Maximizing Revenue Through Two-Dimensional Shelf-Space Allocation

Article *in* Production and Operations Management · November 2014

Impact Factor: 1.44 · DOI: 10.1111/poms.12316

4 authors:

H. Neil Geismar
Texas A&M University

Milind Dawande
University of Texas at Dallas

B. P. S. Murthi
University of Texas at Dallas

Chelliah Sriskandarajah
Texas A&M University

Maximizing Revenue Through Two-Dimensional Shelf-Space Allocation

August 12, 2014

Abstract

We consider the problem of optimally allocating contiguous rectangular presentation spaces in order to maximize revenues. Such problems are encountered in the arrangement of products in retail shelf-space and in the design of feature advertising displays or webpages. Specifically, we allow (i) the shape of a product's presentation to have a vertical as well as a horizontal component and (ii) displays to extend across multiple shelves for in-store presentations. Since the vertical location of the shelf on which a product is displayed affects its sales, each vertical location is assigned its own effectiveness with regard to revenue generation.

The problem of maximizing the total weighted revenue of a display is strongly NP-hard. Therefore, we decompose it into two subproblems. The first consists of allocating products to different cabinets. In the second, within each cabinet, each product's units are arranged in a contiguous rectangle and assigned a location. These subproblems are solved using an innovative approach that uses a combination of integer programming and an algorithm for the maximum-weight independent set problem. Based on computational studies on both real-world and simulated data, we demonstrate the efficiency and effectiveness of our approach. Specifically, the revenue generated by this scheme is within 1% of the optimum for actual data and within 5% for simulated data.

Key words and phrases: shelf-space allocation; two-dimensional display; retail; location effects

1 Introduction

We consider the problem of optimally allocating contiguous rectangular presentation spaces in order to maximize revenues. This problem is of great importance to retailers because the increasing number of brands and stock keeping units (SKUs) within each product category has increased the value of every inch of a retailer's shelf-space. In fact, shelf-space has been referred to as "the retailer's scarcest resource" (Bultez and Naert 1988, Lim et al. 2004). Drèze et al. (1994) estimate that the cost of shelf-space in a typical grocery store ranges from $20 per square foot for dry groceries to $70 per square foot for frozen foods. The shortage of retail shelf-space and the increasing intensity of competition have greatly magnified the importance of how merchandize is displayed.

The limited shelf life of many products intensifies this problem. A short lifetime may be due to physical characteristics (e.g., perishability of produce or other refrigerated products) or due to market factors (e.g., seasonality of fashion items or magazines). This enhances the need to display merchandize in a manner that maximizes revenue over a short period of time. Furthermore, of all the options available to a retailer for increasing revenue (such as advertising, adding new product lines, or promotional discounting), modifying the display of the current product portfolio may be the least expensive and the easiest to execute.

Most retailers currently employ PC-based software systems such as Apollo and Spaceman, which allocate space using either revenue or profit as the objective, and treat handling and inventory costs as constraints. The simple heuristics employed in these systems may not result in effective global optimization of shelf-space (Desmet and Renaudin 1998, Yang 2001, Lim et al. 2004). Some retailers also use planograms (diagrams—usually software-generated—of the display of retail products) to allocate shelf-space and to map out alternate arrangements on the screen without having to physically move the products. In such cases, planograms help retailers save time and effort, but they do not optimize the shelf-space usage (Drèze et al. 1994). In a recent study, Hansen et al. (2010) used store data to show that planogram-based heuristics led to a 1.7% improvement in performance. These studies suggest that there is a need to develop better optimization models for space allocation.

Whereas most of the models proposed in the literature (see Section 2) consider the length of shelf-space available to be the major resource that must be judiciously allocated to each product, others allow units of a product to be stacked upon one another, so a shelf's vertical space is also a key constraint. Some studies also recognize the importance of the height (i.e., vertical location) of a product's display (relative to eye-level), as this affects the likelihood that a customer will see it. A major contribution of our study is that it not only considers length and height, it also is the first to consider the *tallness* of a product's display, which is the vertical measurement of the space it occupies (see Figure 1). Tallness is significant because our model allows a product's display to extend over multiple vertically-adjacent shelves. To the best of our knowledge, no prior model has focused on the tallness of each product's display, though Murray et al. (2010) does allow a product's display to extend over more than one shelf, but with no coordination, e.g., horizontal alignment, of the units on those shelves.

This study is motivated by observed practice in DVD rental stores and our interactions with executives of Blockbuster, Inc. (Despite Blockbuster's demise in the U.S. market, it is still viable in Mexico, having generated 2.3B pesos (US$172.5M) in sales through 320 stores in 2013 (Laya 2014).) Movie titles rented on DVD have an extremely short product life cycle; the majority of rental activity occurs in the first two to four weeks of a title's release (Sawhney and Eliashberg 1996, Chung 2010, Chung et al. 2011, 2012). In contrast, a grocer's standard offering of goods and the demand for each good does not change substantially from week to week, which allows for learning through experimentation

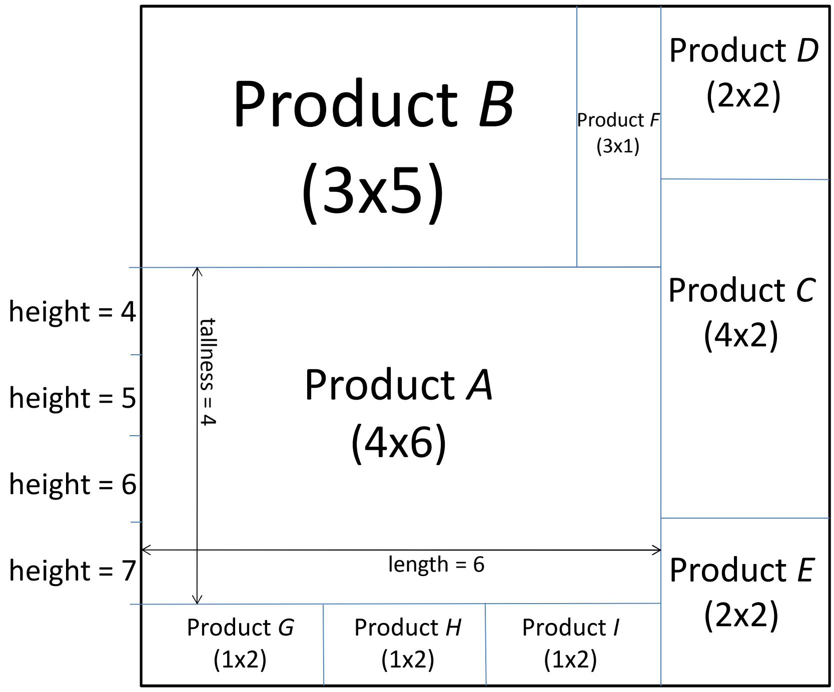

Figure 1: A schematic of an 8×8 layout under the proposed new policy.

by changing displays under relatively controlled conditions. Since this option is not available to a DVD store, managers need to accurately forecast the expected sales/rentals of each DVD title each week and to use this forecast effectively to allocate space to the titles. External studies of the DVD industry (Bhattacharya and Comerford 2006) also emphasize the importance of retail shelf-space for influencing the new releases' sales patterns.

Video rental stores typically arrange recently-released DVDs (for which the customer must pay a premium) in cabinets along the walls and in approximate alphabetical order (explicitly defined in Appendix C) so that customers can easily find a particular title. Within the cabinet the alphabetical order is not precise because the vertical locations of displays are allocated based on the titles' revenue potential. This is done because Blockbuster's internal studies show that the middle shelves are more effective for generating sales since they have higher visibility. Another contribution of this study is the development of a process to achieve this alphabetization without significantly reducing the total weighted revenue of the display.

Most of Blockbuster's cabinets holding new releases contain eight shelves that can each hold eight

DVDs. In the current practice, those titles for which at least eight discs are displayed consume an integer number of shelves, i.e., some integer multiple of eight slots, within one cabinet. Titles with the greatest revenue potential (generally three-to-five titles per week) each consume an entire cabinet. Others whose revenue potential merits at least eight discs are generally concentrated in the middle four rows (rows 3, 4, 5, 6) of their respective cabinets, with those having larger revenue potential in rows 4 and 5. The remaining new releases occupy fewer than eight slots and generally are relegated to rows 1, 2, 7, and 8. Such a title's discs are displayed in only one row. Those within this group having larger revenue potential are in rows 2 and 7. This heuristic is easy to implement, but the restriction of high-revenue titles to full shelves severely constrains a manager's choices by limiting the possible quantities displayed.

A DVD vendor has many display options beyond those specified by this heuristic because all DVDs have the same shape and size, i.e., all units are interchangeable. This makes our problem a generalization of the Unit Length Shelf-Space Allocation Problem (ULSSAP) (Lim et al. 2004) to two dimensions. We propose a new policy to arrange the discs for each title in its own contiguous rectangle within a single cabinet. The new policy does not mandate either dimension of any title's rectangle, so managers have greater flexibility in the number of units displayed for each title. This allows for displays that can generate greater revenue, but it also makes the problem more challenging. Figure 2 demonstrates how a display generated by our proposed methodology could differ from one generated by current practice.

Many other products can fit within our generalization of the ULSSAP, including canned goods, potato chips, breakfast cereals, frozen foods, dairy products, and video games. The stacking of wine, floral arrangements, and other products in kiosks (e.g., at malls or airports) and the layout of snacks or electronics in vending machines (also at airports) further suggest that this is an important problem. More generally, the unit length and height requirements are not especially restrictive because the units considered could simply be square inches of display space. A third contribution of this study is the generalization of our analysis to display ads, web pages, and differently-shaped products, and to the incorporation of the effects of product interactions.

Empirical research in the Marketing field also supports our new scheme. Pieters et al. (2010) show that an increase in *design complexity*, which is the intricacy of the shapes, arrangements, and organization of a display, increases the viewers' attention, comprehension, and approval of the items being considered (this is based on research in Psychology, e.g., Berlyne (1958)). Conversely, an increase in *feature complexity*, which is the variations in the basic visual content such as color, luminance, and

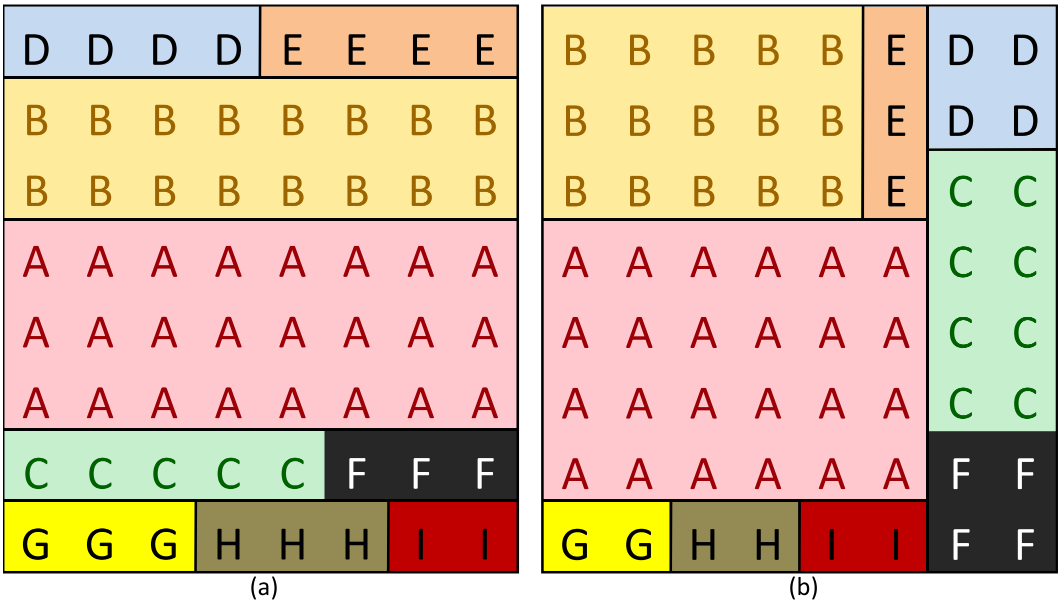

Figure 2: (a) A display generated from current practice compared to (b) one from our proposed methodology. Some products differ in the number of units displayed because of the schemes' different requirements.

edges, decreases the viewers' interest. These concepts map nicely onto our problem. Displaying the discs of the DVD titles in rectangles of varying aspect ratios (*length*:*tallness*) adds design complexity to the display, but it does not impose the feature complexity that would arise from having discs of various titles randomly scattered throughout a display or aggregated into non-rectangular shapes.

The problem's analytic complexity leads us to develop a novel solution methodology by decomposing the general problem (without alphabetization) into two subproblems. The first consists of allocating products to different cabinets. In the second, within each cabinet, the units of each product are arranged in contiguous rectangles that together cover the entire cabinet. These subproblems are solved via an innovative approach that uses a combination of integer programming and an algorithm for the maximum-weight independent set problem. The resulting weighted revenue is within 1% of the optimum for actual data from Blockbuster stores and within 5% of the optimum for randomly generated data. Achieving this revenue level with a scheme that increases both a manager's options and the display's design complexity is this study's key contribution. We also illustrate the percentage increase in revenue realized from the new policy.

Here is the outline of this paper. Section 2 reviews previous studies on retail shelf-space allocation. Section 3 describes our model. Section 4 details our solution procedure. Section 5 provides the computational results for studies based on observed data from a Blockbuster, Inc., retail store. Section 6 provides extensions of our model to include other practical environments. Section 7 concludes this study and proposes directions for future research. The appendices present proofs of theoretical results, mixed integer linear programming formulations, the algorithm to improve the alphabetical ordering of a near-optimal solution with minimal reduction of total weighted revenue, an example of forming a display via the maximum weight independent set problem, and results of our computational study on randomly generated data.

2 Literature Review

We begin with the influence of shelf-space on sales and some allocation schemes that put these principles into practice. Later studies acknowledge that a display's vertical location affects sales. Table 1 summarizes the literature.

The common practice in retailing is to allocate display space in proportion to a product's unit sales, revenue, or profit (Bultez and Naert 1988). Early research shows that shelf-space allocation affects sales by stimulating demand (e.g. Curhan 1972, Anderson and Amato 1974, Corstjens and Doyle 1981, 1983). These studies led Bultez and Naert (1988) to develop a heuristic that allocates shelf-space based on *within* product-class sales-share elasticities (both direct and cross-elasticities), as opposed to elasticities *between* product-classes. This is a reasonable approach because space allocation between classes is usually a strategic problem, whereas allocation within a class is an operational decision (Bai et al. 2012). Yang and Chen (1999) solve a more tractable problem by assuming that a product's profit function, although non-linear globally, is linear within a small range of its current number of facings (the *number of facings* is the number of units visible to the customer). Later studies by Yang (2001) and Bai et al. (2012) each use a two-phase heuristic that considers problems in which each product has a lower bound and an upper bound on the number of facings, and each unit of a given product must reside on the same shelf, i.e., at the same height. Lim et al. (2004) suggest a network flow approach that combines local search and heuristics to arrive at better solutions. Bai and Kendall (2008) allocate display space for products while simultaneously determining inventory and ordering policy.

Most of the research into the allocation of shelf-space assumes that it is homogeneous, i.e., the

Shelf Space Optimization Models	**Strengths**	**Limitations**	**Solution Methodology**
Corstjens and Doyle (1981)	Margins, inventory costs, shelf space elasticity, cross-elasticities	No marketing mix variables, allows for fractional facings	Geometric programming
Corstjens and Doyle (1983)	Different products have different growth potential. All else same as in 1981 paper.	No marketing mix variables, allows for fractional facings	Dynamic programming
Zufryden (1986)	Marketing mix, space elasticity, cost of sales	No cross-elasticities. Does not optimize marketing mix	Dynamic programming, simulation
Bultez and Naert (1988)	Within product class elasticities, profits are a function of shelf-space	Highly complex, allows for fractional facings	Heuristic based on non-linear equations
Borin et al. (1994)	Decreasing marginal sales in response to increasing a product's display space	Non-linear objective function	Simulated annealing
Yang and Chen (1999)	Profit is a linear function of shelf space within a small range	Approximation to a non-linear programming model	Integer programming, hierarchical, multi-stage
Yang (2001)	Objective function is a linear function of shelf-space. Each product has min and max number of facings. Vertical locations have different values.	Two-stage process with myopic first stage, limited test data	Integer programming, iterative allocation based on gradient
Lim et al. (2004)	S-shaped profit functions, cross elasticities, groupings and affinities	Non-linear profit function, 1-dimensional displays	Network flow, local search, heuristics, tabu search, squeaky wheel
Hwang et al. (2005)	Vertical location affects sales, integrates inventory control with display decisions	Non-linear objective for which it is difficult to find closed form optimal solution	Gradient search heuristic, genetic algorithm
Bai and Kendall (2008)	Simultaneously decides ordering policy and allocates shelf-space among different products	Continuous decision variables, prone to local optima	Generalized reduced gradient algorithm
Bai et al. (2012) Murray et al. (2010)	2-dimensional displays: units may be stacked. Each product has min and max number of facings. Vertical positions have different values.	2-dimensional displays are within one shelf	Integer programming, iterative allocation based on gradient, simulated annealing
Current paper	2-dimensional displays, considers the effect of vertical position	No cross-elasticity or shelf elasticity, no marketing mix effects	Integer programming, heuristics, network formulation

Table 1: Related literature.

amount of shelf-space consumed is the only significant factor, and the position—vertical or horizontal—in the display is ignored. Studies that do not make this limiting assumption include Drèze et al. (1994), who conclude that a product's vertical location is more important than the amount of shelf-space it occupies, noting that most products generate more revenue when displayed at eye level (see also Van Nierop et al. 2008 and Desmet and Renaudin 1998). Recent empirical evidence also suggests that vertical location affects sales by twice as much as horizontal shelf length (Hansen et al. 2010), which is consistent with the magnitude of vertical location effects observed in Drèze et al. (1994).

The model of Murray et al. (2010) allows a product to be allocated to multiple shelves. This model explicitly considers the two-dimensional area consumed by each product, but does not account for the number of shelves occupied by each product (i.e., the product's display's tallness). Also, there is no requirement that these shelves' displays be coordinated: the two (or more) placements of one product need not cover the same linear width on each shelf, need not be on vertically adjacent shelves, nor do they need to be aligned horizontally. Hence, not only does this model generate displays with high feature complexity (defined in Section 1), it would be difficult to apply it to feature advertising or to webpage design. Bai et al. (2012) allow a product's units to be stacked within one shelf. However, this model does not address displays across multiple shelves, so the tallness of a product's display is strictly limited and cannot significantly exceed that of any other product.

Our study follows from the above-described line of research by assigning upper bounds and lower bounds to the number of units that can be displayed for each product and by recognizing the value of a display's vertical location when maximizing revenue. For each product, we use its *revenue potential* (defined in Section 3) to calculate its maximum number of facings. These potentials and the relative values of the different shelves' heights are used to determine the allocation of display locations that maximizes the total weighted revenue of the display. Our work is distinguished further in that we allow each product to be displayed on multiple shelves and require that the units of each product be displayed in a contiguous rectangle; this allows generalization to feature and webpage advertising. We know of no previous study of shelf-space allocation that has considered such coordinated two-dimensional displays for each product to maximize total weighted revenue. This feature—allocating and arranging rectangular spaces with minimal restriction on either dimension—necessitates the development of new solution methods; thus, we formulate it as a maximum weight independent set problem on a network (*MWIS*).

Characteristics of the DVD rental business preclude the consideration of complementarity or cross-price elasticity effects in our case study. First, the titles' short lifecycles and their approximately alphabetical ordering prevent the trials required to calculate these effects. Second, the price of a movie rental is the same across all the newly-released titles. We describe how our model could be extended to include complementarity effects for studying other applications in Section 6.

3 Conceptual Model

We start the description of our conceptual model with variable and parameter definitions:

Parameters:

D	the number of products to be displayed.
r_d	revenue potential for product d, $d = 1, \ldots, D$.
u_d	upper bound on the number of facings for product d, $d = 1, \ldots, D$.
ℓ_d	lower bound on the number of facings for product d, $d = 1, \ldots, D$.
K	number of cabinets.
R_k	number of rows in cabinet k, $k = 1, \ldots, K$.
C_k	number of columns in cabinet k, $k = 1, \ldots, K$.
a_i	the effectiveness of row i, $i = 1, \ldots, R_k$; $k = 1, \ldots, K$.

Variable:

y^k_{dij} $= 1$, if product d is displayed in row i and column j of cabinet k,
$= 0$, otherwise.

We now formally state the problem:

Display problem *P*: D products of the same shape and size are to be displayed in K cabinets. Cabinet k has R_k rows and C_k columns, hence $R_k \times C_k$ slots (capacity). A unit can be placed in each slot, i.e., unit size is equal to slot size. Each slot must contain exactly one item. Each product must have all of its units displayed within a single cabinet, and those units must be displayed in a contiguous rectangle. The revenue potential per unit displayed of product d is r_d; $d = 1, 2, \ldots, D$, and a_i denotes the display effectiveness of row i within cabinet k, $i = 1, 2, \ldots, R_k$; $k = 1, \ldots, K$. For product d, u_d (respectively, ℓ_d) is the upper bound (respectively, lower bound) on the number of units to be displayed. The objective is to maximize the total weighted revenue of the display. This objective function can be expressed as $F = \sum_{d=1}^{D} r_d \sum_{k=1}^{K} \sum_{i=1}^{R_k} a_i \sum_{j=1}^{C_k} y^k_{dij}$, where $y^k_{dij} = 1$ if a unit of product d is displayed in slot (i, j) of cabinet k; otherwise $y^k_{dij} = 0$. Without loss of generality, rows are numbered from top to bottom, and columns are numbered from left to right. The terms *row* and *height* are used interchangeably.

When considering the shelf management problem, a retailer typically knows the assortment of D products to be displayed for a given planning horizon. To model the large variations in demand among the products displayed, each product d has its own *revenue potential* per unit displayed r_d for the current planning period (no time index is required because the planning periods are analyzed independently). The revenue potential is calculated by multiplying the expected demand for the product by its price. For our computational study, the expected demands are calculated via the model of Chung (2010) and Chung et al. (2012), which is based on the well-known Bass (1969) model; more details may be found in Subsection 5.1.

We use the revenue potentials to determine the upper bound u_d on the number of units to be displayed for each product. Upper bounds enforce the differences in the amount of display space assigned to the different products. Specifically, a product's upper bound is based on the percentage of the sum of all revenue potentials that the product's revenue potential represents $\left(r_d / \sum_{h=1}^{D} r_h\right)$. Thus, the relative ratios of the products' revenue potentials have a major influence when allocating display space.

Because each product must have all of its units in the same cabinet, each product's upper bound should not exceed the largest cabinet's capacity ($u_d \leq \max_k\{R_k C_k\}$). In practice, a product d' may have a relative revenue potential large enough to warrant an upper bound $u_{d'}$ that is greater than $\max_k\{R_k C_k\}$. Our system can model this as being two products, one of which occupies an entire cabinet k', with the second title having the upper bound $u_{d'} - R_{k'} C_{k'}$.

Each product also has a lower bound ℓ_d on the number of its units to be displayed. This constraint may represent a manager's decision to maintain a certain assortment of products, or it may enforce contractual obligations with product suppliers (Yang 2001) that require minimum display areas.

Because the vertical location at which a product is displayed may be as important as the amount of shelf-space it occupies (Drèze et al. 1994, Zytkowska 2003, Bai et al. 2012), we use the row *effectiveness* a_i to quantify the effect of display location. To incorporate Blockbuster's findings on display height, the a_i values are large for the middle rows and decrease for rows closer to the top or bottom. The *weighted revenue* from displaying one unit of product d on shelf i for one period is defined as $a_i r_d$ (this is analogous to the quantity that Yang and Chen (1999) label p_{di} and call the *per facing profit* of product d on shelf i, but they do not specify how its value is found), where *weighted* refers to the influence of the row effectiveness values a_i.

Some previous works on shelf space allocation, e.g., Drèze et al. (1994), van Nierop et al. (2008), Murray et al. (2010), propose diminishing marginal returns on the number of units displayed for a given product. Our objective function uses the row effectiveness values a_i to approximate this concavity while maintaining a linear objective and, thus, tractability. As a product's display becomes larger, it extends over more rows, typically away from the middle rows to rows with lower effectiveness. Thus, the product's total weighted revenue per unit displayed decreases.

Note also that our problem considers only the *display*, i.e., that which the customer sees as he or she peruses the shelves. We ignore on-shelf inventory that is stored behind the units facing the customer, in accordance with Drèze et al. (1994) and Murray et al. (2010). Another distinguishing requirement of this study is that the units for a particular product must be displayed contiguously within a single cabinet and in a rectangular configuration. One could simplify the problem by treating the entire display area as one large cabinet, but this would reduce the generality of our solution, e.g., it would prohibit using it to design a multi-page flyer. Furthermore, it would not represent standard retail practice.

Theorem 1 asserts that our problem is strongly NP-hard. It is proven in Appendix A.

Theorem 1 *The display problem P is strongly NP-hard.*

4 Solution Procedure

Our formulation of problem P as a mixed integer linear program (MILP) is in Appendix B.1. Neither CPLEX nor Xpress-MP (commercial MILP solving software packages) could solve practical-sized (500 products) versions of this formulation before running out of memory (CPLEX) or exhausting a two-week time limit (Xpress-MP). Therefore, to find good solutions efficiently for large-scale problems, we break P into two subproblems:

SP1: Assign products to cabinets.

SP2: Arrange units within cabinets.

This decomposition is natural, in the sense that it allows us to break the multi-cabinet optimization problem into several smaller optimization problems, one for each cabinet. We also solve a third subproblem for Blockbuster that also applies to other applications, e.g., canned soups:

SP3: Improve alphabetical ordering.

We describe our solution procedures for Subproblems *SP1* and *SP2* in the following subsections. The procedure for *SP3* is presented in Appendix C. Figure 3 schematically summarizes our approach.

4.1 Subproblem 1: Assign Products to Cabinets

Mixed-integer program *MIP-SP1* assigns products to cabinets. Because it does not specify the particular slot for each unit, *MIP-SP1* cannot use the row effectiveness values to evaluate the total weighted revenue. The resulting objective maximizes the number of units assigned to a cabinet for products with larger revenue potentials and minimizes the units for products with smaller potentials, while allocating products with similar revenue potentials evenly across the cabinets. We use the following decision variables:

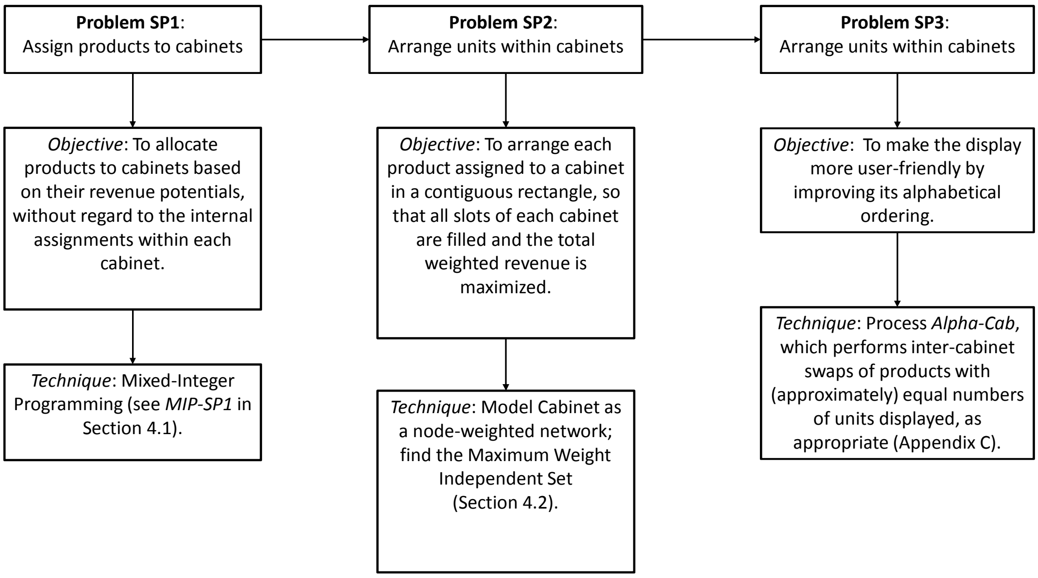

Figure 3: A schematic representation of our solution approach.

$x_d^k = 1$, if product d is allocated to cabinet k; $x_d^k = 0$, otherwise.

z_d^k: the number of units of product d allocated to cabinet k.

$$(\textit{MIP-SP1}) \quad \text{maximize} \sum_{d=1}^{D} \sum_{k=1}^{K} r_d z_d^k, \tag{1}$$

subject to

$$\sum_{k=1}^{K} x_d^k = 1, \quad d = 1, \ldots, D, \tag{2}$$

$$\sum_{d=1}^{D} z_d^k \leq R_k C_k, \quad k = 1, \ldots, K, \tag{3}$$

$$x_d^k \ell_d \leq z_d^k, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K, \tag{4}$$

$$z_d^k \leq x_d^k u_d, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K, \tag{5}$$

$$x_d^k \in \{0, 1\}, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K, \tag{6}$$

$$z_d^k \geq 0, \; z_d^k \text{ integer}, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K. \tag{7}$$

The objective (1) represents the total revenue of the display scheme without regard to row effectiveness: each product's revenue potential is multiplied by its number of facings. Constraints (2) ensure that all of the units for a specific product will be stored in exactly one cabinet. Constraints (3) limit the number of units assigned to each cabinet to the number of slots in that cabinet. Constraints (4) and (5) enforce lower bounds and upper bounds, respectively, on the number of slots allocated to a particular product. Constraints (6) ensure that the x-variables are binary. Constraints (7) ensure that the z-variables are non-negative integers.

Because *SP1* is a relaxation of problem *P*, *SP1* is infeasible only if *P* is infeasible. Two obvious examples of conditions that cause infeasibility are $\sum_d \ell_d > \sum_k R_k C_k$ and $\sum_d u_d < \sum_k R_k C_k$. Other combinations of parameter values could also lead to infeasibilities, such as $D = 4$, $K = 2$, $R_1 = R_2$, $C_1 = C_2$, and $\ell_d > R_1 C_1/2$ for $d = 1, 2, 3$.

Assume (without loss of generality) that $r_1 \geq r_2 \geq \cdots \geq r_m$ for the m products that *MIP-SP1* assigns to cabinet k. One can easily show that there exists an integer $q \in \{1, \ldots, m\}$ such that the numbers of units displayed for the first $q-1$ products match their respective upper bounds, the numbers of units displayed for the last $m-q$ products match their respective lower bounds, and the q^{th} product is unrestricted between its lower and upper bound. Formally, for cabinet k,

- $z_d^k = u_d$, for $d = 1, 2, \ldots, q-1$,
- $z_d^k = \ell_d$, for $d = q+1, q+2, \ldots, m$,
- $z_q^k = (R_k C_k - \sum_{i=1}^{q-1} u_i - \sum_{i=q+1}^{m} \ell_i)$, where $\ell_q \leq z_q^k \leq u_q$,

and q is maximal. In the following example, $q = 2$. This example demonstrates the allocation of units to a cabinet by *MIP-SP1*.

Product	d	**Revenue potential (r_d)**	**Upper bound (u_d)**	**Lower bound (ℓ_d)**
A	1	12	9	2
B	2	5	4	2
C	3	3	3	2
D	4	2	2	2

Table 2: Data for Example 1.

Example 1: Suppose *MIP-SP1* assigned four products to cabinet k with $R_k = 4$ and $C_k = 4$, and that the revenue potentials, upper bounds, and lower bounds are as given in Table 2. The resulting assignment is $z_1^k = 9$, $z_2^k = 3$, $z_3^k = 2$, and $z_4^k = 2$. If we assume that the nominal values for effectiveness of row i are $a_1 = a_4 = 1$, $a_2 = a_3 = 2$, then the arrangement shown in Figure 4 satisfies this assignment and maximizes total weighted revenue simply by putting the product with the largest revenue potential in the rows with highest effectiveness. This solution, however, is infeasible for the full problem because it does not satisfy the contiguous rectangles requirement.

C	C	D	A
A	A	A	A
A	A	A	A
B	D	B	B

Figure 4: A display for Example 1 assigned by *MIP-SP1*. The total weighted revenue is 229. A facing of product d in row i has a weighted revenue of $a_i r_d$.

4.2 Subproblem 2: Arrange Units within Cabinets

After each product has been assigned to a cabinet, the units for each product must be displayed in a contiguous rectangle within its cabinet. Figure 5 shows the cabinet of Example 1 after the display has been arranged to obey this constraint.

B	A	A	A
B	A	A	A
B	A	A	A
C	C	D	D

Figure 5: A feasible display for Example 1. The total weighted revenue is 215. A facing of product d in row i has a weighted revenue of $a_i r_d$.

For clarity and precision, we formally state subproblem *SP2* and assert its intractability in Theorem 2.

Subproblem *SP2***:** m products of the same shape and size are to be displayed in cabinet k. The cabinet has R_k rows and C_k columns, hence $R_k \times C_k$ slots (capacity), and a unit can be placed in each slot in the sense that unit size is equal to slot size. Each slot must contain exactly one item. Each product's display must be in a contiguous rectangle. The revenue potential of product d is r_d, where

$d = 1, 2, \ldots, m$, and a_i denotes the display effectiveness of row i within the cabinet, $i = 1, 2, \ldots, R_k$. For product d, u_d (respectively, ℓ_d) is the upper bound (respectively, lower bound) on the number of units to be displayed. The objective is to maximize the total weighted revenue of the cabinet's display. This objective function can be expressed as $F' = \sum_{d=1}^{m} r_d \sum_{i=1}^{R_k} a_i \sum_{j=1}^{C_k} y_{dij}$, where $y_{dij} = 1$ if a unit of product d is displayed in slot (i, j); otherwise $y_{dij} = 0$. Without loss of generality, rows are numbered from top to bottom, and columns are numbered from left to right.

Theorem 2 *Problem SP2 is NP-hard in the ordinary sense.*

Theorem 2 is proven in Appendix A.

To solve *SP2* efficiently for any cabinet k, we formulate it as a maximum-weight independent set problem (*MWIS*) on a network. An *independent set* on a network is a collection of nodes in which no pair shares an edge. The weight of an independent set is the sum of the weights of its nodes. Each node of our network represents a possible rectangular display for a product, i.e., it specifies the product and the slots covered by a specific rectangle. A node's weight is the total weighted revenue generated by placing that product in that rectangle. The network's edges are drawn so that an independent set represents a display in which (i) each product has exactly one rectangle and (ii) no two rectangles occupy the same slot (explicit specifications of nodes and edges are provided below). Hence, a maximum-weight independent set represents a feasible display with maximum total weighted revenue. We now summarize the network's creation with pseudo-code, followed by a detailed description.

Algorithm Form Network

Create Nodes:

For each product

    For each possible number of units

        For each feasible aspect ratio for a rectangle

            Find all feasible placements and create one node for each feasible placement

Create Edges:

For each product

    Form a clique of all nodes representing that product

For each slot in the display

    Form a clique of all nodes representing placements that cover that slot.

To form nodes, we use the number of units of each product that is specified by the solution to *SP1*, and allow one unit of slack. Thus, because the solution to *SP1* has z_d^k units for product d in cabinet k, consider all feasible rectangles that cover $\max\{\ell_d, z_d^k - 1\}$, z_d^k, or $\min\{u_d, z_d^k + 1\}$ slots. Recall that for at most one product (labeled q above) will these be three distinct numbers: for $d < q$, consider $u_d - 1$ or u_d units; for $d > q$, consider ℓ_d or $\ell_d + 1$ units. All feasible rectangles for each number of units are candidates for placements. Table 3 provides all possible rectangles for the products in Example 1; recall $q = 2$.

Product (d)	z_d^k	**All possible rectangles for each product**
A	9	2×4, 3×3, 4×2
B	3	1×2, 2×1, 1×3, 3×1, 1×4, 4×1, 2×2
C	2	1×2, 2×1, 1×3, 3×1
D	2	1×2, 2×1, 1×3, 3×1

Table 3: All possible rectangles for products in Example 1, based on z_d^k, ℓ_d, and u_d.

Given a possible rectangle for product d with length L_d and tallness T_d, its top-left slot can be in any row between 1 and $R_k - T_d + 1$ and in any column from 1 to $C_k - L_d + 1$. To illustrate, if Product A in Example 1 is displayed in a 3×3 rectangle, then its top-left slot may be any one of (1,1), (1,2), (2,1), or (2,2), where (ρ, c) indicates the slot in the ρ^{th} row and c^{th} column. Table 4 provides all possible placements for each rectangle of product A and their corresponding nodes in the *MWIS* network. For example, node A_1 signifies that product A occupies the first two rows. Its weight is $W(A_1) = r_1 \times (a_1 + a_2) \times (c_2 - c_1 + 1) = 12 \times (1 + 2) \times 4 = 144$, where c_1 and c_2 are the first and last columns, respectively, of placement A_1.

Product A	**Aspect Ratio**	**Possible placements in cabinet (rows, columns)**	**Nodes for placements**
Rectangle 1	2×4	(1-2,1-4), (2-3,1-4), (3-4,1-4)	A_1, A_2, A_3
Rectangle 2	3×3	(1-3,1-3), (1-3,2-4), (2-4,1-3), (2-4,2-4)	A_4, A_5, A_6, A_7
Rectangle 3	4×2	(1-4,1-2), (1-4,2-3), (1-4,3-4)	A_8, A_9, A_{10}

Table 4: All possible placements for Product A of Example 1 for subproblem *SP2*. Notation (ρ_1-ρ_2,c_1-c_2) means that the rectangle covers the intersection of rows ρ_1 through ρ_2 with columns c_1 through c_2.

An edge connects two nodes if the corresponding rectangles cannot both be in a feasible display. The cliques formed for each product need no further explanation. To demonstrate the edges that connect two nodes that share a slot, consider node A_1 = (1-2,1-4) and node B_9 = (2-4,1-1), which covers the lower three elements of the first column. They share slot (2,1), so they should be joined by an edge. Formally, consider two nodes X_j = (ρ_1-ρ_2,c_1-c_2) and Y_ℓ = (ρ_3-ρ_4,c_3-c_4), for two distinct products X and Y. Join X_j and Y_ℓ by an edge if their rows intersect and their columns intersect, i.e., if $(\rho_1 \leq \rho_3 \leq \rho_2 \vee \rho_3 \leq \rho_1 \leq \rho_4) \wedge (c_1 \leq c_3 \leq c_2 \vee c_3 \leq c_1 \leq c_4)$, where $\vee$ indicates *logical OR* and $\wedge$ indicates *logical AND*.

A maximum-weight independent set of size m in the above network corresponds to an optimal solution to subproblem *SP2*. Such a network can easily be extended beyond four products. Appendix D.1 presents a complete example of using an *MWIS* problem to find an optimal display.

After the network is created, the *MWIS* problem for solving *SP2* can be formulated as a binary integer program as follows. Suppose the network has Q nodes, and each node q has weight w_q. Let $\mathcal{E}$ be the set of edges in the network. Define binary variables x_q such that $x_q = 1$ if node q is in the independent set, $x_q = 0$ otherwise. Thus, the *MWIS* can be found by solving

$$(\textit{BIP-MWIS}) \qquad \text{Maximize} \sum_{q=1}^{Q} w_q x_q$$

$$\text{subject to}$$

$$\begin{aligned} x_p + x_q &\leq 1, \ \forall (p,q) \in \mathcal{E}, \qquad (8) \\ x_q &\in \{0,1\}, \ q = 1,2,\ldots,Q. \end{aligned}$$

Each node q applies to a particular product d and a specific rectangular presentation of that product. That $x_q = 1$ for only one of the nodes representing product d is enforced by Constraint (8) and the structure of the network.

Efficient heuristics to find a maximum-weight independent set in a network are available in the literature (examples include Hifi 1997, Sakai et al. 2003, Balaji et al. 2009, Gamarnik et al. 2009, Pelillo 2009). Our calculations were performed with a recently published exact algorithm (Trukhanov et al. 2013). Even though we use an exact algorithm for *MWIS*, our solution for Problem P could have total weighted revenue less than the optimum. This is because our decomposition into two subproblems separates the assignment of products *to* cabinets (*SP1*) from the arrangement of those products *within* cabinets (*SP2*), so not all possible displays are considered. However, our numerical results (see Table 10 in Appendix E.1) show that this loss is less than 5%.

Another concern is that the assignment of products to cabinets by the solution to subproblem *SP1* could make subproblem *SP2* infeasible. That is, it may not be possible to arrange each of the product's units so that (i) they form contiguous rectangles, (ii) the upper and lower bounds are respected, and

(iii) all slots of the cabinet are filled. No such infeasibilities occurred in any of our computational experiments, but consider the following example:

Example 2: Suppose *MIP-SP1* assigned three products to cabinet k with $R_k = 5$ and $C_k = 3$, the vector of upper bounds is $\vec{u} = (7, 7, 2)$, and the vector of lower bounds is $\vec{\ell} = (2, 2, 2)$. Even though $\sum_{d=1}^{3} \ell_d \leq R_k C_k \leq \sum_{d=1}^{3} u_d$, these three products cannot be arranged in contiguous rectangles so that the bounds are respected and all slots of the cabinet are filled (see Figure 6(a)).

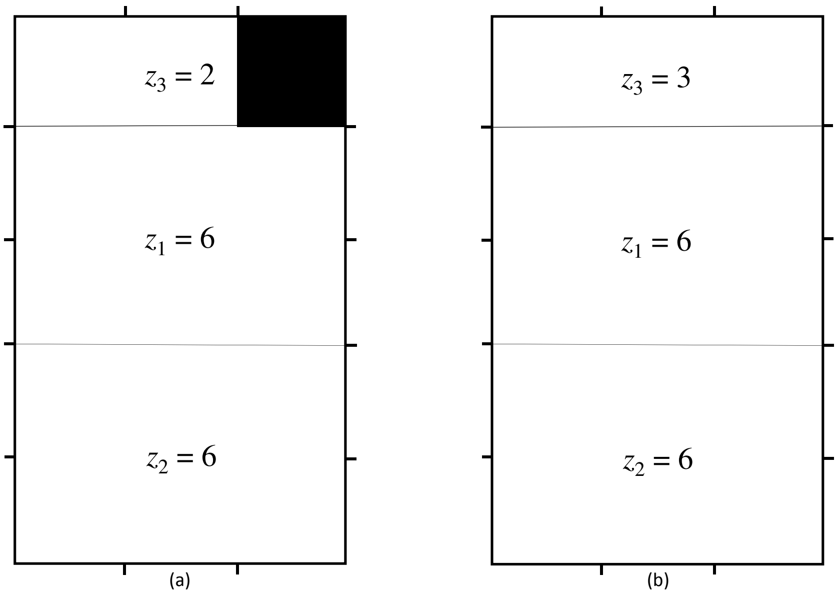

Figure 6: Two displays for Example 3: display (a) demonstrates the assignment's infeasibility. Display (b) is created after applying Algorithm Fill Slots.

Because shelf-space is "the retailers scarcest resource" (Bultez and Naert 1988, Lim et al. 2004), when *SP2* is infeasible, we choose to enforce the requirements to fill each slot and to display each product in a contiguous rectangle, but to violate the bounds. For some $g \in \{0, \ldots, m\}$, we can generate a display in which the first g titles occupy an integer number of rows, and the remaining $m - g$ titles each occupy at most C_k slots within a single row. See Figure 6(b). Algorithm Fill Slots in Appendix D.2 presents the details of a process that generates such a display.

5 Computational Results

We evaluate the performance of the proposed model by using data from a Blockbuster store to compute revenue potentials r_d used to create displays. We then calculate the percentage gain in total weighted revenue from using our method by comparing it to current practice. We begin by describing parameter estimation, which includes Blockbuster's current process for forecasting demands (hence, revenue potentials) for individual titles. Results from performance testing for generated data (1800 instances) can be found in Appendix E.

5.1 Parameter Estimation

Given the titles' compressed life cycles, Blockbuster could calculate their revenue potentials each week and use them to generate a new display. Since all new releases have the same price, forecasting the titles' demands is sufficient. This calculation could be based on environmental data (theatrical sales, performance of similar titles, studio, genre, MPAA ratings, age, etc.) and first period sales (which provide information on the store's local market), according to the model of Chung (2010) and Chung et al. (2012). This model to forecast DVD rentals is based on the Bass (1969) model and its extensions to stochastic settings by Niu (2002, 2006). The model decomposes the total demand into three categories. The purchases of the first group, called *committed buyers*, follow an exponential decay. These customers have already made up their minds to rent the DVD, so this group's sales are highest in the first week and decay exponentially over time. The second group, *potential buyers*, are later adopters; their consumption pattern is assumed to follow the Bass diffusion model (1969). The third segment is attributed to social networking, i.e., word-of-mouth recommendations (for DVDs), or to re-rents (for video games). This group exhibits a pattern in which there are no rentals in the first period, a spike in the second period, and then a slow decline from a combination of exponential decay and diffusion. Chung (2010) and Chung et al. (2012) tested this model against the actual demand for 352 titles and found it to be remarkably accurate (the average R^2 is 0.96). Hence, we assume that each title's revenue potential is known and is an input to the current study. We test our model's sensitivity to changes in the revenue potentials at the end of this section and in Appendix E.2.

We base our displays on the titles' revenue potentials by using them to form upper bounds on the number of discs displayed for each title. We have previously seen that title d's proportion of the display's total slots is determined by r_d's proportion of the total of all potentials. To scale this to the total size of the display, the proportion is multiplied by the number of slots ($\sum_{k=1}^{K} R_k C_k$) in the entire display, and the result is rounded up:

$$u_d = \left\lceil \frac{r_d}{\sum_{h=1}^{D} r_h} \sum_{k=1}^{K} R_k C_k \right\rceil, \ d = 1, \ldots, D.$$

The rounding provides some slack that is needed to ensure feasibility.

For a lower bound, we follow van Nierop et al. (2008) by using $\ell_d = 2$, $d = 1, \ldots, D$, in our computations. Other values may be chosen, provided that $\sum_d \ell_d \leq \sum_{k=1}^{K} R_k C_k$.

We have previously seen that Blockbuster assigns titles with higher revenue potentials to the fourth and fifth rows, and the potentials decrease for those titles put in shelves farther from these middle ones. In our calculations, the values for effectiveness of the rows, based on the judgement of managers, are $a_1 = a_8 = 5$, $a_2 = a_7 = 6$, $a_3 = a_6 = 8$, and $a_4 = a_5 = 10$. Sensitivity analysis performed on these values shows that the actual a_i values matter little, as long as they are in this order (see Appendix E.2).

5.2 Results from Observed Data

We gathered data at two-week intervals over an eight-week period at a Blockbuster store in Texas. This store used Chung et al. (2012)'s method for forecasting demands when determining its displays. Thus, the number of discs displayed for each title directly reflects that title's demand and, hence, its revenue potential. The number of titles considered varied from visit to visit: 179, 163, 580, 541, 498. This large variation occurred because in the early weeks, the store had many more titles that occupied at least one-half of a cabinet (15, compared to 5 in later weeks), thereby reducing the space available for other titles. Since 75-85% of these remaining titles have only one or two facings in an average display, far fewer titles could be displayed in the early weeks. These were managerial decisions based on week-to-week changes in the titles available to the market. To moderate these fluctuations, we limited our computational study to only those titles that had at least two discs in a particular week's display.

Table 5 presents the relative improvement in weighted revenue realized from using our new method. Our proposed model showed an improvement in performance ranging from 2.82% to 4.77%, with an average improvement of 3.72%.

	14-May	**28-May**	**11-Jun**	**25-Jun**	**9-Jul**	**average**
titles	179	163	579	541	498	
improvements	4.77%	2.82%	4.73%	3.27%	3.00%	3.72%

Table 5: Relative improvement of our solution procedure over current practice for observed data.

As a further measure of the quality of our procedure, we compare its performance to upper bounds derived from (*IP-UB*), a binary integer program that solves a relaxed version of problem P (see Appendix B.2). This integer program assigns each disc to a specific location, but does not impose the contiguous rectangle constraint. Because (*IP-UB*) requires excessive computational time, we use its linear programming relaxation to find an upper bound on the optimum weighted revenue for the two smaller datasets (i.e., we use a linear relaxation of a solution procedure for the relaxed problem). The total weighted revenues generated by our algorithm were within 1% of optimum, even though the upper bounds are not necessarily tight.

Results from experiments run on randomly generated data are presented in Appendix E.1. They show an average improvement of 13.66%, with weighted revenues averaging within 5% of optimum.

We use sensitivity analysis to examine the potential loss from incorrect estimates of the revenue potentials. The details of our methodology are given in Appendix E.2. Over the 130 cabinets with more than one title, the average loss for 10%, 15%, and 20% estimation errors were 0.17%, 0.49%, and 0.34% of revenue, respectively. Hence, our method is robust with respect to revenue potentials.

6 Model Extensions

First, we explain how our model applies to display advertising, webpages, and mobile devices. Next, we demonstrate how it can be augmented to allow for interactions between products, i.e., cross-product elasticities for displaying a pair of products in the same cabinet and for displaying a pair in the same row. We then show how the model can be applied to displays of products with different shapes and sizes.

6.1 Other Environments

Other venues in which businesses frequently need to allocate contiguous rectangular spaces to different items include designing retail flyers, advertising on a webpage, and managing content on mobile devices. Retailers send flyers in various formats (free-standing inserts, door-to-door, and direct mail) to millions of households each week, in addition to run-of-press ads, to advertise their current specials, so they must allocate space to different products (see Figure 7) to maximize a flyer's weighted revenue and to satisfy their suppliers' requirements. By the same token, firms allocate rectangular areas when presenting different types of content on webpages (see Figure 8) or on mobile devices such as iPads and iPhones, as seen in FlipBoard, an application for iPad.

Just like in the retail shelf-space problem, the square inches allocated to a particular product's ad can be arranged into various aspect ratios. Similarly, the limited advertising space on websites is an

Figure 7: An example layout of a flyer for a national grocery chain.

important source of revenue. These ads are typically placed in rectangular shapes with width and tallness measured in pixels. The Internet Advertising Bureau (2014) publishes standard layouts for online ads. Examples include (i) *Leaderboard*, 728 pixels wide by 90 pixels high, (ii) *Button 2*, 120 pixels wide by 60 pixels high, (iii) *Billboard*, 970 pixels wide by 250 pixels high, (iv) *Filmstrip*, 300 pixels wide by 600 pixels high, etc. Website owners are paid by the number of clicks on the ad, so the quality of the layout has a direct effect on revenues.

Our conceptual model can easily be applied to flyer displays and webpage designs for the following reasons. Choosing on which page of a flyer each product's ad appears is equivalent to allocating products to different cabinets. Arranging rectangular ads within a page is equivalent to arranging products into contiguous rectangles that together cover the entire cabinet. Hence, z_d^k here represents the number of square inches of display allocated to product d's ad on page k of the flyer or insert. Alphabetization of items (*SP3*) is generally not an issue in these types of applications.

We know of no prior research on allocating spaces in the context of retail flyer design, but Pieters et al. (2007) provide an interesting complement to our work. They investigate how design elements and competitive clutter affect consumers' attention to feature ads. Hence, their work measures the individual feature ads' effectiveness that arises from characteristics of their *contents*, whereas ours maximizes the effectiveness of the total display based on *where* within the display each ad is placed and the total *size* of each ad. Adler et al.'s (2002) study of webpage ads briefly considers space allocation for two-dimensional ads. These ads' sizes are restricted to be divisible in each dimension: product d's ad has

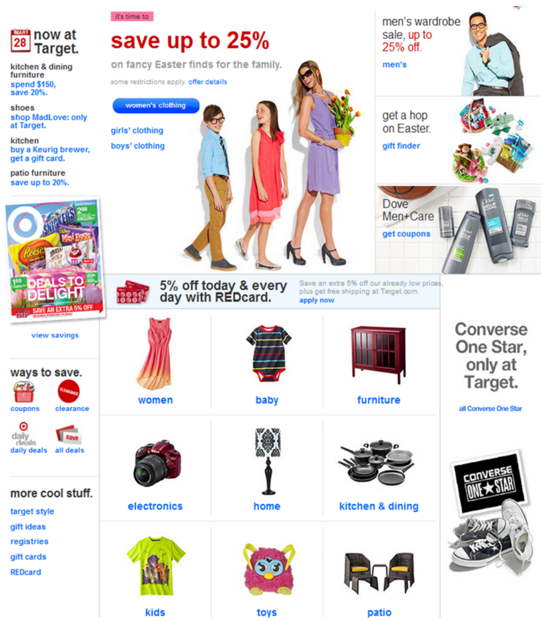

Figure 8: An example layout of a webpage for a national retailer.

dimensions $H_d \times V_d$ such that H_d is an integer multiple of H_{d+1}, and V_d is an integer multiple of V_{d+1}, for $d = 1, \ldots, D$. Early work on two-dimensional bin-packing was done by Baker et al. (1980), Coffman et al. (1980), and Baker and Schwarz (1983).

6.2 Product Interactions

We now demonstrate how the effect of product interactions can be incorporated into our two subproblems.

<u>Subproblem *SP1*</u>

Let π_{de} be a *cabinet affinity factor* that quantifies the benefit of having products d and e in the same cabinet. Its value is assigned by the manager and is predetermined as in Lim et al. (2004). The additional profit realized by putting z_d^k units of product d and z_e^k units of product e both in cabinet k is $\pi_{de} \times \min\{z_d^k, z_e^k\}$. We implement this scheme by updating the objective of *MIP-SP1* to

$$\text{maximize} \sum_{d=1}^{D} \sum_{k=1}^{K} r_d z_d^k + \sum_{k=1}^{K} \sum_{d=1}^{D-1} \sum_{e=d+1}^{D} \pi_{de} y_{de}^k,$$

where the value of $y_{de}^k = \min\{z_d^k, z_e^k\}$ is set by adding the following three constraints to *MIP-SP1*:

$$\begin{aligned}
y_{de}^k &\leq z_d^k, \quad d = 1, \ldots, D-1;\ e = d+1, \ldots, D;\ k = 1, \ldots, K, \\
y_{de}^k &\leq z_e^k, \quad d = 1, \ldots, D-1;\ e = d+1, \ldots, D;\ k = 1, \ldots, K, \\
y_{de}^k, z_e^k &\geq 0,\ z_e^k \text{ integer}, \quad d = 1, \ldots, D, \quad e = 1, \ldots, D, \quad k = 1, \ldots, K.
\end{aligned}$$

The previous constraint set for *MIP-SP1* (2) - (7) remains the same.

Subproblem *SP2*

In the binary integer program (*BIP-MWIS*) that finds a maximum weight independent set, each node q implies a value for the number of units of product d that are displayed in row i. Denote this constant by θ_{id}^{q}. Thus, the number of units of product d displayed in row i is $\sum_{q=1}^{Q} \theta_{id}^{q} x_q$.

We can now update (*BIP-MWIS*) to allow for product interactions on rows in much the same way as was done for cabinets. Let ψ_{de} be a *row affinity factor* that quantifies the benefit of having products d and e in the same row of the same cabinet. Thus, for cabinet k, the following mixed integer program solves the *MWIS* that arranges its products (recall that *MIP-SP1* assigns m products to cabinet k):

$$\text{Maximize} \qquad \sum_{q=1}^{Q} w_q x_q + \sum_{i=1}^{R_k} \sum_{d=1}^{m-1} \sum_{e=d+1}^{m} \psi_{de} v_{de}^{i},$$

subject to

$$\begin{aligned}
x_p + x_q &\leq 1, \ \forall (p,q) \in \mathcal{E}, \\
v_{de}^{i} &\leq \sum_{q=1}^{Q} \theta_{id}^{q} x_q, \quad d = 1,\ldots,m-1; \ e = d+1,\ldots,m; \ i = 1,\ldots,R_k, \\
v_{de}^{i} &\leq \sum_{q=1}^{Q} \theta_{ie}^{q} x_q, \quad d = 1,\ldots,m-1; \ e = d+1,\ldots,m; \ i = 1,\ldots,R_k, \\
x_q &\in \{0,1\}, \ q = 1,2,\ldots,Q, \\
v_{de}^{i} &\geq 0, \quad d = 1,\ldots,m-1; \ e = d+1,\ldots,m; \ i = 1,\ldots,R_k.
\end{aligned}$$

6.3 Products with Different Dimensions

We now consider a generalization of our model in which each product has its own dimensions. The units in which these dimensions are measured are kept unspecified for generality. We start with Subproblem *SP1*. The upper bounds (u_d) and lower bounds (ℓ_d) refer to the number of units, not the amount of area covered, so constraints relating to them do not require changes. Similar reasoning applies to the objective in *MIP-SP1*. The only change to the constraints is that (3) is replaced by (9). Two parameters are added:

Parameters:

V_d, H_d: vertical and horizontal dimensions of product d, $d = 1,\ldots,D$.

$$\sum_{d=1}^{n} V_d H_d z_d^{k} \ \leq \ R_k C_k, \quad k = 1,\ldots,K. \tag{9}$$

Constraints (9) ensure that the amount of space allocated to each cabinet does not exceed the cabinet's capacity. They do not require that the length and width combinations allow for rectangles that can be formed into a feasible display. That requirement is considered next in the update to Subproblem *SP2*.

As before, we use the number of units of each product that is specified by the solution to *SP1*, plus some slack that may be required to meet the contiguous rectangle requirement while filling each slot. Thus, consider all feasible rectangles that use $\max\{\ell_d, z_d^k - 1\}$, z_d^k, or $\min\{u_d, z_d^k + 1\}$ units of product d. For example, a product for which $\ell_d = 2$, $z_d = 9$, and $u_d = 9$ would have three possible rectangles of units: 2×4, 3×3, 4×2, and the amount of space $L_d \times T_d$ covered by these is $2V_d \times 4H_d$, $3V_d \times 3H_d$, $4V_d \times 2H_d$, respectively.

Given product d's possible rectangle with length L_d and tallness T_d, its top-left slot can be in any row between 1 and $R_k - T_d + 1$ and in any column from 1 to $C_k - L_d + 1$. To illustrate, if eight units of product d are displayed in a 2×4 rectangle, then $L_d \times T_d = 2V_d \times 4H_d$. If this rectangle is within cabinet k and its top-left slot is (a, b), then its bottom right slot is $(a + 2V_d - 1, b + 4H_d - 1)$, assuming $a + 2V_d - 1 \leq R_k$ and $b + 4H_d - 1 \leq C_k$. With these limitations, the network for the maximum weight independent set problem can be constructed as before.

7 Conclusion and Suggestions for Future Research

This study represents the first analysis of retail shelf-space allocation for a two-dimensional display that coordinates the multiple shelves of a product's presentation. Our novel formulation creates displays in which each product is arrayed in a contiguous rectangle. This provides the display designer with more flexibility than does our client's current process by allowing a greater variety in the aspect ratios and in the number of units displayed for each product. Additionally, the proposed solution can also be applied to retail flyer design and to webpage advertising layout. All previous studies, even those that recognized the value of a display's vertical location, do not consider that the tallness (number of shelves) of a product's display may be as important as its length (linear feet of shelf-space).

We have shown that our proposed methods for implementing a new display policy at a world-wide retailer of DVDs can increase revenue significantly. Furthermore, we also show in small-scale problems that the total weighted revenues of the generated displays are very close to optimal. Our network formulation enables the solution procedure to scale efficiently to larger implementations.

Since ours is the first analysis of such two-dimensional displays, there are several interesting practically-relevant questions and directions available for future investigation. We propose the following:

- Previous studies allocate shelf-space based on space elasticities, product complementarities, and substitutes. None of these issues has been studied in the DVD market since they cannot be reliably tested in this fast-changing market that varies from store to store. It would be an interesting exercise to see how these aspects can be included with our proposed solution so that the method can be applied to a grocery store shelf allocation problem. Section 6 provides guidelines for extending our model to include product interactions.

- Should there be additional restrictions on the allowable aspect ratios of the rectangular displays of particular products? We suspect that tall, skinny displays are not as effective as short, wide ones, and that neither is as effective as square displays. Data from webpage advertisements may be useful for such a study.

- Displays for products with different dimensions could be studied more thoroughly, based on the structure outlined in Section 6.

- The depth of facings is not considered in our study. This topic is not relevant to the applications that we focused on, such as DVD rentals or web-page design. In fact, it is considered in very few published papers on shelf-space. Models may be developed to incorporate this element in future studies.

References

Adler, M. Gibbons, P., Matias, Y. 2002. Scheduling space-sharing for internet advertising. *Journal of Scheduling* **5** 103-119.

Anderson, E.E., H.N. Amato. 1974. A Mathematical Model for Simultaneously Determining the Optimal Brand-Collection and Display-Area Allocation. *Operations Research* **22**(1) 13-21.

Bai, R., G. Kendall. 2008. A Model for Fresh Produce Shelf-Space Allocation and Inventory Management with Freshness-Condition-Dependent Demand. *INFORMS Journal on Computing* **20**(1) 78-85.

Bai, R., T. van Woensel, G. Kendall, E.K. Burke. 2012. A New Model and a Hyper-heuristic Approach for Two-dimensional Shelf-Space Allocation. *4OR : A Quarterly Journal of Operations Research* **11**(1) 31-55.

Baker, B.S., E.G. Coffman, R.L. Rivest. 1980. Orthogonal Packings in Two Dimensions. *SIAM Journal on Computing* **9**(4) 846-855.

Baker, B.S., J.S. Schwarz. 1983. Shelf Algorithms for Two-Dimensional Packing Problems. *SIAM Journal on Computing* **12**(3) 508-525.

Balaji, S., V. Swaminathan, K. Kannan. 2009. Approximating Maximum Weighted Independent Set Using Vertex Support. *World Academy of Science, Engineering and Technology I: Mathematical and Computational Sciences* **4**(4) 211-216.

Bass, F. M. 1969. A New Product Growth Model for Consumer Durables. *Management Science* **15** 215-227.

Berlyne, D.E. 1958. The Influence of Complexity and Novelty in Visual Figures on Orienting Responses. *Journal of Experimental Psychology* **55**(3) 289-296.

Bhattacharya, S., W. Comerford. 2006. DVD Supply Chain: The Emerging Challenge to Media Profitability. Accessed at `http://www.infosys.com/industries/media-entertainment/white-papers/Documents/DVD-supply-chain.pdf` on February 25, 2013.

Borin, N., P. W. Farris, J. R. Freeland. 1994. A Model for Determining Retail Product Category Assortment and Shelf Allocation. *Decision Sciences* **25**(3) 359-384.

Bultez, A., P. Naert. 1988. SH.A.R.P.: Shelf Allocation For Retailers' Profit. *Marketing Science* **7**(3) 211-231.

Chung, C. 2010. Demand-forecasting, Product-allocation, and Production-scheduling Applications at Blockbuster, Ph.D Thesis, University of Texas at Dallas.

Chung, C., M. Dawande, C. Sriskandarajah. 2011. A Short-Range Scheduling Model for Blockbuster's Order Processing Operation, *Interfaces* **41**(5) 466-484.

Chung, C., S.C. Niu, C. Sriskandarajah. 2012. A Sales Forecast Model for Short-Life-Cycle Products: New Releases at Blockbuster, *Production and Operations Management* **21**(5) 851-873.

Coffman, E.G., M.R. Garey, D.S. Johnson, R.E. Tarjan. 1980. Performance Bounds for Level-Oriented Two-Dimensional Packing Algorithms. *SIAM Journal on Computing* **9**(4) 808-826.

Corstjens, M., Doyle, P. 1981. A Model for Optimizing Retail Space Allocations. *Management Science* **27**(7) 822-833.

Corstjens, M., Doyle, P. 1983. A Dynamic Model for Strategically Allocating Retail Space. *The Journal of the Operational Research Society* **34**(10) 943-951.

Curhan, R.C. 1972. The Relationship between Shelf Space and Unit Sales in Supermarkets. *Journal of Marketing Research* **9**(4) 406-412.

Desmet, P., V. Renaudin. 1998. Estimation of Product Category Sales Responsiveness to Allocated Shelf Space. *International Journal of Research in Marketing* **15** 443-457.

Drèze, X., S.J. Hoch, M.E. Purk. 1994. Shelf Management and Space Elasticity. *Journal of Retailing* **70**(4) 301-326.

Gamarnik, D., D. Goldberg, T. Weber. PTAS for Maximum-Weight Independent Set Problem with Random Weights in Bounded Degree Graphs. In *Proceedings of 21-st ACM-SIAM Symposium on Discrete Algorithms, (SODA)*, 2010.

Hansen, J.M., S. Raut, S. Swami. 2010. Retail Shelf Allocation: A Comparative Analysis of Heuristic and Meta-Heuristic Approaches. *Journal of Retailing* **86**(1) 94-105.

Hifi, M. 1997. A Genetic Algorithm-Based Heuristic for Solving the Weighted Maximum Independent Set and Some Equivalent Problems. *The Journal of the Operational Research Society* **48**(6) 612-622.

Hwang, H., B. Choi, M.-J. Lee. 2005. A Model for Shelf-Space Allocation and Inventory Control Considering Location and Inventory Level Effects on Demand. *International Journal of Production Economics* **97** 185-195.

Internet Advertising Bureau. 2014. IAB Display Advertising Guidelines. Accessed at `http://www.iab.net/guidelines/508676/508767/displayguidelines` on July 17, 2014.

Laya, P. 2014. Blockbuster Is Still a Hit—South of the Border. Business Week. Accessed at `http://www.businessweek.com/articles/2014-02-06/in-mexico-blockbuster-video-chain-lives-on`. on February 12, 2014.

Lim, A., B. Rodrigues, X. Zhang. 2004. Metaheuristics with Local Search Techniques for Retail Shelf-Space Optimization. *Management Science* **50**(1) 117-131.

Murray, C. C., D. Talukdar, A. Gosavi. 2010. Joint Optimization of Product Price, Display Orientation and Shelf-Space Allocation in Retail Category Management. *Journal of Retailing* **86**(2) 125-136.

Niu, S.-C. 2002. A Stochastic Formulation of the Bass Model of New-Product Diffusion. *Mathematical Problems in Engineering* **8** 249-263.

Niu, S.-C. 2006. A Piecewise-Diffusion Model of New-Product Demands. *Operations Research* **54** 678-695.

Pelillo, M. 2009. Heuristics for Maximum Clique and Independent Set, in *Encyclopedia of Optimization*, Christodoulos A. Floudas and Panos M. Pardalos, eds. 1508-1520.

Pieters, R., M. Wedel, R. Batra. 2010. The Stopping Power of Advertising: Measures and Effects of Visual Complexity. *Journal of Marketing* **74** 48-60.

Pieters, R., M. Wedel, J. Zhang. 2007. Optimal feature advertising design under competitive clutter. *Management Science* **53**(11) 1815-1828.

Sakai, S., M. Togasaki, K. Yamazaki. 2003. A Note on Greedy Algorithms for the Maximum Weighted Independent Set Problem. *Discrete Applied Mathematics* **126** 313-322.

Sawhney, M.S., J. Eliashberg. 1996. A Parsimonious Model for Forecasting Gross Box-Office Revenues of Motion Pictures. *Marketing Science* **15**(2) 113-131.

Trukhanov, S., C. Balasubramaniam, B. Balasundaram, S. Butenko. 2013. Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations. *Computational Optimization and Applications* **56** 113-130.

Van Nierop, E., D. Fok, P.H. Franses. 2008. Interaction between Shelf Layout and Marketing Effectiveness and its Impact on Optimizing Shelf Arrangements. *Management Science* **27**(6) 1065-1082.

Yang, M.-H. 2001. An Efficient Algorithm to Allocate Shelf Space. *European Journal of Operational Research* **131** 107-118.

Yang, M.-H., W.-C.Chen. 1999. A Study on Shelf-Space Allocation and Management. *International Journal of Production Economics* **60-61** 309-317.

Zufryden, F.S. 1986. A Dynamic Programming Approach for Product Selection and Supermarket Shelf-Space Allocation. *Journal of the Operational Research Society* **37**(4) 413-22.

Zytkowska, A. 2003. *A Micro-Marketing Model for Optimizing Shelf Space Allocation.* Master's Thesis, Tilburg University, The Netherlands.

Supporting Information

Additional supporting information may be found in the online version of this article:

Appendix A: Proofs of Theoretical Results

Appendix B: Mixed Integer Linear Program Formulations

Appendix C: Subproblem *SP3*: Improve Alphabetical Ordering

Appendix D: Subproblem *SP2*

Appendix E: Results for Generated Data

Online Appendix for "Maximizing Revenue Through Two-Dimensional Shelf-Space Allocation"

A Proofs of Theoretical Results

Theorem 1 *The display problem P is strongly NP-hard.*

Proof of Theorem: Consider an arbitrary instance of 3-PARTITION (Garey and Johnson 1979):
3-PARTITION: Given $B \in \mathbb{Z}^+$, a set $A = \{z_1, z_2, \ldots, z_{3t}\}$, $z_i \in \mathbb{Z}^+$ and $B/4 < z_i < B/2$ for $i = 1, \ldots, 3t$, and $\sum_{i=1}^{3t} z_i = tB$, does there exist a partition of A into disjoint subsets $A_1, A_2, \ldots, A_t$ such that $|A_k| = 3$ and $\sum_{z_i \in A_k} z_i = B$ for $k = 1, \ldots, t$?

Given an instance of 3-PARTITION, we construct a specific instance of the decision problem version of problem P as follows: $K = t$, $R_k = 1$, $C_k = B$, for $k = 1, \ldots, K$, $a_1 = 1$. Consider a set of $D = 3t$ products $\mathcal{P} = \{P_d : d = 1, \ldots, 3t\}$, in which the upper bound, lower bound, and revenue potential of product P_d are $u_d = z_d$, $\ell_d = 1$, and $r_d = B$, respectively, $d = 1, \ldots, 3t$.

Decision Problem: Does there exist a display σ such that the total weighted revenue of the display satisfies $F_\sigma \geq tB^2$?

The decision problem belongs to class NP. Also, it is easy to verify that the construction of the decision problem can be done in polynomial time $O(tB)$. We now show that there is a display σ of D products in K cabinets such that $F_\sigma \geq tB^2$ if and only if there exists a solution to the 3-PARTITION problem.

If Part: Suppose there exists a 3-Partition. Without loss of generality, we may assume $z_{3k-2} + z_{3k-1} + z_{3k} = B$, $k = 1, \ldots, t$. Let P_{3k-2}, P_{3k-1}, P_{3k}, $k = 1, \ldots, t$, denote products corresponding to partition elements. We can create a display σ that has $F_\sigma = tB^2$ by assigning products P_{3k-2}, P_{3k-1}, P_{3k} to cabinet k, for $k = 1, \ldots, t$.

Only If Part: Suppose there exists a feasible display σ_o of the products in $\mathcal{P}$ across t cabinets such that $F_{\sigma_o} \geq tB^2$. This implies that all slots of the display be filled with one item from $\mathcal{P}$. The restriction $B/4 < u_d = z_d < B/2$ implies that each cabinet holds exactly three items. ∎

Theorem 2 *Problem SP2 is NP-hard in the ordinary sense.*

Proof of Theorem 2: Consider an arbitrary instance of PARTITION (Garey and Johnson 1979):
PARTITION: Given $B \in \mathbb{Z}^+$, a set $A = \{z_1, z_2, \ldots, z_t\}$, $z_i \in \mathbb{Z}^+$, and $\sum_{i=1}^{t} z_i = 2B$, does there exist a partition of A into two disjoint subsets A_1, A_2 such that $\sum_{z_i \in A_1} z_i = B$?

Given an instance of PARTITION, we construct a specific instance of the decision problem version of *SP2* as follows: $R_k = 2$, $C_k = B$, $a_1 = a_2 = 1$. Consider a set of $m = t$ products $\mathcal{P} = \{P_d : d = 1, \ldots, t\}$, and the upper bound, lower bound, and revenue potential of product P_d are $u_d = z_d$, $\ell_d = 1$, and $r_d = 1$, respectively, $d = 1, 2, \ldots, t$;

Decision Problem: Does there exist a display σ such that the total weighted revenue of the display satisfies $F'_\sigma \geq 2B$?

The decision problem belongs to class NP. Also, it is easy to verify that the construction of the decision problem can be done in polynomial time $O(t)$. We now show that there is a display σ of m products in a cabinet such that $F'_\sigma \geq 2B$ if and only if there exists a solution to the PARTITION problem.

If Part: Suppose there exists a Partition. Without loss of generality, we may assume $z_1+z_2+\cdots+z_q = B$. Let P_1, P_2, ..., P_t, denote products corresponding to partition elements. Placing items $P_1, \ldots, P_q$ into row 1 and items $P_{q+1}, \ldots, P_t$ into row 2 generates a feasible display σ with $F'_\sigma = 2B$.

Only If Part: This is similar to the corresponding part of the proof for Theorem 1. If there exists a feasible display σ_o of $m = t$ products in a cabinet such that $F'_{\sigma_o} \geq 2B$, then all B slots in each row must be filled. Hence, there exists a solution to the PARTITION problem. ■

B Mixed Integer Linear Program Formulations

B.1 Formulation of Problem P (Section 3)

The following mixed integer linear program solves problem P. We begin by defining variables. The parameters are as defined in Section 3.

$x_d^k = 1$, if product d is allocated to cabinet k; $x_d^k = 0$, otherwise.
$y_{id}^k = 1$, if any units of product d are assigned to row i of cabinet k; $y_{id}^k = 0$, otherwise.
z_d^k: the number of units of product d allocated to cabinet k.
Γ_{id}^k: the number of elements of product d in row i of cabinet k.

$$\text{maximize} \sum_{k=1}^{K} \sum_{d=1}^{D} \sum_{i=1}^{R_k} a_i r_d \Gamma_{id}^k \tag{10}$$

subject to

$$\sum_{k=1}^{K} x_d^k = 1, \quad d = 1, \ldots, D \tag{11}$$

$$\sum_{i=1}^{R_k} \Gamma_{id}^k = z_d^k, \quad d = 1, \ldots, D; \; k = 1, \ldots, K \tag{12}$$

$$x_d^k \ell_d \leq z_d^k, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K \tag{13}$$

$$z_d^k \leq x_d^k u_d, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K \tag{14}$$

$$\sum_{d=1}^{D} \Gamma_{id}^k = C_k, \quad i = 1, \ldots, R_k; \; k = 1, \ldots, K \tag{15}$$

$$
\begin{aligned}
\Gamma^k_{id} &\leq w_d, \quad i=1,\ldots,R_k;\; d=1,\ldots,D;\; k=1,\ldots,K && (16)\\
w_d - \Gamma^k_{id} &\leq C_k(1-y^k_{id}), \quad i=1,\ldots,R_k;\; d=1,\ldots,D;\; k=1,\ldots,K && (17)\\
\Gamma^k_{id} &\leq C_k y^k_{id}, \quad i=1,\ldots,R_k;\; d=1,\ldots,D;\; k=1,\ldots,K && (18)\\
\Gamma^k_{i+1,d} + w_d &\geq \Gamma^k_{id} + \Gamma^k_{i+2,d}, \quad i=1,\ldots,R_k-2;\; d=1,\ldots,D;\; k=1,\ldots,K && (19)\\
\Gamma^k_{i+1,d} + w_d &\geq \Gamma^k_{id} + \Gamma^k_{i+3,d}, \quad i=1,\ldots,R_k-3;\; d=1,\ldots,D;\; k=1,\ldots,K && (20)\\
&\vdots\\
\Gamma^k_{i+1,d} + w_d &\geq \Gamma^k_{id} + \Gamma^k_{i+h,d}, \quad i=1,\ldots,R_k-h;\; d=1,\ldots,D;\; k=1,\ldots,K && (21)\\
&\vdots\\
\Gamma^k_{i+1,d} + w_d &\geq \Gamma^k_{id} + \Gamma^k_{i+R_k-2,d}, \quad i=1,2;\; d=1,\ldots,D;\; k=1,\ldots,K && (22)\\
\Gamma^k_{2,d} + w_d &\geq \Gamma^k_{1,d} + \Gamma^k_{R_k,d}, \quad d=1,\ldots,D;\; k=1,\ldots,K && (23)\\
x^k_d,\; y^k_{id} &\in \{0,1\}, \quad i=1,\ldots,R_k;\; d=1,\ldots,D;\; k=1,\ldots,K && (24)\\
z^k_d,\; \Gamma^k_{id},\; w_d &\in \mathbb{Z}^+, \quad i=1,\ldots,R_k;\; d=1,\ldots,D;\; k=1,\ldots,K && (25)
\end{aligned}
$$

The objective (10) maximizes total weighted revenue of the display. Constraints (11) ensure that each product is displayed in exactly one cabinet. Constraints (12) link the two ways of tracking the number of units displayed for product d: the total when calculated per row (Γ^k_{id}) within a cabinet must equal the total when specified by cabinet (z^k_d). Constraints (13) and (14) enforce lower bounds and upper bounds, respectively, on the number of units allocated to a particular product. Constraints (15) ensure that exactly C_k units are displayed in each row of each cabinet. Constraints (16) and (17) set the width of product d's display and ensure that it is uniform ($\Gamma^k_{id} = 0$ or $\Gamma^k_{id} = w_d$, $\forall i, \forall d, \forall k$). Constraints (17) and (18) set the value of y^k_{id}. Constraints (19)-(23) ensure that each product's display occupies contiguous rows: if product d has units in row i and row j of cabinet k, $i+2 \leq j$, then product d has units in rows $i+1,\ldots,j-1$ of cabinet k. It follows that each product is displayed in contiguous rows and has the same number of units in each of those rows; hence, it can be displayed as a rectangle. Note that Constraints (19)-(23) can be condensed into the following:

$$\Gamma^k_{i+1,d} + w_d \;\geq\; \Gamma^k_{id} + \Gamma^k_{i+h,d}, \quad h=2,\ldots,R_k-1;\; i=1,\ldots,R_k-h;\; d=1,\ldots,D.$$

Constraints (24) require that x^k_d and y^k_{id} are binary variables. Constraints (25) ensure that the z^k_d, Γ^k_{id}, and w_d variables are non-negative integers.

B.2 An Upper Bound on Optimal Revenue

We develop binary integer program (*IP-UB*) to generate an upper bound on the optimal value of Problem P by relaxing the requirement that each product must be displayed in a contiguous rectangle. Recall that $x_d^k = 1$ if product d is allocated to cabinet k; $x_d^k = 0$, otherwise. Let $y_{dij}^k = 1$ if a unit of product d is allocated to slot (i,j) of cabinet k; $y_{dij}^k = 0$ otherwise.

$$(\textit{IP-UB}) \quad \text{maximize} \sum_{d=1}^{D} r_d \sum_{k=1}^{K} \sum_{i=1}^{R_k} a_i \sum_{j=1}^{C_k} y_{dij}^k, \tag{26}$$

subject to

$$\sum_{k=1}^{K} x_d^k = 1, \quad d = 1, \ldots, D, \tag{27}$$

$$\sum_{d=1}^{D} y_{dij}^k = 1, \quad i = 1, \ldots, R_k, \ j = 1, \ldots, C_k, \ k = 1, \ldots, K, \tag{28}$$

$$x_d^k \ell_d \le \sum_{i=1}^{R_k} \sum_{j=1}^{C_k} y_{dij}^k, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K, \tag{29}$$

$$\sum_{i=1}^{R_k} \sum_{j=1}^{C_k} y_{dij}^k \le x_d^k u_d, \quad d = 1, \ldots, D, \quad k = 1, \ldots, K, \tag{30}$$

$$x_d^k \in \{0,1\}, \ y_{dij}^k \in \{0,1\}, \ i = 1, \ldots, R_k, \ j = 1, \ldots, C_k, \ k = 1, \ldots, K, \ d = 1, \ldots, D. \tag{31}$$

The objective (26) represents the total weighted revenue of the display. Constraints (27) ensure that all of the units for a specific product will be stored in exactly one cabinet. Constraints (28) ensure that each slot will carry exactly one unit. Constraints (29) and (30) enforce lower bounds and upper bounds, respectively, on the number of slots allocated to the units of a particular product. Constraints (31) ensure that the variables are binary.

C Subproblem *SP3*: Improve Alphabetical Ordering

Blockbuster displays new releases in alphabetical order, where possible. Soups and spices are also products of uniform shape that are displayed in alphabetical order. We now present a method for creating such arrangements with a minimal reduction in the total weighted revenue generated by solving Subproblems 1 and 2.

The two-dimensional nature of our model precludes true alphabetization because the products' units are displayed in rectangles of differing quantities, differing aspect ratios, and differing vertical locations based on revenue potentials. Thus, alphabetic ordering cannot be applied within a cabinet. Instead, we attempt to arrange the products into *approximately alphabetical order*, which we define to mean that the products in a given cabinet are each alphabetically later than all those in the preceding cabinet and alphabetically earlier than those in the succeeding cabinet.

Process *Alpha-Cab* modifies the assignment of products to cabinets without changing the number of units displayed for each product or the number of products in each cabinet. The process exchanges the locations of two products in different cabinets only if they have the same number of units displayed and if the exchange improves the overall display's alphabetical order. Because there is a strong correlation between a product's revenue potential and the number of its units that are displayed, there is little change to the total weighted revenue from this process.

Here are the operational details of *Alpha-Cab*. It first renumbers the cabinets so that they are in alphabetical order based on the product in each cabinet with the highest revenue potential. Next, the remaining $D - K$ products are divided into groups so that each product in a given group has the same number of units displayed. Within each such group the products are pulled from their cabinets, alphabetized, and reinserted into those cabinets in the new order (see Figure 9).

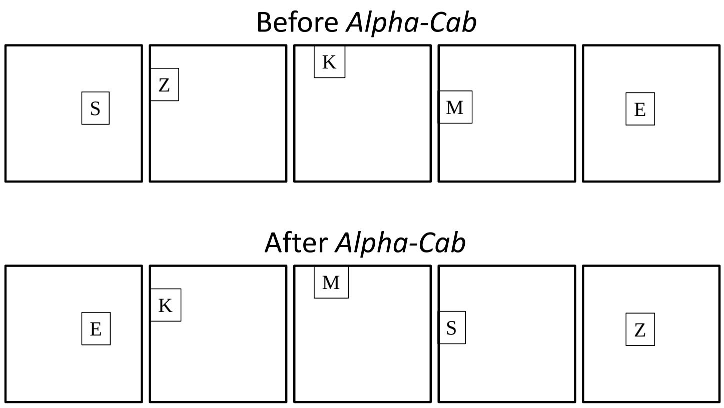

Figure 9: Demonstration of process *Alpha-Cab*.

Computational experiments verified that this process has minimal effect on the weighted revenue of the resulting display: the average difference in total weighted revenue between displays before using *Alpha-Cab* and those afterward was 0.027%. A more significant question is by how much does *Alpha-Cab* improve the display by making it more user-friendly via alphabetization. We quantified this by measuring the total deviation of a display from our definition of approximately alphabetic order. A product's *alpha value*, denoted $\alpha(d)$, is simply its numerical rank if all D products were arranged alphabetically. A product's deviation is measured by the minimum distance from its alpha value that its cabinet assignment implies (order within its cabinet is not considered because of the two-dimensional display and the differing effectiveness values of the rows). Let m_j be the number of products in cabinet j. In a perfect display, for title d in cabinet k, the title's alpha value should be less than or equal to the total number of titles in cabinets 1 through k and greater than the number of titles in cabinets 1 through $k-1$: $\sum_{j=1}^{k-1} m_j < \alpha(d) \leq \sum_{j=1}^{k} m_j$. Such a title's deviation equals zero. Formally, if product d resides in cabinet k, then product d's deviation from alphabetic order is defined as

$$\delta(d) = \max \left\{ 0,\ \sum_{j=1}^{k-1} m_j + 1 - \alpha(d),\ \alpha(d) - \sum_{j=1}^{k} m_j \right\}.$$

A display's total deviation is $\Delta = \sum_{d=1}^{D} \delta(d)$. On average, the total deviation when using *Alpha-Cab* is less than one-fifth (18.7%) of the deviation obtained when using *MIP-SP1* without it. Thus, we judge this process to be highly effective.

Alpha-Cab cannot generate infeasible solutions, but it may have no feasible actions available to it. If the display's total deviation is above an acceptable level, then *Alpha-Cab* can be repeated, but with fewer (but larger) groups. Assume that the largest number of units displayed among these groups is $2G$. Form G new groups (some may be empty), where each group g is composed of products that have either $2g-1$ or $2g$ units displayed, $g = 1, \ldots, G$. Hence, a product with $2g-1$ units can be switched with one having $2g$ units, if the amount displayed for each is adjusted so that each fits in its new space. This allows for more robust alphabetization with minimal change in the total weighted revenue and can be repeated with groups that increase in size but decrease in number until the desired total deviation is achieved.

For the generalized problem with products having different dimensions (Section 6.3), we arrange the products into approximate alphabetic order by using *Alpha-Cab* with an additional restriction. Products in different cabinets are exchanged only if they cover the same amount of area and if their dimensions ($V_d \times H_d$) allow them to fit into each other's current location. Consider three products in three separate cabinets; see Figure 10(a). Each product's display covers 24 square units:

- Product C has dimensions 2×2, is displayed in a 3×2 configuration, and resides in Cabinet 1. Product C covers a 6×4 space.

- Product A has dimensions 1×4, is displayed in a 2×3 configuration, and resides in Cabinet 2. Product A covers a 2×12 space.

- Product B has dimensions 3×2, is displayed in a 1×4 configuration, and resides in Cabinet 3. Product B covers a 3×8 space.

Product A can be switched with Product C. See Figure 10(b). Product A's new configuration in Cabinet 1 is 6×1; Product C's new configuration in Cabinet 2 is 1×6. However, Product C cannot be

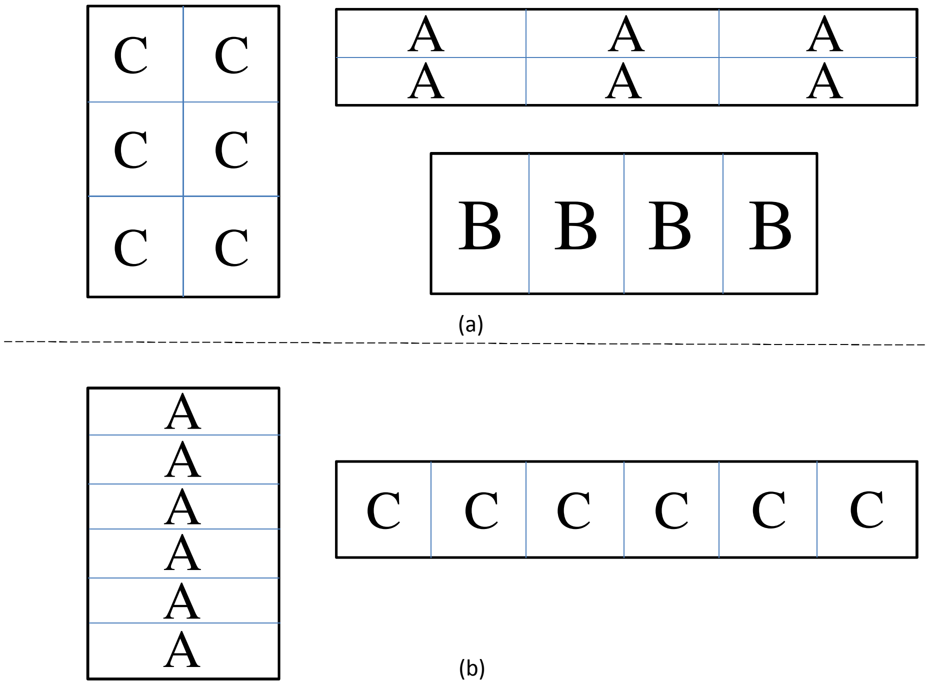

Figure 10: Products with different shapes. Products A and C can be switched, but B cannot switch with either.

switched from Cabinet 2 to Product B's location in Cabinet 3: six units of Product C cannot fit into a 3×8 space, and four units of Product B cannot fit into a 2×12 space.

D Subproblem *SP2*

This appendix presents an example of finding an optimal display by finding a maximum weight independent set. It also describes Algorithm Fill Slots.

D.1 Maximum Weight Independent Set Example

We now illustrate the formulation of the independent set problem using the following example that displays two products in a cabinet. We then present the display corresponding to the optimal solution to the independent set problem.

Suppose *MIP-SP1* assigned two products to cabinet k with $R_k = 3$ and $C_k = 3$, and that the revenue potentials, upper bounds, and lower bounds are given in Table 6. Assume that the nominal values for effectiveness of row i are $a_1 = a_3 = 1$, $a_2 = 2$. The resulting assignment is $z_1^k = 6$ and $z_2^k = 3$.

Product	d	Revenue potential (r_d)	Upper bound (u_d)	Lower bound (ℓ_d)
E	1	12	6	2
F	2	4	3	2

Table 6: Data for Example 2.

We formulate and solve an *MWIS* to find the aspect ratio and the location of each product's rectangular display. Table 7 provides all possible rectangles for the products in Example 2.

Product	d	z_d^k	All possible rectangles for each product
E	1	6	2×3, 3×2
F	2	3	1×3, 3×1, 1×2, 2×1

Table 7: All possible rectangles for products in Example 2, based on z_d^k, ℓ_d, and u_d. A rectangle with $z_1^k - 1 = 5$ units is not feasible for this 3×3 cabinet.

Product	Aspect Ratio	Possible placements in cabinet (rows, columns)	Nodes for placements
Product E: Rectangle 1	2×3	(1-2,1-3), (2-3,1-3)	E_1, E_2
Product E: Rectangle 2	3×2	(1-3,1-2), (1-3,2-3)	E_3, E_4
Product F: Rectangle 3	1×3	(1-1,1-3), (2-2,1-3), (3-3,1-3)	F_1, F_2, F_3
Product F: Rectangle 4	3×1	(1-3,1-1), (1-3,2-2), (1-3,3-3)	F_4, F_5, F_6

Table 8: All possible placements for Products E and F for Example 2 of subproblem *SP2*. Notation (ρ_1-ρ_2,c_1-c_2) means that the rectangle covers the intersection of rows ρ_1 through ρ_2 with columns c_1 through c_2.

Table 8 provides all possible placements for each rectangle of products E and F. We ignore rectangles in which only two units of Product F are displayed because these lead to infeasible displays, i.e., displays with an empty slot. Each feasible placement corresponds to a node in the network shown in Figure 11. There are four possible independent sets, $\{E_1, F_3\}$, $\{E_2, F_1\}$, $\{E_3, F_6\}$, $\{E_4, F_4\}$, with total weighted revenues of 120, 120, 112 and 112, respectively. Thus, the optimal solution to the independent set problem will be the solution corresponding to either $\{E_1, F_3\}$ or $\{E_2, F_1\}$. Figure 12 shows the optimal display corresponding to the independent set $\{E_1, F_3\}$.

D.2 Algorithm Fill Slots

If Subproblem *SP2* is infeasible, then the following procedure (Algorithm Fill Slots) generates a default display that arranges each of the product's units so that they form contiguous rectangles and all slots

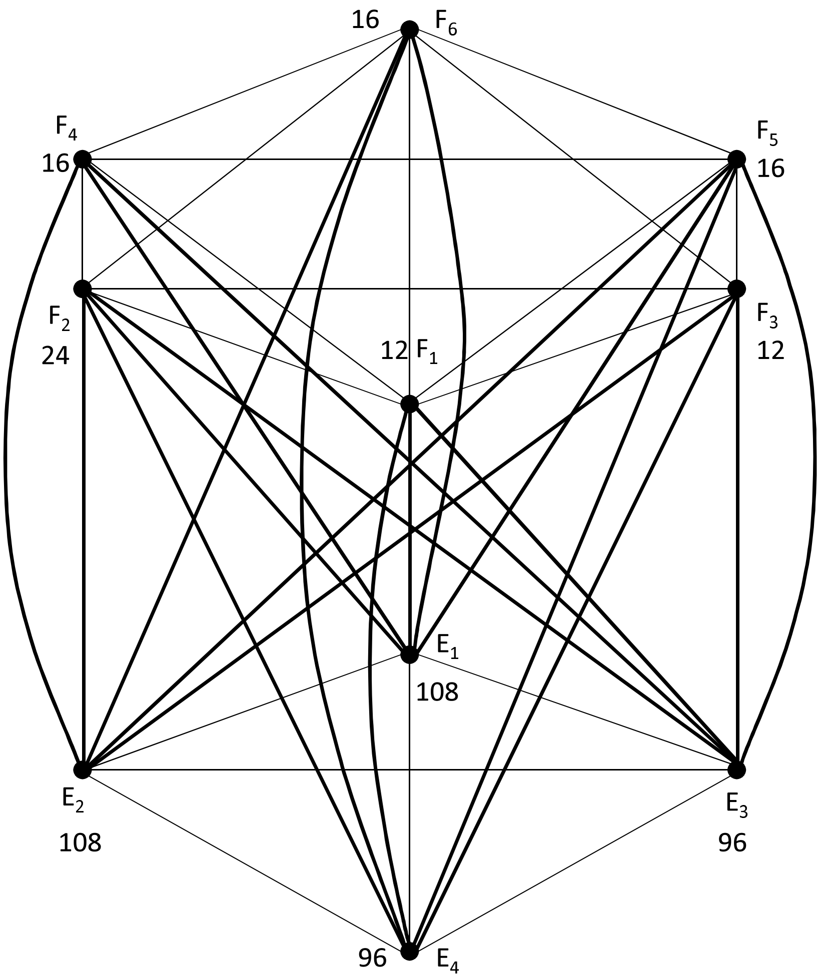

Figure 11: Graph for maximum-weight independent set problem of Example 2.

of the cabinet are filled. Suppose that *MIP-SP1* assigns m products to cabinet k. Note that $\sum_{d=1}^{m} z_d = R_k C_k$. Without loss of generality, assume that $z_1 \geq \cdots \geq z_m$ and $C_k \leq R_k$. We first provide an informal description of the procedure and then define it precisely.

Algorithm Fill Slots generates the promised display by allowing no product's display to extend beyond one row, except for the product with the highest revenue potential. The algorithm places products in the cabinet by non-decreasing order of units: $m, m-1, \ldots, 1$. For each row, the algorithm first determines which products are displayed in that row by summing the products' number of units until this sum is at least C_k. If necessary, the product in that row with the most units has its number of units reduced so that the row is filled precisely ($\sum_{i=d}^{last} z_i = C_k$). This process is repeated for the remaining products and successive rows until Product 2 is placed. Product 2 may have units added

F	F	F
E	E	E
E	E	E

Figure 12: The solution corresponding to set $\{E_1, F_3\}$, the total weighted revenue is 120.

or removed so that it completes its row exactly. Product 1 then fills the remaining integer number of rows within the cabinet. The order given for the values of *current_row* indicates that we fill the least valuable rows first. For ease of exposition, we have assumed R_k even and $z_1 > C_k$.

Algorithm Fill Slots

$last = m$

For $current\_row = (1, R_k, 2, R_k - 1, 3, R_k - 2, \ldots, R_k/2, R_k/2 + 1)$

  /* determine which products are displayed in this row */

  $d = last$

  While $\sum_{i=d}^{last} z_i < C_k$ Do

    If $d = 2$ Then Exit Loop

    $d = d - 1$

  Loop

  $z_d = C_k - \sum_{i=d+1}^{last} z_i$ /* ensure products $d, \ldots, last$ fill the row exactly*/

  Fill *current_row* with products $d, d + 1, \ldots, last$

  If $d > 2$ Then

    $last = d - 1$ /* set value of $last$ for next row */

  Else

    Exit For /* products $2, \ldots, m$ have been assigned */

  End If

Next *current_row*

Fill the remaining rows with Product 1

E Results for Generated Data

To supplement the testing of our solution procedures, we generated 1800 instances of the problem for data from various distributions. Values for revenue potential (r_d) were generated from log normal, negative binomial, normal, and uniform populations. Multiple sets of parameters were chosen for each distribution, and for each such set we generated 100 instances with 200 products and 100 instances with 500 products. The first distribution used was log normal, first with mean $\mu = 3$ and standard deviation $\sigma = 2$, and then with $\mu = 8, \sigma = 10$, as these were similar to two of the datasets observed at a Blockbuster store (details can be found in the next subsection), according to Oracle Crystal Ball, a forecasting and simulation application suite. For the negative binomial, the parameters were $r = 20$, $p = 0.8$, and $r = 10$, $p = 0.4$. (For the negative binomial distribution, r represents the number of failures until the series of trials is stopped, and p is the probability of success on each trial. Agrawal and Smith (1996) demonstrated that the negative binomial can effectively simulate retail demand data.) For the normal distribution, the parameters were $\mu = 12, \sigma = 4$; $\mu = 12, \sigma = 8$; and $\mu = 3, \sigma = 8$. The uniform distributions covered the discrete intervals $[2, 20]$ and $[2, 36]$.

E.1 Computational Results

Cabinets using our proposed scheme were compared to those using Blockbuster's current practice, which is described in Section 1. Results for instances with 200 products (using 26 cabinets) and for instances with 500 products (using 70 cabinets) are in Table 9.

Parameters	**ln nrm (3,2)**	**ln nrm (8,10)**	**neg bin (20,0.8)**	**neg bin (10,0.4)**	**normal (12,4)**	**normal (12,8)**	**normal (3,8)**	**unifm [2,20]**	**unifm [2,36]**	**average**
200 Products	10.29%	13.29%	9.51%	7.97%	12.08%	17.07%	20.90%	15.01%	10.64%	12.97%
500 Products	10.48%	17.79%	8.45%	8.57%	13.33%	19.01%	26.59%	16.05%	8.91%	14.35%

Table 9: Relative advantage of our solution procedure over current practice for generated data.

We find that our procedure is significantly better than the current practice, based on the metric derived from Blockbuster's criteria. The data generated using the normal distribution provided the most improvement (12-26%). The improvements in performance were observed for both small (200 products) and large (500 products) datasets, with the large datasets having greater effects on average.

As a further measure of the quality of our procedure, we compare its performance to upper bounds derived from (*IP-UB*), a binary integer program that solves a relaxed version of problem P (see Appendix B.2). This integer program assigns each disc to a specific location, but does not impose the contiguous rectangle constraint. Because it requires excessive computational time, we use its linear programming relaxation to find an upper bound on the optimum weighted revenue for displays with 200 products. The results are presented in Table 10; each entry represents the average percentage of the upper bound that is achieved by our solution procedure over 100 instances for the listed distribution. The results are quite good: the average across all instances is within 5% of optimum, even though the upper bounds are not necessarily tight.

ln nrm (3,2)	ln nrm (8,10)	neg bin (20,0.8)	neg bin (10,0.4)	normal (12,4)	normal (12,8)	normal (3,8)	unifm [2,20]	unifm [2,36]	average
92.11%	91.78%	97.45%	93.99%	98.52%	97.06%	95.28%	94.42%	96.50%	95.23%

Table 10: Average percentage of upper bound achieved by our procedure for generated data (200 products).

E.2 Sensitivity Analysis

We used this generated data to perform sensitivity analysis on the row effectiveness values. For each distribution of generated demand data, we solved the 100 instances with 200 products using four different vectors $\vec{a} = (a_1, a_2, a_3, a_4) = (a_8, a_7, a_6, a_5)$ of effectiveness coefficients: (5,6,8,10), (1,6,8,14), (7,7,7,8), (1,2,12,14). The results are in Table 11. Each row shows the weighted revenue for all distributions for a given vector of coefficients. Given the different parameters and characteristics of the distributions, we cannot expect any pair of values within any row to be the same. The same can be said for the effectiveness vectors and the values within any column. However, these values are tightly coupled in that for each pair of rows (and, consequently, for each pair of columns), the correlation coefficient of the respective weighted revenues is $r = 0.999$; therefore, there is no preference for using certain vectors of effectiveness values with specific distributions. Furthermore, this strongly suggests that the optimal displays for each distribution are the same for each vector of effectiveness values. Our observations of the solutions indeed support this assertion. It follows that the optimal display is largely unaffected by the choice of effectiveness values, provided $a_1 = a_8 \leq a_2 = a_7 \leq a_3 = a_6 \leq a_4 = a_5$.

$\vec{a}$	ln nrm (3,2)	ln nrm (8,10)	neg bin (20,0.8)	neg bin (10,0.4)	normal (12,4)	normal (12,8)	normal (3,8)	unifm [2,20]	unifm [2,36]
(5,6,8,10)	19107.1	198756.5	384946.9	49684.9	218049.5	289913.0	84382.1	228433.4	578531.9
(1,6,8,14)	22369.5	223942.2	410691.3	55351.3	230550.8	313933.9	96624.2	243823.7	613854.9
(7,7,7,8)	17245.5	184896.5	371090.6	46711.9	211362.1	277148.4	77398.9	220780.1	564541.7
(1,2,12,14)	23435.8	235969.7	428805.9	56570.5	236707.6	328688.1	101918.7	250102.6	634156.4

Table 11: Weighted revenue for different row-effectiveness vectors (200 products).

We test our method's sensitivity to changes in the revenue potentials by comparing the solution generated from using one set of potentials ($r_d, d = 1, \ldots, D$) to the solution generated using a set of potentials $\bar{r}_d$ that are formed by randomly-perturbing the first set of potentials: $\bar{r}_d = (1 + \alpha_d)r_d$, $d = 1, \ldots, D$, where α is uniformly distributed over $[-\Omega, \Omega]$, for $\Omega = 0.1, 0.15, 0.2$. The results suggest that our method is stable: over 100 instances for each value of Ω, the averages for the absolute differences between the total weighted revenues are 2.56%, 4.00%, and 5.21%, respectively.

We test the robustness of our method by using these same three sets of $\bar{r}_d$ values to examine the potential loss from incorrect estimates of the revenue potentials. In each instance, a display is generated using one set of potentials r_d. We then calculate the total weighted revenue of this display if potentials

$\bar{r}_d$ were the actual values, i.e., the test assumes that the $\bar{r}_d$ are the true potentials, but the display was generated using the r_d values. This total weighted revenue is then compared to the optimum total weighted revenue for a display generated using the $\bar{r}_d$ values. This comparison measures how much is lost if we were to use incorrect potentials r_d. The average losses from using the wrong potentials are negligible: 0.69%, 1.13%, and 1.87%, respectively. Furthermore, these losses are significantly less than the marginal improvement realized by using our scheme rather than Blockbuster's current methodology. In sum, our methodology is robust with respect to both effectiveness coefficients and revenue potentials.

References

Agrawal, N., S.A. Smith. 1996. Estimating Negative Binomial Demand for Retail Inventory Management with Unobservable Lost Sales. *Naval Research Logistics* **43** 839-861.

Bultez, A., P. Naert. 1988. SH.A.R.P.: Shelf Allocation For Retailers' Profit. *Marketing Science* **7**(3) 211-231.

Garey, M.R., D.S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, San Francisco.

Lim, A., B. Rodrigues, X. Zhang. 2004. Metaheuristics with Local Search Techniques for Retail Shelf-Space Optimization. *Management Science* **50**(1) 117-131.