounded intervals. Let \(\bar{n}\) be the maximum total number of products in the assortment and \(\bar{n}_L\) and \(\bar{n}_H\) be the maximum number of low and high quality products respectively. We will assume that for each quality level of product there is at least one location where a product may be placed to receive positive profit (otherwise the assortment is empty or problem reduces to the single quality problem).

When \(l_L > l_H\), these bounds are given as in equations (3.3) and (3.4). \(\bar{n}\) is then given as:

\[
\bar{n} = \bar{n}_H + \left\lfloor \frac{(b_L - \bar{b}_H) + (b_H - b_L)}{2l_L} \right\rfloor.
\]

For cases where \(l_H \geq l_L\), these bounds may be found in a similar manner.

Assume \(F(z)\) is unimodal with mode \(m\). For a given assortment, \(b_1^-\) represents the lower endpoint of the first choice interval of the first product, i.e. where coverage for the first product in the assortment begins. The dynamic program (DP) finds an assortment for a given \(b_1^-\) that maximizes profit by determining the number of products to be included in the assortment, \(n \leq \bar{n}\), and the quality level of each product, \(y = (y_1, y_2, \ldots, y_{\bar{n}})\).
The stage of the DP represents the current product being evaluated. Let \( k \) represent the current stage of the DP. In this case we have \( k = 1, 2, \ldots, \tilde{n} \) (\( \tilde{n} \) total stages). For each stage, we must determine an action, \( a_k \). The action for each stage is the decision of what quality level of product will be added to the sequence. Because \( \tilde{n} \) is an upper bound and may be greater than the number of products in the assortment, we have three possible actions: L (add a low quality product to the assortment), H (add a high quality product) or E (add an “empty” product, i.e. do not add a product at this stage):

\[
a_k \in A = \{L, H, E\}.
\]

Suppose first that the sequence of products is not restricted in any way, that is \( a_k \in A \forall k \). In this case it sufficient to let the state for the DP, \( s_k \), represent \( b_k^+ \), which is the location of the beginning of the first choice interval for product \( k \). Note that it is trivial to transform this state to \( b_k^- \) however \( b_k^- \) provides a simpler state transition function, \( g(s_k, a_k) \). This function is given as

\[
s_{k+1} = g(s_k, a_k) = \begin{cases} 
  s_k + 2l_L & \text{for } a_k = L \\
  s_k + 2l_H & \text{for } a_k = H \\
  s_k & \text{for } a_k = E.
\end{cases}
\]

The optimal profit with \( \tilde{n} - k \) stages to go is a function of the action, \( a_k \), and the current state, \( s_k \). This function, \( J_k(s_k) \), is given as

\[
J_k(s_k) = \max_{a_k \in A} \begin{cases} 
  (p_L - c)\lambda[F(b_k^- + 2l_L) - F(b_k^-)] - K + J_{k+1}(s_{k+1}) & \text{for } a_k = L \\
  (p_H - c)\lambda[F(b_k^- + 2l_H) - F(b_k^-)] - K + J_{k+1}(s_{k+1}) & \text{for } a_k = H \\
  0 + J_{k+1}(s_{k+1}) & \text{for } a_k = E
\end{cases}
\]

with terminal condition \( J_{\tilde{n}}(s) = 0 \) for all \( s \). The optimal solution is given by \( J_1(s_1) \), where \( s_1 = b_1^- \).
While the general formulation above will yield a solution, it is computationally expensive and inefficient since it treats each product independently. To improve the formulation, we can integrate three rules regarding the relationship of neighboring products into the formulation to reduce the solution space. These rules are dependant on a normal distribution due to R4.2, but may be extended to the general unimodal case.

**R4.1.** Each type of product will not occur before its minimum or after its maximum possible position. This rule follows from definition of $b_L$ and $b_H$.

**R4.2.** Low quality products will not occur between high quality products. This rule follows from proposition 4.1.2.

**R4.3.** Empty products will never occur between two non-empty products. As empty products do not require space, this rule prevents redundant solutions.

In order to use rule R4.2, we must establish that high quality products will always be “sandwiched” between low quality products in a mixed assortment. In other words, we must show that it will never be optimal to have a low quality product located between two high quality products. In order to show this, we must first demonstrate that when a high quality product is located next to a low quality product and the low quality product is closer to the mode then it will be advantageous to modify the assortment so that the high quality product is closer to the mode.

**Lemma 4.1.1.** Consider an assortment $(b, y)$ with two products $i$ and $j$ where $y_i = H$, $y_j = L$, $i = j + 1$, and $b_j \leq m$. If $|b_j^+ - m| < |b_i^- - m|$, then there exists an alternate assortment, $(\hat{b}, \hat{y})$, with $\Pi(\hat{b}, \hat{y}) \geq \Pi(b, y)$. That is, if the low quality product is closer to the mode than the high quality product, an alternate assortment must exist.

**Proof.** The assortment $(\hat{b}, \hat{y})$ is identical to $(b, y)$ for all products other than $i$ and $j$. Let $\hat{y}_i = L$ and $\hat{y}_j = H$. Locate products $i$ and $j$ in $(\hat{b}, \hat{y})$ such that $\hat{b}_j^+ = b_j^+$ and $\hat{b}_i^- = b_i^-$. 

36
Note that:

\[ \hat{b}_j^+ = b_j^+ \]
\[ \hat{b}_j^- = \hat{b}_i^+ = b_j^+ - 2l_H \]
\[ b_i^+ = b_j^+ - 2l_L \]
\[ b_i^- = \hat{b}_i^- = b_j^+ - (2l_L + 2l_H) \].

The profits for all products other than \( i \) and \( j \) remains unchanged. The profits for products \( i \) and \( j \) in \((b, y)\) are given by:

\[
\Pi_i(b, y) = (p_H - c)\lambda[F(b_j^+ - 2l_L) - F(b_j^+ - (2l_L + 2l_H))] - K
\]
\[
\Pi_j(b, y) = (p_L - c)\lambda[F(b_j^+ - 2l_L)] - K
\]
\[
= (p_L - c)\lambda[F(b_j^+ - 2l_H) + F(b_j^+ - 2l_H) - F(b_j^+ - 2l_L)] - K.
\]

The profits for products \( i \) and \( j \) in \((\hat{b}, \hat{y})\) are given by:

\[
\Pi_i(\hat{b}, \hat{y}) = (p_H - c)\lambda[F(b_j^+ - 2l_L)] - K
\]
\[
\Pi_j(\hat{b}, \hat{y}) = (p_L - c)\lambda[F(b_j^+ - 2l_H) - F(b_j^+ - 2l_H) + F(b_j^+ - 2l_L) + F(b_j^+ - 2l_L) - F(b_j^+ - 2l_H)] - K.
\]

Subtracting \( \Pi(b, y) \) from \( \Pi(\hat{b}, \hat{y}) \) and removing like terms yields:

\[
\Pi(\hat{b}, \hat{y}) - \Pi(b, y) = (p_H - p_L)\lambda[F(b_j^+ - 2l_L) - F(b_j^+ - (2l_L + 2l_H))] - \]
\[
(p_H - p_L)\lambda[F(b_j^+ - 2l_L) - F(b_j^+ - (2l_L + 2l_H))]
\]
Note that:

\[ b_j^+ - (b_j^- - 2l_H) = (b_j^+ - 2l_L) - (b_j^- - (2l_L + 2l_H)) = 2l_H. \]

However, recall by assumption that \(|b_j^+ - m| < |b_i^- - m|\) and observe that \(b_j^+ > b_j^- - 2l_H > b_j^- - 2l_L > b_j^- - (2l_L + 2l_H)\). As \(p_H > p_L\), it follows that \(\Pi(\hat{b}, \hat{y}) \geq \Pi(b, y)\).

Using this lemma, we can then formalize the argument:

**Proposition 4.1.2.** If \(F(z)\) is normally distributed, then any assortment consisting of both high and low quality products will contain the high quality products “sandwiched” between the low quality products. That is, coverage for the high quality products is continuous and uninterrupted.

**Proof.** Lemma 4.1.1 shows that adjacent low and high quality products located to the left of the mode can be swapped if the result moves the endpoint of the high quality product’s FCI closer to the mode. The lemma also implies a second case (omitted here) where low and high quality products located to the right of the mode can be swapped in this manner if the result moves start of the high quality product’s FCI closer to the mode. Any assortment which violates this Proposition can be modified by applying this lemma recursively to both sides of the assortment. The result is an assortment for which all high quality products are adjacent and “sandwiched” between the low quality products.

To accommodate rules **R4.1** and **R4.3** we must redefine our state to include information about the last product placed in the sequence. We will redefine our state as \(s_k = (\rho_k, s_k)\) where \(\rho_k\) represents the product quality from the previous stage \((\rho_k = a_{k-1} \in \{L, H, E\})\) and \(s_k = b_k^-\).

From using all three rules, **R4.1–R4.3**, we can limit the available actions based on the
current state. Now \( a_k \in A(s_k) \) where

\[
A(s_k) = \begin{cases} 
E & \text{for } \rho_k = E \\
L, E & \text{for } b_k^- < b_H - l_H \\
H & \text{for } \rho_k = H, b_k^- < m - l_H \\
L & \text{for } \rho_k = L, b_k^- > m - l_L \\
L, H, E & \text{otherwise.}
\end{cases}
\]

The state transition function is given by:

\[
s_{k+1} = g(s_k, a_k) = \begin{cases} 
(L, s_k + 2l_L) & \text{for } a_k = L \\
(H, s_k + 2l_H) & \text{for } a_k = H \\
(E, s_k) & \text{for } a_k = E.
\end{cases}
\]

The optimal profit is then given by

\[
J_k(s_k) = \max_{a_k \in A(s_k)} \begin{cases} 
(p_L - c)\lambda[F(b_k^- + 2l_L) - F(b_k^-)] - K + J_{k+1}(s_{k+1}) & \text{for } a_k = L \\
(p_H - c)\lambda[F(b_k^- + 2l_H) - F(b_k^-)] - K + J_{k+1}(s_{k+1}) & \text{for } a_k = H \\
0 + J_{k+1}(s_{k+1}) & \text{for } a_k = E.
\end{cases}
\]

The DP produces an optimal assortment given some initial \( b_{-1}^- \). Let \( J(b) = J_1(b) \) represent the optimal assortment where \( b = b_1^- \). Then:

\[
J(b) = \max_y \sum_{j=1}^{\tilde{n}} \Pi_j(d_j) \quad \text{s.t.} \quad b = b_1^- , (2.1), (2.2)\ and\ (2.3)
\]

where \( d_j = 0 \ for \ y_j = E \).

To find the global optimal solution, we must search for the \( b \) which will maximize \( J(b) \).
Unfortunately, $J(b)$ is often non-concave and unpredictable as shown in Figure 4.1a. This example is produced using the parameters:

\[
\begin{align*}
    p_H &= 1.7525 \\
    p_L &= 1.5 \\
    c &= 0.5 \\
    K &= 0.025 \\
    \lambda &= 50 \\
    F(z) &\sim N(0, 1) \\
    l_L &= 0.1 \\
    l_H &= 0.0515.
\end{align*}
\]

Contrast this with a single quality assortment as in Figure 4.1b which can be solved as in Gaur and Honhon [4]. This example is produced using the parameters:

\[
\begin{align*}
    p_H &= 1.5050 \\
    p_L &= 1.5 \\
    c &= 0.5 \\
    K &= 0.2 \\
    \lambda &= 50 \\
    F(z) &\sim N(0, 1) \\
    l_L &= 0.1 \\
    l_H &= 0.1090.
\end{align*}
\]

Whereas in the single quality assortment we will only have to search over a maximum of two intervals, in the mixed quality assortment the number of intervals is unknown and will change depending on the parameters. Next we will describe a line search method that provides one option for finding the optimal value of $b$.

### 4.2 Line Search

Finding the position of the first product using the dynamic program is intractable. We are searching over a continuous line segment which produces an infinite solution space. Recall that we have established a lower bound on the position of the first product. From Mayorga [15], we know that in the optimal location of the first product must lie within $2l_L$ of this minimum (assuming $l_L > l_H$, $2l_H$ otherwise). Given that we have a bound on the location of the first product as well as a method of evaluating the objective function for each position, it then becomes possible to find the global optimum through a line search.

Let $J(b)$ be the function describing the optimal solution to the problem given a starting
Figure 4.1: Two Examples of the $J(b)$ Function
point $b = b_1^-$. The optimal solution could be found with using

$$\max_{b \leq b \leq b + 2L} J(b).$$

Unfortunately, we run into two problems. First, we don’t have a closed form representation of $J(b)$ for which we can maximize analytically. However, we do have the capability (through the dynamic program) of evaluating $J(b)$ numerically. The second problem is that in many (if not most) cases $J(b)$ is nonconcave. This proves to be a hindrance in that most numerical line search techniques rely on convexity (at least pseudo- or quasi-convexity).

Recall the numerical $J(b)$ example from Figure 4.1a. This is similar to what was observed by Gaur and Honhon [4], with the difference being that we have multiple inflection points in the multiple quality problem. While it is possible to calculate the number of potential inflection points for a particular problem instance, there is no known algorithm for providing this information for the general case. If we had a way to find these points, we would theoretically be able to perform multiple line searches over each interval and find the optimal solution.

Because we can only evaluate $J(b)$ directly we have limited choices for finding an optimal solution. Kolda, Lewis and Torczon [10] provide an overview of modern direct search methods which may be employed. Our problem differs from many direct search problems in that we do not have the benefit of unimodality in our function. Bazaar, Sherali, and Shetty [1] provide several methods for unconstrained optimization of a single variable. To solve our problem we propose an algorithm reminiscent of a uniform search – we uniformly sample from shortening line segments iteratively until we converge on a solution. By providing a high resolution to the sampling, it is unlikely that the algorithm will be stuck in a local optimum. The algorithm is as follows:
Summary of the search method

Initialization step: Choose a search resolution, \( r \), and the number of iterations \( s \). The interval of uncertainty (the interval which contains the optimal location), \([I^-, I^+]\), begins as \([b, b + 2l_L]\).

Main Steps

1. Divide the interval of uncertainty into equal segments using \( r \) points and evaluate \( J \) at each point. [For \( i = 0, ..., r - 1 \) evaluate \( J(I^- + i(I^+-I^-)/(r-1)) \)].

2. Of the \( r \) points, find the one that maximizes \( J \) and choose its neighbors as the new bounds for the interval of uncertainty.

3. Repeat for \( s \) iterations. On the final iteration, keep the \( r \) which maximizes \( J \) as the best estimate of the optimal \( b^* = b_1^- \).

As an example, consider the following problem:

\[
p_H = 1.7 \quad p_L = 1.5 \quad c = 0.5 \quad K = 0.2
\]

\[
\lambda = 50 \quad F(z) \sim N(0,1) \quad l_L = 0.125 \quad l_H = 0.1.
\]

Let \( r = 5 \) and \( s = 3 \) (while these values are small for normal use, they will be used for illustrative purposes). \( b = -2.6667945 \) and \( b + 2l_L = -2.4167945 \). Table 4.1 shows the evaluations for each of the three iterations.

For each iteration, the bolded row is the highest objective value and the boxed rows give the endpoints of the new interval of uncertainty. In this example, \( b = -2.4949195 \) is our best estimate of the optimal location with objective value \( J(b) = 54.254772 \).
4.3 Computational Studies

To test the performance of the above method, two computational studies were performed. First, we compare this method against the previously described metaheuristic methods (Chapter 3) to evaluate the performance of both methods. Second, we run experiments using varying parameters to explore the properties of the optimal solution. All experiments were implemented in Java 6 and run on a Intel Core 2 PC with 4GB of RAM. Line searches were performed using parameters \( r = 100 \) and \( s = 10 \).

### 4.3.1 Metaheuristic Comparisons

The metaheuristic study in Chapter 3 used a data set of 656 test cases to evaluate performance. Using the same test cases, we ran the combined approach and compared the results to the best solution generated on each case by the three metaheuristic methods. In each case, the combined approach produced a solution that was better than or equal to the solution produced by the best metaheuristic result (although even the largest difference is so small it is negligible). We can take two conclusions from this result: first, the combined approach performs at least as well as the metaheuristic for any given test case. This is not unexpected, as the dynamic program was designed to give an optimal result. Second, these results give further confirmation to the performance of the metaheuristic techniques. The metaheuristic methods are capable of producing a near optimal result even in cases when the result is not tight to the upper bound.

<table>
<thead>
<tr>
<th></th>
<th>( s = 1 )</th>
<th>( s = 2 )</th>
<th>( s = 3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( r ) ( b ) ( J(b) )</td>
<td>( b ) ( J(b) )</td>
<td>( b ) ( J(b) )</td>
</tr>
<tr>
<td>1</td>
<td>-2.6667945 54.236603</td>
<td>-2.5417945 54.250661</td>
<td>-2.5417945 54.250661</td>
</tr>
<tr>
<td>2</td>
<td>-2.6042945 54.243344</td>
<td>-2.5105445 54.254547</td>
<td>-2.5261695 54.253039</td>
</tr>
<tr>
<td>3</td>
<td>-2.5417945 54.250661</td>
<td>-2.4792945 54.253712</td>
<td>-2.5105445 54.254547</td>
</tr>
<tr>
<td>4</td>
<td>-2.4792945 54.253712</td>
<td>-2.4480445 54.247737</td>
<td>-2.4949195 54.254772</td>
</tr>
<tr>
<td>5</td>
<td>-2.4167945 54.236603</td>
<td>-2.4167945 54.236603</td>
<td>-2.4792945 54.253712</td>
</tr>
</tbody>
</table>

Table 4.1: Three Iterations of Search Example
From a practical viewpoint we must now address the issue of which technique is preferred for assortment problems. We believe the combined approach holds two key advantages. First, the dynamic program is a deterministic method. It requires only one replication to achieve a solution and is not sensitive to parameter tuning or random number generation. This leads to the second point in that the combined approach is faster in many cases than the metaheuristic. While a single replication is similar (in computational time) between both methods (a small number of seconds on a modern processor), the metaheuristics should be run for multiple replications due to their stochastic natures. The exception is cases which contain a large number of potential products. In cases where $\tilde{n} > 30$ the metaheuristic methods may be faster (and the advantage grows with size), but this threshold is larger than most practical cases. These are generally cases in which the carrying cost ($K$) is very small in comparison to the intensity of demand ($\lambda$) and products are still profitable very far from the mode ($m$). These cases are often not representative of practical problems and are not very sensitive to suboptimal solutions. This leads us to suggest that the combined approach is now the preferred method for solving most assortment problems.

Note that we have not addressed the performance of the dynamic program against an upper bound on profit. Because the results were so tight to the results of the metaheuristics, the discussion and evaluation of upper bounds still holds from Chapter 3. Our results confirm the hypothesis that the upper bound is very poor for cases with a large $K$ and the objective gap with the upper bound does not imply a bad solution.

### 4.3.2 Insights to the Optimal Solution

Mayorga [15] describes a method for classifying solutions as being composed of all high quality products, all low quality products, or unknown. This can be thought of as “mapping” the optimal solution space for a set of problems in which $q$ and $p_H - p_L$ are varied and all
other parameters are fixed. This unknown area may be either all high, all low, or mixed. With the ability to generate solutions, we can then further explore this unknown region. An example of this region is shown in Figure 4.2.

In this example, we evaluate problems at 100 values each of $q$ and $p_H - p_L$ (10,000 total evaluations). The resulting points which contain mixed quality levels are marked on the figure. Mayorga states that all problems above the solid line are all high and below the dotted line are all low with the unknown section between. From our example we see that

Figure 4.2: An Example of Solution Mapping
these properties hold. However, we can also see that the actual number of mixed quality assortments is much lower than the unknown region would suggest. Further, we can see that the mixed quality assortments all occur in the vicinity of the solid line. This example as well as other similar results allow us to infer that mixed quality assortments will only occur under a very specific set of circumstances and are more unlikely than assortments of homogeneous quality (in the example above only 11% of solutions are mixed assortments). This is an important result for assortment planning in practice because solutions for single quality assortments are easier to find and implement, although the exact criteria for a mixed quality assortment is still unknown.

To further explore properties of the optimal solution, we conducted a sensitivity analysis to observe changes in the optimal solution and objective value. Starting with a standard normal distribution, $F \sim N(0, 1)$, we varied $\sigma, p_H - p_L, q$ and $K$ and recorded the resulting optimal assortment and objective value. A full factorial experiment was conducted using the following parameters:

- **Fixed:** $p_L = 1.5, c = 0.5, \lambda = 2, v = 2, t = 5, \mu = 0$
- **Variable:** $p_H \in \{1.6, 1.7, \ldots, 2.4, 2.5\}, K \in \{0.085, 0.09, 0.095, 0.1, 0.105\}$,
  
  $q \in \{0, 0.1(p_H - p_L), \ldots, 0.9(p_H - p_L), (p_H - p_L)\}, \sigma \in \{0.5, 0.75, 1.0, 1.25, 1.5\}$.

We were then able to consider the following scenarios:

▷ *How does the optimal solution change in response to changes in the diversity of customer preference?*

Through marketing studies or other mechanisms, retailers may find that the diversity of customer preference in their customers is changing. In our model this is represented by a change in $\sigma$, the standard deviation of our distribution. The first concern to the retailer is
expected profit. As \( \sigma \) grows, the expected profit to the retailer decreases in a nearly-linear fashion. This is expected, as when \( \sigma \) grows we require more products to capture the same density and thus must pay more \( K \) costs. An example of this trend is illustrated in Figure 4.3.

![Figure 4.3: \( \Pi \) as a Function of \( \sigma \) and \( K \)](image1)

The next concern to retailers in this scenario is how their optimal assortment will change. First we examine the total number of products in the assortment. Figure 4.4 shows the total number of products in the optimal assortment for varying \( \sigma \) and \( K \) values. Notice the concave shape of the trend and the sensitivity of the drop in the tail to \( K \). Retailers should expand their assortment with additional customer diversity but must be aware of the tipping point where there is no longer enough aggregate demand to support the “fringe” products – especially when that retailer is paying a high price to include those products.

![Figure 4.4: \( n \) as a Function of \( \sigma \) and \( K \)](image2)
The makeup of the assortment is also important. Figures 4.5 and 4.6 show the optimal number of high and low quality products in the assortments respectively. Note that the number of high quality products remains relatively unchanged over $\sigma$, while the low quality trend again has a concave shape. This is similar to the earlier observation regarding the dropping of the “fringe” products in the assortment.

![Figure 4.5: $n_H$ as a Function of $\sigma$ and $K$](image)

![Figure 4.6: $n_L$ as a Function of $\sigma$ and $K$](image)

How much optimal profit deviation results from implementing a policy of only carrying single quality assortments?

Because of the ease in finding an optimal solution (and implementing the methods to find that solution) a retailer may wonder what the potential loss is in using only single quality assortments. We have established above that in most cases a single quality solution will be
optimal. However, in the cases where a mixed quality assortment is optimal, what is the potential lost profit to the retailer? In the example above, we isolated each case where a mixed quality assortment was found to be optimal. For each of these cases, we found the optimal solution when the model is constrained to contain either high or low quality products exclusively. First, we can examine the average objective value deviation using three methods: all high, all low, or the best of all high and all low. These results are shown in Table 4.2.

<table>
<thead>
<tr>
<th></th>
<th>All High</th>
<th>All Low</th>
<th>Maximum of All High and All Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>-5.04%</td>
<td>-11.00%</td>
<td>-2.43%</td>
</tr>
<tr>
<td>Median</td>
<td>-2.73%</td>
<td>-7.36%</td>
<td>-1.59%</td>
</tr>
<tr>
<td>Worst Case</td>
<td>-30.97%</td>
<td>-41.54%</td>
<td>-12.63%</td>
</tr>
<tr>
<td>Best Case</td>
<td>-0.00%</td>
<td>-0.02%</td>
<td>-0.00%</td>
</tr>
</tbody>
</table>

Table 4.2: Percentage Profit Deviation (rounded to nearest hundredth percent)

The results suggest that if one must choose a single quality level, with no other information, it would be best to choose the optimal all high quality assortment with a median objective value deviation from optimality of $-2.73\%$. A better alternative is to find the optimal low and high quality assortments and choose the one which has the highest objective value. This method provides a median deviation of $-1.59\%$ which may be an acceptable solution to some retailers. However, because the potential deviation in the worst case scenario can be as high as 41.54\%, our recommendation would be to implement the combined method for finding the optimal solution. The worst case scenarios for the all high assortment will occur when $l_H$ is small and the worst case scenarios for the all low assortment will occur when $\sigma$ is small. The best of high and low method performs similarly to the all high assortment, as in most cases the all high quality assortment is the one that is chosen. The high quality assortment was chosen in over 66\% of the cases. The actual mechanism for choosing between the all high and all low assortments solely from the parameters is unknown.

With a solution methodology to the homogenous quality preference case, we then look to
increase the utility of the model by relaxing assumptions. Specifically, we look to relax the assumption that each customer will always see the benefit of the high quality product. In the next chapter, we propose an alternate model removing this assumption, with the goal to produce a model that more accurately simulates the behavior of customers for some product types.
Chapter 5

Assortment Planning with
Heterogeneous Customer Quality Preferences

Previous work has used the assumption that, with all other factors being equal, all customers prefer a high quality product which meets their attribute preference. We propose that this assumption is quite limited in practice, and that many customers make purchasing decisions based purely on price, i.e. they have no want or need for the added features or benefits of the high quality item. As an example, consider the case of a company who is selling portable mp3 players. The players may be offered in multiple colors – black, white, red, etc. This is a horizontal attribute where the utility is derived solely on the customer’s preference. Suppose this player is also offered with an optional wi-fi/internet browser combination which carries some premium price above that of the base player. This is vertical differentiation with a low quality (no internet) and high quality item (wi-fi and browser). Previous assortment planning models assume that all customers will prefer the high quality item to the low quality item (if the price is not too high). In reality (and assuming appropriate pricing) we would find that some customers would enjoy the internet
capability and be willing to pay extra to receive it. However, some customers would likely not see the need for this feature (perhaps a customer uses the player only while exercising). To this customer the high quality product is actually less appealing (because of the higher price). Note that this customer may still buy the higher quality product if he is given no other option from the retailer and he still gains positive utility after accounting for the price.

Let us refer to the previous models as homogenous quality preference models. In the following section we introduce a heterogeneous quality preference model which attempts to relax the problematic assumption by allowing for two types of customers. The first type of customer is known as a “quality seeker” and his behavior is the same as the homogeneous quality preference models. A second type of customer, known as a “price seeker”, is introduced. This customer does not derive the extra benefit of the high quality item. We believe that the heterogeneous quality preference model provides new insight into assortment planning by creating a more realistic mechanism for customer product selection.

5.1 Model

In the homogeneous quality preference model a retailer wishes to maximize expected profit by choosing an assortment \((b, y)\). Let \(b\) be a vector representing the locations of chosen products along the horizontal attribute space \((b_j \in \mathbb{R})\) and let \(y\) be a vector representing the quality levels of the products in the vertical space \((y_j \in \{L, H\})\). \(n(b, y)\) gives the number of products in the assortment. The distribution of customer preference along the horizontal attribute is given by \(F(z), z \in B \subset \mathcal{R}\). Each product has a selling price, \(p_j \in \{p_L, p_H\}\), which depends upon the chosen quality level. The unit cost to the retailer of all products is given by \(c\) and the retailer must also pay a fixed cost, \(K\), for each type of product in the assortment. The intensity of demand is given as \(\lambda\). Each product has a base value to customers, \(v\). High quality products offered have an additional premium value, \(q\). Customers
pay a per distance travel cost, $t$, for purchasing a product which is at a location other than that customer’s preference.

For the heterogeneous quality preference model, we introduce a new component where each customer belongs to a class, $\theta$. Customers may either be price seekers (P) or quality seekers (Q), $\theta \in \{P, Q\}$. For a given problem, $\alpha$ represents the proportion of quality seeking customers. We will now redefine a number of terms from the traditional model. For the original definitions you may refer to Chapter 2. An illustration of these concepts is provided in Figure 5.1.

![Figure 5.1: Illustration of Heterogeneous Model Definitions](image)

**Base utility**, $u_j(y_j, \theta)$: The base utility of a product is the utility gained by a customer when purchasing a product at his preferred location. A high quality product has a base utility of $v + q$ for the quality seeker and all other product/customer combinations will only
produce \( v \) base utility. As such, \( u_j(y_j, \theta) \) is given by:

\[
u_j(y_j, \theta) = \begin{cases} 
v + q & \text{if } y_j = 1 \text{ and } \theta = Q \\
v & \text{Otherwise.}
\end{cases}
\]

**Purchasing utility, \( U_j(z, b_j, y_j, \theta) \):** The utility gained by a customer with preference \( z \) and class \( \theta \) from purchasing location \( b_j \) with quality \( y_j \) is given by \( U_j \). This utility is a surplus derived from subtracting the purchase price and travel cost from the base utility. Purchasing utility is defined as:

\[
U_j(z, b_j, y_j, \theta) = u_j(y_j, \theta) - p_j - t |z - b_j|.
\]

Each customer will purchase the product which gives him the highest purchasing utility, provided that utility is positive.

**Coverage distance, \( l_{\theta j} \):** For ease of analysis it is useful to compute the distance \( |z - b_j| \) where \( U_j = 0 \). This distance represents the maximum distance a customer’s preference can be from a product’s horizontal location where the customer receives nonnegative utility. This is known as coverage distance and it depends on the product quality and customer class and is independent of the actual location of a product. \( l_{\theta j} \) is given by:

\[
l_{\theta j} = \frac{u_j(y_j, \theta) - p_j}{t}.
\]

As a shorthand, we use the notation \( l_j \in \{l_L, l_{HP}, l_{HQ} \} \) where \( l_L \) is the coverage distance for a low quality product (regardless of customer class) and \( l_{HP} \) and \( l_{HQ} \) are the coverage distances of high quality products for price seekers and quality seekers respectively (note that \( l_L = l_{LP} = l_{LQ} \)).

**First choice interval, \([b_{\theta_j}^-, b_{\theta_j}^+]\):** The first choice interval of a product \( j \) gives the hori-
zontal locations of every customer in class \( \theta \) who will choose that product to purchase. The endpoints of the interval are given as:

\[
b_{\theta j}^- = \min\{z : U_j(z, b_j, u_j(y_j, \theta)) > U_i(z, b_i, u_i(y_i, \theta)) \quad \forall \ i \neq j\} \tag{5.1}
\]
\[
b_{\theta j}^+ = \max\{z : U_j(z, b_j, u_j(y_j, \theta)) > U_i(z, b_i, u_i(y_i, \theta)) \quad \forall \ i \neq j\} \tag{5.2}
\]

This definition assumes unique products – there will never be two products of the same quality at the same location (this would be an irrational decision for obvious reasons). Recall that this definition was simplified in the homogeneous quality preference model, but that simplification is no longer possible. For any product \( j \), we will refer to \([bP_j^-, bP_j^+]\) as the price seeker interval and \([bQ_j^-, bQ_j^+]\) as the quality seeker interval.

**First choice probability, \( d\theta_j(b, y) \):** From the first choice interval we can compute the probability that a given customer of class \( \theta \) will choose a product \( j \). This probability is given by:

\[
d\theta_j(b, y) = \int_{b\theta_j^-}^{b\theta_j^+} f(z) dz = F(b\theta_j^+) - F(b\theta_j^-).
\]

**Product profit, \( \Pi_j(b, y) \):** The profit for an individual product \( j \) in an assortment \((b, y)\) is given by:

\[
\Pi_j(b, y) = (p_j - c_j) \lambda [\alpha dQ_j(b, y) + (1 - \alpha) dP_j(b, y)] - K.
\]

**Assortment profit, \( \Pi(b, y) \):** The profit for an assortment \((b, y)\) is given by:

\[
\Pi(b, y) = \sum_{j=1}^{n(b,y)} \Pi_j(b, y).
\]
The goal of the assortment planning problem is to select \((b, y)\) in order to maximize \(\Pi(b, y)\). Note that if \(\alpha = 0\) (all price seekers) the problem reduces to the single quality homogenous problem described by Gaur and Honhon [4] and if \(\alpha = 1\) (all quality seekers) the problem reduces to the multiple quality homogenous problem described by Mayorga [15].

A few additional terms are needed to simplify analysis. Define the cumulative first choice price seeker probabilities given by an assortment \((b, y)\), for high and low quality products respectively, as:

\[
\beta = \sum_{\{j: y_j = H\}} dP_j(b, y) \\
\gamma = \sum_{\{j: y_j = L\}} dP_j(b, y).
\]

Note that \(\beta + \gamma \leq 1\). The cumulative first choice quality seeker probabilities are then given by:

\[
\beta' = \sum_{\{j: y_j = H\}} dQ_j(b, y) \\
\gamma' = \sum_{\{j: y_j = L\}} dQ_j(b, y).
\]

Again, \(\beta' + \gamma' \leq 1\).

Additionally, when referring to a given assortment we will use the shorthand \(n_H\) to denote a count of the number of high quality products in the assortment and \(n_L\) for the number of low quality products.
5.2 Analysis

With the homogeneous quality preference model, there were two key properties of the optimal solution which aided analysis (see Mayorga [15] and Chapter 2):

[P5.1] Coverage is non-overlapping – each location on \( \mathbb{R} \) is within the coverage distance of at most one product. This follows from the result that products will not be “dominated” – that is no product will have a first choice interval contained within another product’s first choice interval.

[P5.2] Coverage is continuous – there are no gaps in coverage within the assortment. Each location in the interval between the location of the first and last products is within the coverage distance of some product.

Mathematically, these properties taken together produce the constraint \( bQ^+_k = bQ^-_{k+1} \) for all \( k \in [1, ..., n(b, y) - 1] \).

However, by introducing heterogeneity in customer preferences, these properties must be re-evaluated. First, note the following observations in regards to property [P5.1]:

Observation 5.2.1. When \( 0 < \alpha < 1 \), products may overlap in the quality seeker intervals.

Example Consider an assortment containing two products, as in Figure 5.2(a), where \( y_1 = H \) and \( y_2 = L \). The first product is positioned where \( b_1 = l_{HQ} \) and the second where \( b_2 = 2l_{HQ} + l_L \) and \( 2l_L > 1 - 2l_{HQ} \). The profit for the assortment is given by:

\[
\Pi(b, y) = (p_H - c_H) \lambda [\alpha \beta' + (1 - \alpha) \beta] + (p_L - c_L) \lambda [\alpha \gamma' + (1 - \alpha) \gamma] - 2K.
\]

Now consider a second assortment, as in Figure 5.2(b), where \( \hat{b}_1 = b_1 \) and \( \hat{b}_2 = b_2 - \Delta; \Delta \in (0, \min (l_{HQ} - l_{HP}, 2l_L - (1 - 2l_{HQ}))) \). The profit for \((\hat{b}, y)\) is given as:

\[
\Pi(\hat{b}, y) = (p_H - c_H) \lambda [\alpha \beta' + (1 - \alpha) \beta] + (p_L - c_L) \lambda [\alpha \gamma' + (1 - \alpha) \gamma] - 2K.
\]
Figure 5.2: A Necessary Overlap Example
To prefer \((\hat{b}, y)\), \(\Pi(\hat{b}, y)\) must be greater than \(\Pi(b, y)\):

\[
\Pi(\hat{b}, y) - \Pi(b, y) = (p_H - c_H)\lambda \alpha \left[ \hat{\beta}' - \beta' \right] + (p_L - c_L)\lambda \alpha \left[ \hat{\gamma}' - \gamma' \right] + (1 - \alpha) \left[ \hat{\gamma} - \gamma \right].
\]

Note that \((p_H - c_H)\lambda \alpha \left[ \hat{\beta}' - \beta' \right] < 0\) and \((p_L - c_L)\lambda \alpha \left[ \hat{\gamma}' - \gamma' \right] + (1 - \alpha) \left[ \hat{\gamma} - \gamma \right] > 0\). As such, \((\hat{b}, y)\) is more profitable than \((b, y)\) when:

\[
(p_L - c_L) \left[ \alpha \left[ \hat{\gamma}' - \gamma' \right] + (1 - \alpha) \left[ \hat{\gamma} - \gamma \right] \right] > (p_H - c_H)\alpha \left[ \hat{\beta}' - \beta' \right].
\]

As a specific example, consider the following problem with \(F(z) \sim U(0, 1)\):

\[
\begin{align*}
p_H &= 1.8 & p_L &= 1.3 & c_H = c_L &= 0.5 \\
K &= 0.2 & \lambda &= 50 & \alpha &= 0.25 \\
v &= 2 & q &= 0.5 & t &= 2
\end{align*}
\]

In this case \(l_L = 0.35\), \(l_{HQ} = 0.35\), and \(l_{HP} = 0.1\). Consider an assortment \((b, y)\) with two products, where \(b_1 = 0.35\) and \(b_2 = 1.05\). The profit for this assortment, \(\Pi(b, y)\), is 32.725. Now consider a second assortment (again with two products), \((\hat{b}, y)\), where \(b_1 = 0.35\) and \(b_2 = 1.00\). The profit for this assortment, \(\Pi(\hat{b}, y)\), is 34.069.

With the homogenous quality preference model we only needed to search for \(b_1\) as the remaining product locations were determined by the quality levels of the products. For example, \(b_2\) could always be computed as \(b_2 = b_1 + l_{Q1} + l_{Q2}\). For the heterogeneous quality preference model we have an additional continuous variable for each product in the assortment as \(b_j\) may be less than \(b_{j-1} + l_{Q_{j-1}} + l_{Q_j}\). This result is significant because it causes a great increase in complexity for the problem. In the homogeneous quality preference model, searching for a single continuous variable proves to be a great hinderance to finding the optimal solution as (depending on the distribution of \(F(z)\)) the optimal objective value as a function of this variable is often non-concave (see Chapter 4. In the heterogeneous
quality preference model, we must search for $n(b, y)$ continuous variables. Further, these variables all depend upon one another and the sequence of product quality levels – none of these subproblems can be solved independently.

Property [P5.1] was implied by eliminating the possibility of dominated products in the homogenous quality preference model by not allowing the products to overlap. As this may not still hold, we must reevaluate the possibility of dominated products. For the heterogeneous quality preference model, we show that:

**Observation 5.2.2.** When $0 < \alpha < 1$, the optimal assortment may contain dominated products.

**Example** Recall that for the homogenous case it is never optimal to dominate products, because dominated products produce no profit. This is also true in the heterogeneous case when $\alpha$ is zero or one. In between, however, we can produce a partial domination when we
place a high quality and low quality product at or near the same location. In this case, the low quality product dominates for the price seekers and the high quality product dominates for the quality seekers.

Consider the assortments shown in Figure 5.3. In Figure 5.3a we have a low quality product \((b, y)\) at some location \(b_1\). Figure 5.3b shows a high quality product \((b, \hat{y})\) at the same \(b_1\) and Figure 5.3c shows both products \((\hat{b}, \hat{y})\) in the same assortment at the same location. Observe that the quality seeker will only see the high quality product and the price seeker will only see the low quality product. In some cases, this kind of domination can be more profitable than a single product. This suggests that an optimal assortment may include this behavior. We will examine the conditions under which this sort of domination can be beneficial.

The profit for each assortment can be split into its price seeker and quality seeker components. In the assortments above, the profits can be written as:

\[
\Pi(b, y) = (p_L - c_L)\lambda_2l_L + (p_L - c_L)\lambda(1 - \alpha)2l_L - K
\]

\[
\Pi(b, \hat{y}) = (p_H - c_H)\lambda_2l_{HQ} + (p_H - c_H)\lambda(1 - \alpha)2l_{HP} - K
\]

\[
\Pi(\hat{b}, \hat{y}) = (p_H - c_H)\lambda_2l_{HQ} + (p_L - c_L)\lambda(1 - \alpha)2l_L - 2K.
\]

Domination can be beneficial when \(\Pi(\hat{b}, \hat{y}) > \Pi(b, y)\) and \(\Pi(\hat{b}, \hat{y}) > \Pi(b, y)\). Substituting into these inequalities and removing like terms gives

\[
(p_H - c_H)2l_{HQ} > (p_L - c_L)2l_L + K
\]

\[
(p_L - c_L)2l_L > (p_H - c_H)2l_{HP} + K
\]

as one set of conditions under which we may have partial domination in the optimal solution.
For example, consider the following problem:

\[
\begin{align*}
  p_H &= 1.9 & p_L &= 1.7 & c_H = c_L &= 0.5 \\
  K &= 0.2 & \lambda &= 50 & \alpha &= 0.5 \\
  v &= 2 & q &= 0.35 & t &= 2
\end{align*}
\]

All products are placed at location 0.5. The profit for assortment \((b, y)\) is 17.8 and the profit for assortment \((b, \hat{y})\) is 19.05. However, the profit for placing both a high and low quality product at location 0.5 is 24.35. While this is not necessarily the optimal solution, it is an improvement over a single product. This sort of improvement by domination was not possible in the homogenous case.

With the homogenous quality preference model we know that the optimal assortment will never contain a dominated product because it could be shown that an assortment with dominated products could always be improved by modifying the assortment so that \(bQ^+_k = bQ^-_{k+1}\) for all \(k \in [1, ..., n(b, y) - 1]\) (or by removing the product). However, in the case above we show that a non-dominated assortment can be improved by creating a dominated product. Unfortunately, this also implies that products may overlap in both the quality seeker and price seeker intervals. With this information, the only constraint we have on product location is that no two products with the same quality will share the same location.

As for property \([P5.2]\), continuous coverage still holds, but only for the quality seeker intervals:

**Proposition 5.2.3.** Let \(\alpha \in [0, 1]\). There exists an optimal assortment containing \(n\) products such that \(bQ^+_j \geq bQ^-_{j+1}\) for all \(j < n\).

**Proof.** The proof is by contradiction. Suppose an assortment \((b, y)\) with corresponding profit \(\Pi(b, y)\) from \(n\) products is the only optimal solution and the assortment contains a
product $k < n$ such that $bQ_k^+ < bQ_{k+1}^-$. We construct an alternate assortment $(\hat{b}, y)$ with profit $\Pi(\hat{b}, y)$ and $bQ_k^+ = bQ_{k+1}^-$ such that $\Pi(\hat{b}, y) \geq \Pi(b, y)$, thus $(b, y)$ cannot be the uniquely optimal.

Consider two assortments, $(b, y)$ and $(\hat{b}, y)$, where $(\hat{b}, y)$ is the same as $(b, y)$ except that products $k+1$ through $n$ are shifted to the left by $\Delta = bQ_{k+1}^- - bQ_k^+$. We now compare the profits of the two assortments.

All products preceding product $k+1$ remain unchanged between the two assortments, thus

$$dP_j(b, y) = dP_j(\hat{b}, y) \forall j = 1, \ldots, k$$
$$dQ_j(b, y) = dQ_j(\hat{b}, y) \forall j = 1, \ldots, k$$

Products $k+1$ to $n$ are each shifted by $\Delta$ such that

$$b\hat{Q}_j^+ = bQ_j^+ - \Delta \quad b\hat{Q}_j^- = bQ_j^- - \Delta \quad \forall j = k+1, \ldots, n$$

As a result, there are three possible cases for products $j = k+1, \ldots, n$ based on their positions in $(b, y)$:

- **Case 1**: $bQ_j^+ < 1$ implies $dQ_j(\hat{b}, y) = dQ_j(b, y)$
- **Case 2**: $bQ_j^- < 1$ and $bQ_j^+ > 1$ implies $dQ_j(\hat{b}, y) > dQ_j(b, y)$
- **Case 3**: $bQ_j^- > 1$ implies $dQ_j(\hat{b}, y) \geq dQ_j(b, y)$

This result is useful because it still establishes an upper bound on the distance between products. Note that in the case of $\alpha = 0$ there is continuous coverage in the price seeker intervals. This proof follows from the single quality problem in Gaur and Honhon [4] and...
In the homogeneous quality preference case, we developed an upper bound on the optimal objective value (Chapter 3). However, this bound was shown to be poor in some cases for which the fixed cost \( K \) was high. For some distributions (uniform and unimodal) an optimal solution can be found for the homogenous case. For the heterogeneous quality preference case, we need to establish a bound for the evaluation of potential solution methods.

We submit the following to be true:

**Proposition 5.2.4.** For any assortment \((b, y)\), \(\Pi(b, y)\) is increasing with \(\alpha\).

**Proof.** Consider an assortment for which \(bQ_j^+ \geq bQ_{j+1}^-\) for all \(j \in [1, n-1]\). It is trivial to show that any assortment for which this is not true for all \(j \in [1, n-1]\) can be divided into assortments for which it is true for some interval. The total quality seeker coverage for \((b, y)\) is given by \(\beta' + \gamma'\) and the price seeker coverage is given by \(\beta + \gamma\). The profits for pure quality seeker and price seeker populations, \(\Pi_{QS}\) and \(\Pi_{PS}\) are given by:

\[
\Pi_{QS} = \lambda [(p_H - c_H)\beta' + (p_L - c_L)\gamma'] - n(b, y)K \\
\Pi_{PS} = \lambda [(p_H - c_H)\beta + (p_L - c_L)\gamma] - n(b, y)K.
\]

The total profit, \(\Pi(b, y)\) can be written as:

\[
\Pi(b, y) = \alpha \Pi_{QS} + (1 - \alpha) \Pi_{PS}.
\]

\(\Pi(b, y)\) is a linear combination of \(\Pi_{QS}\) and \(\Pi_{PS}\) with \(\alpha \in [0, 1]\). Hence, if \(\Pi_{QS} > \Pi_{PS}\), \(\Pi(b, y)\) is increasing in \(\alpha\). Recall that, by definition, \(l_{HQ} > l_{HP}\) and \(l_{LQ} = l_{LP}\). This implies \(\beta' + \gamma' \geq \beta + \gamma\). Also recall that, by assumption, \((p_H - c) > (p_L - c)\). This implies that if \(\beta' - \beta \geq \gamma - \gamma'\), then \(\Pi_{QS} > \Pi_{PS}\).
If \((b,y)\) consists of only low quality products, \(\beta - \beta' = 0\) and \(\gamma' - \gamma = 0\). If \((b,y)\) consists of only high quality products, \(\beta - \beta'\) is positive and \(\gamma' - \gamma = 0\). For a mixed assortment, consider the specific case where \(\gamma' - \gamma > 0\). This case can only happen when at least one low quality product overlaps at least one high quality product in the quality seeker interval. Because \(t\) is fixed, \(\beta' - \beta = \gamma - \gamma'\) (see Figure 5.4).

![Figure 5.4: Illustration of \(\Pi(b,y)\) Increasing in \(\alpha\)]

This result is significant for two reasons: first, for cases where we have an optimal solution (or upper bound) to the \(\alpha = 1\) case, we now have an upper bound on the optimal objective value. This allows us to assess the performance of proposed solution methods. Second, because the \(\alpha = 0\) case reduces to the Gaur and Honhon [4] model, for many cases we have a bounded interval over which the optimal objective value will exist. This can be advantageous, as when the optimal solution is unknown if we can produce a solution with an objective value in this range we can be reasonably sure that we have produced a “good” solution.
With the model defined we begin to explore solution methodology. In the following section we will describe one attempt at producing solutions for the heterogeneous quality preference model. Based on previous work, this solution method helps to illustrate the increased complexity of this problem and provides insight into future approaches to finding solutions.

### 5.3 Genetic Algorithm Solutions

Three solution methods have been proposed for the homogeneous quality preference model. For the case where $F(z)$ is uniform, McElreath, Mayorga and Kurz have provided an way to effectively enumerate the solution space using a modified knapsack approach [18]. For the unimodal case, both a heuristic genetic algorithm (Chapter 3) and an optimal dynamic programming (Chapter 4) approach have been proposed. The knapsack approach relies on property [P5.1] and is unavailable for the heterogeneous quality preference case. The dynamic programming approach could theoretically be used but is much too computationally expensive to be a practical method. This leaves the genetic algorithm as the only possibility of the existing solution techniques.

The possible existence of dominated products produces an impediment to the use of computational methods. To properly compute the objective value of a solution, we must be able to compute the first choice interval of each product in the assortment. Using equations (5.1) and (5.2) quickly become impractical for a reasonably sized problem as we must search the entire horizontal range and compute the purchasing utility of each product at each location. For this analysis we will make the assumption that we will not have dominated products in our assortment. This assumption will produce a suboptimal solution, but the objective value loss of this assumption should be reasonable for most problems but will be largest when $\alpha$ is close to 0. In that case the likelihood of a low quality products dominating high quality products for the price seekers increases. Using this assumption, we can calculate
the first choice intervals as follows:

\[
\begin{align*}
    b\theta_j^- &= \max \left\{ b_j - l_{\theta_j}, \max_{i<j} \frac{(p_j - u(y_j, \theta)) - (p_i - u(y_i, \theta)) + b_j t + b_i t}{2 t} \right\} \\
    b\theta_j^+ &= \min \left\{ b_j + l_{\theta_j}, \min_{i>j} \frac{(p_i - u(y_i, \theta)) - (p_j - u(y_j, \theta)) + b_j t + b_i t}{2 t} \right\}
\end{align*}
\]

### 5.3.1 Representation

In the representation for the homogeneous quality preference model, the distance between products was a length such that coverage was continuous and non-overlapping. A lower bound on the location of the first product was calculated \((b)\) and the representation used an offset from this value to locate the first product \((b_1)\). No other distances were needed. Our new representation allows for products to overlap by allowing for an individual offset for each product. The offset for product \(j\) gives the distance between \(b_{j-1}\) and \(b_j\) (the offset for product 1 is the distance between \(b\) and \(b_1\)).

By bounding the offsets we can reduce the size of the problem. Because we know that coverage must be continuous, we can find an upper bound on the offset distance given by \(\max (2l_{HQ}, 2l_{LQ})\). We can also find the maximum number of products in the assortment \((\bar{n})\). There are multiple heuristic means of finding \(\bar{n}\) that will be discussed later. The previous representation also included a large number of products and was able to dynamically determine how many would be included by “turning off” any products which yielded a non positive profit. This allowed for maximum flexibility (albeit sacrificed performance) by taking advantage of the modularity of the problem. In this case, however, we do not have that modularity. Whereas the profit for each product could be computed individually in the homogeneous case, in the current case the profit for each product depends heavily on the locations of the products which proceed and follow it. We do know ahead of time the maximum, \(\bar{n}\) number of products which can be in the assortment. Each solution is populated
with \( \bar{n} \) products. In our new representation we include a variable, *Express Length*, which is a uniform random integer on \([0, \bar{n}]\). This variable gives the number of genes to decode for that solution.

Formally, our representation consists of an integer value, *Express Length*, followed by a series of products. Each product consists of three components: an offset, a quality value (as in Chapter 3), and a random key (as in Chapter 3). The representation is illustrated in 5.5.

<table>
<thead>
<tr>
<th>Express Length</th>
<th>Quality</th>
<th>Offset</th>
<th>Random Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1.295</td>
<td>0.659</td>
<td>1.364</td>
</tr>
<tr>
<td></td>
<td>0.128</td>
<td>0.423</td>
<td>0.352</td>
</tr>
</tbody>
</table>

Figure 5.5: Illustration of new representation

### 5.4 Computational Studies

Our goal of the computational studies was to access the performance of the GA in generating “good” solutions for the heterogeneous problem. To that end, we performed small tests focused on tuning the parameters of the GA: population size, number of generations, stopping criteria, solution seeding, etc. We will begin by discussing a few of the parameters of interest for these experiments.

#### 5.4.1 Minimum distance between products

As stated, we do not allow domination of products in our assortments. However, we can (and do) allow products to “overlap” with one another, which is a significant change from the homogenous problem. This is important for two reasons: first, the minimum distance allowed between products determines the maximum number of products that can exist in
the assortment. More products leads to bigger chromosomes which leads to a large increase in the computational time required. Second, allowing the products to be closer together also requires an increase in the resolution of the search for the optimum distance between products. Changing this resolution can dramatically increase the size of the solution space, which in turn increases the computational time required. We do know from Proposition 5.2.3 that there is continuous coverage over the quality seeker intervals, so the optimal minimum distance is bounded over the interval \((0, l_{HQ}]\).

5.4.2 Solution seeding

For the homogenous case, the GA converged very quickly using random starting solutions. Because of the added complexity of the heterogeneous case, we observed in early experiments that convergence was much more slow. To obtain a somewhat uniform population required as much as several orders of magnitude more computational time than the homogeneous case. To aid in the speed of convergence, we developed construction heuristics to seed what we hypothesized to be reasonable starting point solutions into the population. Examples of these seeded solutions include:

- All high quality products with the \(b_1 = l_{HQ}\) and each additional product offset at \(2l_{HQ}\) for as long as full products will fit in the assortment. We then add this solution with and without a partial high quality product with the same offset (2 solutions).

- All low quality products with the \(b_1 = l_{L}\) and each additional product offset at \(2l_{L}\) for as long as full products will fit in the assortment. We then add this solution with and without a partial low quality product with the same offset (2 solutions).

- All high quality products with the \(b_1 = l_{HP}\) and each additional product offset at \(2l_{HP}\) for as long as full products will fit in the assortment. We then add this solution with and without a partial high quality product with the same offset (2 solutions).

- All high quality products with the \(b_1 = \frac{l_{HP} + l_{HQ}}{2}\) and each additional product offset at \(2\frac{l_{HP} + l_{HQ}}{2}\) including one partial product.
• A low quality product at location $b_1 = l_L$ with alternating high and low quality products with offset $l_{HQ} + l_L$ including one partial product.

• A low quality product at location $b_1 = l_L$ followed by a high quality product with offset $l_{HQ} + l_L$ followed by high quality products with offset $2l_{HQ}$ and one final partial low quality product at the end of the assortment.

• A low quality product at location $b_1 = l_L$ followed by a high quality product with offset $l_{HQ} + l_L$ followed by a high quality product with offset $2l_{HQ}$. This string is repeated until there is one partial product.

5.4.3 Reproduction and mutation

Reproduction occurs using parametric uniform crossover, which in experimentation provided superior results to single point crossover. Mutation for the Express Length occurs at a rate of 10% per chromosome, with a 0.50 probability of increasing by one and a 0.50 probability of decreasing by one. Each gene in the population then has a 10% mutation rate, and once mutation is chosen there is a 0.50 probability of mutating the offset – otherwise the quality level is mutated. If the quality level is chosen, it is changed to the opposite quality level. If the offset is chosen it will be raised or lowered with equal probability. The offset is modified by uniformly sampling a value between the current value and the minimum or maximum offsets. Mutated chromosomes are treated as offspring as opposed to replacing the original chromosome in order to keep our best solutions due to the high mutation rate. Each generation keeps the 20 solutions with the best objective values from the previous generation through elite reproduction. Chromosomes are then subject to mutation and the remaining population is created through reproduction.

5.4.4 Stopping criteria

The homogenous case used a fixed number of generations as a stopping criteria. In the heterogeneous case, the time needed for convergence varied greatly depending on the prob-
lem parameters. To allow proper time for convergence in all cases the stopping criteria was changed. In the heterogenous case we stop the GA after 20,000 generations without improvement in the best solution, provided that it has run for at least 50,000 generations. This threshold was chosen through experimentation with the total computational time required used as a consideration.

5.4.5 Results

On the whole, we believe the GA did not perform as well in finding solutions for the heterogeneous case, regardless of the parameters chosen. Specifically, the GA struggled to converge on a common assortment over multiple replications of the same parameters. This is particularly true in cases where $\alpha$ is close to 0.5. When $\alpha$ is close to 0 or 1 (approaching the homogeneous case), we could often converge to a common assortment. When $\alpha$ was equal to 0 or 1, we could produce a verifiably optimal solution but this does not reflect on the GA because this solution was seeded during the initialization process. After several rounds of tuning using large problem sets (12,654 cases), we focused on a smaller data set for closer examination using a set of 500 problem instances. Each instance was run for 10 replications. The following parameters were used:

Fixed: $p_L = 1.5, c = 0.5, \lambda = 2, K = 1, v = 2, t = 2, F(z) \sim U(0, 1)$

Variable: $p_H \in \{1.55, 1.6, ..., 1.95, 2.0\}, q \in \{0.025, 0.075, ..., 0.425, 0.475\}, \alpha \in \{0, 0.25, 0.50, 0.75, 1\}$.

We also chose a specific problem instance out of this set to perform a finer resolution study on the effects of $\alpha$. This set used 21 values of $\alpha$ and again was run for 10 replications. The following parameters were used:

Fixed: $p_H = 1.95, p_L = 1.5, c = 0.5, \lambda = 2, K = 1, v = 2, q = 0.475, t = 2, F(z) \sim U(0, 1)$

Variable: $\alpha \in \{0, 0.05, ..., 0.95, 1\}$.
Our best solutions were produced using a minimum distance than ensured no products would overlap in the price seeker intervals. To illustrate our results, consider the following example. This example was run with a population size of 250 using the stopping criteria described above. 10 solutions were seeded for the initial population and the remainder were generated randomly. The minimum of $l_{HP}$ and $l_{HQ}$ was given as the minimum distance. The parameters for used for this problem instance were:

$$
p_H = 1.95 \quad p_L = 1.5 \quad c = 0.5 \quad K = 1 \quad \alpha = 0.55$$

$$\lambda = 50 \quad F(z) \sim U(0, 1) \quad v = 2.0 \quad q = 0.475 \quad t = 2$$

Over ten replications the objective values and assortment sequences are given in Table 5.1 (sorted by objective value from highest to lowest).

<table>
<thead>
<tr>
<th>Objective Value</th>
<th>Assortment Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>53.49525286</td>
<td>LHHHLHHHL</td>
</tr>
<tr>
<td>53.378125</td>
<td>LHHHLHHHL</td>
</tr>
<tr>
<td>53.37788915</td>
<td>LHHHLHHHL</td>
</tr>
<tr>
<td>53.37738302</td>
<td>LHHHLHHHL</td>
</tr>
<tr>
<td>53.28260963</td>
<td>HHHHHHLHHL</td>
</tr>
<tr>
<td>53.24002276</td>
<td>LHLHHL</td>
</tr>
<tr>
<td>53.1972772</td>
<td>HHHHHHHHLHHL</td>
</tr>
<tr>
<td>53.19384043</td>
<td>LHHHLHHL</td>
</tr>
<tr>
<td>53.19311665</td>
<td>LHHHLHHL</td>
</tr>
<tr>
<td>52.81669647</td>
<td>HLHLH</td>
</tr>
</tbody>
</table>

Table 5.1: An Example of a GA Experiment

In this example we see the instability in the assortment sequence produced by the GA. On the other hand, we see that the variance in the objective value is small (0.0341), suggesting that the objective value is relatively stable. The worst objective value is 1.3% less than the best – while the absolute difference depends on $\lambda$, the percentage difference does not. These results are typical of what we find in the cases where $\alpha$ is close to 0.5, which are the most difficult cases. The remainder of the results follow a similar pattern. Of the 500
test cases in the first example, only 13 cases had a variance greater than zero over the ten replications. These 13 cases are shown in Table 5.2.

<table>
<thead>
<tr>
<th>$p_H$</th>
<th>$g$</th>
<th>$\alpha$</th>
<th>Mean of Objective Value</th>
<th>Variance of Objective Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.95</td>
<td>0.025</td>
<td>0.75</td>
<td>52.95</td>
<td>$3.67 \times 10^{-02}$</td>
</tr>
<tr>
<td>1.95</td>
<td>0.375</td>
<td>0.75</td>
<td>54.2442</td>
<td>$2.90 \times 10^{-03}$</td>
</tr>
<tr>
<td>1.95</td>
<td>0.425</td>
<td>0.75</td>
<td>54.8111</td>
<td>$1.84 \times 10^{-04}$</td>
</tr>
<tr>
<td>1.95</td>
<td>0.475</td>
<td>0.75</td>
<td>55.3931</td>
<td>$4.62 \times 10^{-06}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.225</td>
<td>0.5</td>
<td>48.4058</td>
<td>$9.57 \times 10^{-08}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.275</td>
<td>0.5</td>
<td>48.3706</td>
<td>$4.52 \times 10^{-02}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.325</td>
<td>0.5</td>
<td>48.9516</td>
<td>$9.12 \times 10^{-03}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.325</td>
<td>0.75</td>
<td>52.566</td>
<td>$1.23 \times 10^{-02}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.375</td>
<td>0.5</td>
<td>49.0882</td>
<td>$5.58 \times 10^{-02}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.425</td>
<td>0.5</td>
<td>49.4726</td>
<td>$3.61 \times 10^{-02}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.425</td>
<td>0.75</td>
<td>53.2621</td>
<td>$1.46 \times 10^{-03}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.475</td>
<td>0.5</td>
<td>49.5034</td>
<td>$1.97 \times 10^{-01}$</td>
</tr>
<tr>
<td>2.0</td>
<td>0.475</td>
<td>0.75</td>
<td>54.2323</td>
<td>$1.58 \times 10^{-02}$</td>
</tr>
</tbody>
</table>

Table 5.2: Variance of Test Cases

Using the same data from the second experiment, we can plot the average objective value for each $\alpha$ value. This is seen in Figure 5.6. The trend shows that our objective value is increasing in $\alpha$ as suggested in Proposition 5.2.4. While this does not guarantee that the objective values produced are close to optimal, the absence of this trend would imply that there was a problem with the objective values produced by the GA. All of our results follow this same trend.

Our results leave room for additional work on solution methods for the heterogeneous problem. One initial idea was the use of Tabu Search, as it is a path based metaheuristic as opposed to population based, and it worked well in many cases for the homogeneous problem. However, without a suitable neighborhood definition, we believe that Tabu Search would be computationally intractable. Allowing each product to have its own offset and allowing for modifications in the offset would produce a solution space so large that we
believe a path based metaheuristic would be unsuccessful in this form.

Figure 5.6: Objective Value as a Function of $\alpha$ for a GA Experiment
Chapter 6

Conclusions and Future Work

The assortment planning problem with multiple quality levels and locational choice presents a difficult problem, both analytically and computationally. In this dissertation, we have attempted to reduce the complexity of the existing models as well as present a new model which, while providing a more realistic outcome, has further increased the computational challenge. In the following section we will review our contributions from this dissertation. We will then close by discussing areas of opportunity for continuing this work.

6.1 Contributions

For the assortment planning problem with locational choice and homogeneous quality preferences, we have presented two solution methodologies for finding approximate and optimal solutions to an otherwise intractable problem. First, we used three metaheuristic techniques and showed that both Genetic Algorithms and Tabu Search may be successfully applied to this problem. Even for large problems, we can quickly generate quality solutions. Alternatively, we also introduce a dynamic programming based approach for generating solutions. Superior to the metaheuristics for small to mid size problems, the approach generates optimal solutions in a small amount of time and, due to its deterministic nature, only requires a single replication. These methods have impacts in multiple areas. First, from a practi-
cal standpoint, they allow for retailers and manufacturers to solve real world assortment planning problems. Second, from an analytic standpoint, they provide further insight into properties of optimal solutions and reveal the characteristics of assortments which were previously unknown. Third, we contribute back to the metaheuristic literature by providing performance results of several methods on a difficult continuous and combinatorial problem.

We have also presented the first model for the assortment planning problem with locational choice and heterogeneous quality preferences. This model relaxes a key assumption of the prior models and allows for a more accurate representation of many real world cases. We have provided a first attempt at generating solutions for this model which, while failing to converge on a common assortment in many cases, provides a suitable means of estimating the optimal objective value.

6.2 Future Work

For the homogeneous case, an opportunity exists for extending these solution methodologies to the joint assortment planning/inventory problems. With dynamic substitution we can no longer decouple the inventory and assortment planning problems and new techniques are needed. We could also consider competition between retailers, as this work assumes a monopoly for the retailer. For the heterogeneous case, there is an opportunity to improve the solution methodology used. A new heuristic could be applied or a dynamic programming approach similar to what was used in the homogeneous case could be tried. Additionally the model could be extended in a number of ways. In our model, all customers are either price seekers or quality seekers. Perhaps in reality there is some continuous gradient over preference for quality. Each of these problems has opportunity, but we believe the most helpful area of focus is on the assortment planning problem with heterogeneous quality preferences.
Appendices
Appendix A  Test Cases for Metaheuristic Computational Experiments

The following test cases were used for the metaheuristic study in Chapter 3. The following parameters are fixed for all cases: $\lambda = 50$, $c = 0.5, v = 2, t = 2$ and $p_L = 1.5$. This results in $l_L = 0.25$.

For instances 1–109 we use the following $pH$ and $q$ pairs with $K = 2$. Instances 110–218 use the same pairs with $K = 1$. 

<table>
<thead>
<tr>
<th>pH</th>
<th>1.55</th>
<th>1.6</th>
<th>1.65</th>
<th>1.7</th>
<th>1.75</th>
<th>1.8</th>
<th>1.85</th>
<th>1.9</th>
<th>1.95</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
</tr>
<tr>
<td></td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
<td>0.175</td>
</tr>
<tr>
<td></td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
<td>0.225</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
</tr>
<tr>
<td></td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
<td>0.275</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
<td>0.325</td>
</tr>
<tr>
<td></td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
<td>0.35</td>
</tr>
<tr>
<td></td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
<td>0.375</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
</tr>
<tr>
<td></td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
<td>0.425</td>
</tr>
<tr>
<td></td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
<td>0.45</td>
</tr>
<tr>
<td></td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
<td>0.475</td>
</tr>
</tbody>
</table>

Instances 1–109, with $K=0.2$, and for instances 110–218 with $K=1$. 

<table>
<thead>
<tr>
<th>$pH$</th>
<th>1.55</th>
<th>1.6</th>
<th>1.65</th>
<th>1.7</th>
<th>1.75</th>
<th>1.8</th>
<th>1.85</th>
<th>1.9</th>
<th>1.95</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2375</td>
<td>0.2125</td>
<td>0.1875</td>
<td>0.1625</td>
<td>0.1375</td>
<td>0.1125</td>
<td>0.0875</td>
<td>0.0625</td>
<td>0.0375</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.225</td>
<td>0.2</td>
<td>0.175</td>
<td>0.15</td>
<td>0.125</td>
<td>0.1</td>
<td>0.075</td>
<td>0.05</td>
<td>0.025</td>
<td>0</td>
</tr>
</tbody>
</table>
For instances 218–327 we use the following $pH$ and $q$ pairs with $K = 10$. Instances 328–436 use the same pairs with $K = 50$.

<table>
<thead>
<tr>
<th>$pH$</th>
<th>55</th>
<th>60</th>
<th>65</th>
<th>70</th>
<th>75</th>
<th>80</th>
<th>85</th>
<th>90</th>
<th>95</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q$</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>1.5</td>
<td>2</td>
<td>2.5</td>
<td>3</td>
<td>3.5</td>
<td>4</td>
<td>4.5</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
<td>27.5</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
<td>32.5</td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
<td>37.5</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
<td>42.5</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
<td>47.5</td>
</tr>
</tbody>
</table>
For instances 437–506 we use the following \( \text{pH} \) and \( q \) pairs with \( K = 25 \). Instances 547–656 use the same pairs with \( K = 100 \).

Parameters for all Instances

<table>
<thead>
<tr>
<th>( \text{Lambda} )</th>
<th>( c_H )</th>
<th>( c_L )</th>
<th>resulting ( l_L )</th>
<th>( v )</th>
<th>( t )</th>
<th>( p_L )</th>
<th>( \text{pH} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>5</td>
<td>5</td>
<td>0.4</td>
<td>100</td>
<td>100</td>
<td>60</td>
<td>62.5 65 67.5 70 72.5 75 77.5 80 82.5 85</td>
</tr>
</tbody>
</table>

```
<table>
<thead>
<tr>
<th>pH</th>
<th>62.5</th>
<th>65</th>
<th>67.5</th>
<th>70</th>
<th>72.5</th>
<th>75</th>
<th>77.5</th>
<th>80</th>
<th>82.5</th>
<th>85</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
</tr>
<tr>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
<td>2.5</td>
</tr>
<tr>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
</tr>
<tr>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
<td>12.5</td>
</tr>
<tr>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
<td>13.75</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
<td>17.5</td>
</tr>
<tr>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
<td>18.75</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
<td>22.5</td>
</tr>
<tr>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
<td>23.75</td>
</tr>
</tbody>
</table>
```

resulting \( l_H \)
Bibliography


