

Product Portfolio Restructuring: Methodology and Application at Caterpillar

Masha Shunko, Foster School of Business

University of Washington, mshunko@uw.edu, +1 (206) 685-8599

Tallys Yunes, School of Business Administration

University of Miami, tallys@miami.edu, +1 (305) 284-5107

Giulio Fenu, Caterpillar Inc.

fenu_giulio@cat.com

Alan Scheller-Wolf, Tepper School of Business

Carnegie Mellon University, awolf@andrew.cmu.edu, +1 (412) 268-5066

Valerie Tardif, GT Nexus

vtardif@gmail.com

Sridhar Tayur, Tepper School of Business

Carnegie Mellon University, stayur@andrew.cmu.edu, +1 (412) 268-3584

Abstract

We develop a three-step methodology to restructure a product line by quantifying the restructuring's likely effects on revenues and costs: (i) Constructing migration lists to capture customer preferences and willingness to substitute; (ii) Explicitly capturing the (positive and negative) cost of complexity across different functional areas using statistical analysis of cost data; and (iii) Integrating these tools within a mathematical optimization program to produce a final product line, incorporating the possibility of differentiating products by lead-time (into different *lanes*). Our methodology is highly flexible—each step can be tailored to a company's particular setting, data availability and strategic needs, so long as it produces the necessary output for the next step. We report on the successful application of our methodology to the Backhoe Loader product line at Caterpillar: In collaboration with Caterpillar we were able to significantly simplify this line, reducing the number of configurations from 37,920 to 135, in three lanes, while increasing sales by almost 7%.

Key words: product portfolio optimization; cost of complexity; manufacturing; machinery.

Received: February 2015; *accepted:* August 2017 by Sean Willems after two revisions.

1 Introduction

One of the crucial decisions a company must make concerns its product portfolio. Some companies, especially in highly competitive industries, compete by offering *variety* – a very broad, often highly customizable, portfolio. Such a product line can typically garner high customer satisfaction and help drive market share, but can also complicate the company’s supply chain and service operations: Broad portfolios enable customers to disperse their demand, requiring retailers to hold large amounts of inventory to represent the many possible choices, as well as to satisfy those customers who are unwilling to wait for an out-of-inventory variant. In addition, forecasting for this potentially fragmented portfolio is typically difficult. And, if the company is in a manufacturing environment, developing, managing, assembling and servicing such a broad portfolio may incur other costs – for example due to increased documentation, more frequent change-overs, and reduced learning effects.

So, while the marketing and sales benefits of a large portfolio are obvious, there is also incentive for companies to strategically *reduce*, or *optimize* their product portfolio. But before embarking on such a product line rationalization, three critical questions must be answered:

1. “How will customers react to a product line reduction?” Answering this question requires developing an understanding of *how customers value different elements of the product line*.
2. “How much could be saved by focusing the product line?” Answering this question requires developing an understanding of the *general form of the cost of complexity* for the organization
3. “Given the answers to the first two questions, how should we configure our product line?”
Specifically, which products should we offer, and at what prices?

The primary contribution of this paper is to demonstrate the power of the three-step analytical framework we have developed for product line simplification:

Step 1: answers question 1 by building a detailed analytical model of customer preferences and substitution, capturing customer behavior in *migration lists* (see Section 3).

Step 2: answers question 2 by creating a detailed mathematical representation of the company’s cost of complexity (CoC), which can include both variety-based costs driven by *number* of options offered and attribute-based costs driven by *specific complex* options (see Section 4).

Step 3: answers the final question by combining the migration lists and the cost of complexity function into an optimization model, evaluating different product lines against different potential demand patterns, market scenarios and company objectives (see Section 5).

The generality and flexibility of our framework stem from the fact that the mathematical and statistical techniques used in steps 1, 2, and 3 can be tailored to the situation and company at hand, as long as they produce the output required by each subsequent step. Our framework can also be used as an effective what-if tool for managers, allowing them to successfully evaluate different solutions under varying problem conditions.

We demonstrate our framework through a product line reorganization project we initiated with Caterpillar (CAT) for pricing and marketing their BHL series of small backhoe loaders, one of the most popular products within their Building Construction Products (BCP) division. The outcome of the project was implemented as a new *Lane* strategy at BCP, offering machines within three different lanes: Lane 1, the Express Lane, featuring four built-to-stock configuration choices at an expected lead time of a few days; Lane 2, the Standard Lane, featuring 120 predefined configurations, built-to-order at an expected lead time of a few weeks; and Lane 3, the A-La-Carte Lane, built to order machines with an expected lead time of a few months. Since the completion of our project CAT has continued expanding and refining their BHL lane strategy—for example, they have now reduced the Lane 1 configurations to only *two*. In addition, CAT has applied variations of our cost of complexity analysis to other divisions within the firm, helping to guide portfolio rationalization within the company.

To the best of our knowledge, no previous work has ever combined an empirically-developed CoC function as detailed and comprehensive as ours, with customer preferences regarding product substitution, within an optimization algorithm that was implemented with real-life data, at an industrial scale. Moreover, our work ultimately produced recommendations that were actually implemented and verified to generate significant improvements.

The rest of the paper proceeds as follows. In Section 2, we place our work within the product line optimization and practical application literature. Sections 3 – 5 present our customer behavior, cost of complexity, and optimization models in a generic fashion that is neither company- nor product-specific. In Section 6 we describe CAT’s problem in more detail, and demonstrate how the steps in the three preceding sections were tailored to suit CAT’s specific requirements. We present the implementation details and results for CAT in Sections 7 and 8, discuss sensitivity analysis and general insights in Section 9, and conclude in Section 10.

2 Literature Review

Some marketing research describes how narrowing a product line may detract from brand image or market share, e.g. Chong et al. (1998), while other works posit that *reducing* the breadth of lines and focusing on customer “favorites” may actually *increase* sales, see for example Broniarczyk et al. (1998). Our model is consistent with both of these streams: If a customer finds a product that meets her needs (i.e. a “favorite”) she will make a purchase; if such a product and its acceptable alternatives are no longer part of the product line, she will not.

There is a long history of empirically studying the impact of product line complexity on costs. Foster and Gupta (1990) assess the impacts of volume-, efficiency-, and complexity-based cost drivers within an electronics manufacturing company. They find that manufacturing overhead is associated with volume, but not complexity or variety. Banker et al. (1995), using data from 32 plants, find an association of overhead costs with both volume and transactions, which they take as a measure of complexity. Anderson (1995) identifies seven different types of product mix heterogeneity in three textile factories, and finds that two are associated with higher overhead costs. Fisher and Ittner (1999) analyze data from a GM assembly plant, finding that option variety contributes to higher labor and overhead costs. We complement these works by explicitly formulating and calibrating a detailed model to estimate the total direct and indirect costs (and benefits) of complexity for the BHL line at CAT, based on expert surveys and empirical analysis.

Product line optimization has a rich literature: Kok et al. (2009) and Tang (2010) provide recent surveys. Several recent papers consider the *strategic* selection of a product line via equilibrium analysis: Alptekinoglu and Corbett (2008), Chen et al. (2008), Chen et al. (2010), and Tang and Yin (2010); they focus on deriving general insights via analysis of abstract models. Our paper uses math programming to optimize a detailed model of a company, their customers and products based on data and expert opinion. In addition, we implement our solution in practice.

Bitran and Ferrer (2007) determine the optimal price and composition of a single bundle of items and a single segment of customers in a competitive market. They provide extensions to multiple segments or multiple bundles based on mathematical programming, but this latter problem becomes very complex, and is left as future research. Wang et al. (2009) use branch-and-price to select a line to maximize the share of market, testing their algorithm on problems with a small number of items but many levels of product attributes on simulated and commercial data. Chen and Hausman (2000) demonstrate how choice-based conjoint analysis can be applied to the product portfolio problem; Schoen (2010) extends this work to allow more general costs and

heterogeneous customers. None of these algorithms have been shown to be suitable for problems anywhere near the size and complexity of CAT’s (thousands of customers and millions of potential configurations). This has led to the investigation of heuristic methods: For example, Fruchter et al. (2006) and Belloni et al. (2008). Neither of these are actual implementations.

Kok and Fisher (2007) develop and apply a methodology to estimate demand and substitution patterns for a Dutch supermarket chain, based on empirical demand data. They develop an iterative heuristic that determines the facings allocated to different categories, and the inventory of individual elements within the categories. Fisher and Vaidyanathan (2011) explore how to select retail store assortments; their work enhances a localized choice model with randomization, location at extant configurations, and preference sets for substitution (similar to our *migration lists*). All customers who prefer a particular product have the same preference set. In contrast to our approach that seeks to maximize profits using our empirical cost of complexity function, they maximize revenue with greedy heuristics and demonstrate their approach using two examples—snack cakes and tires— their recommendations for tires leads to a 5.8% revenue increase.

Ward et al. (2010) develops two analytical tools to apply to Hewlett-Packard’s product line problem. Like CAT, HP has product lines that could, in theory, span millions of different configurations. The first tool develops a comprehensive cost of complexity function, comprised of variable and fixed costs, to be used when evaluating the introduction of new products. This function has some similarities to ours, but focuses more on inventory costs, lacking anything related to our attribute based costing. Furthermore, cannibalization, which is how they refer to any substitution effects on inventory, are in their words “subjectively estimated” at a high level. Their second tool uses a heuristic to construct a line from a selection of extant products. This tool does not use their cost of complexity function, nor does it consider substitution—rather it constructs a Pareto frontier of those top k products that would cover the desired percentage of historical order demand (or order revenue). So while they seek the appropriate line to satisfy possibly multi-product orders assuming customers will not substitute, we find the correct line of products to satisfy orders for individual products in which customers may substitute. Rash and Kempf (2012) find the set of products Intel should produce to maximize profit over a time horizon while obeying budget and availability constraints. They perform hierarchical decomposition, utilizing genetic algorithms along with MIPs. Their demand is deterministic, so substitution is not included in the model.

The three-step framework we use was first introduced in Yunes et al. (2007), which describes a product line simplification effort implemented at John Deere & Co. Our current work extends

their work in several dimensions. Specifically, we: (i) Explicitly calculate and validate estimates of the parts utilities; they were exogenous in Yunes et al. (2007); (ii) Create a sophisticated, endogenous, cost of complexity function; the function used in Yunes et al. (2007) was exogenous; (iii) Owing to the form of our endogenous function, we use a different optimization procedure, the “differential approach;” (iv) To achieve CAT’s aggressive product line goals, we make decisions at the *option* level, rather than the machine level, as in Yunes et al. (2007); and (v) we incorporate pricing decisions and migration across models, absent in Yunes et al. (2007).

Compared to the literature, our work is unique in that cost of complexity, utility estimation and substitution behavior is modeled, estimated, *and* incorporated into a modular solution framework for the product portfolio problem, applicable across different industries and problem settings. In addition, we demonstrate how our solution can be used in practice; describing a dramatic redesign of a product portfolio at CAT.

3 Modeling Customer Behavior

The key to evaluating the potential pitfalls of reducing a product line is a good understanding of customers’ purchasing flexibility: While customers will require that the configuration they are buying satisfy some minimum requirements, not every feature needs to be in perfect alignment with their expectations. In addition, customers typically display some degree of price flexibility.

The centerpiece of our approach to capture customer flexibility is the *migration list*, an ordered list of configurations within the customer’s price, utility, and availability tolerance (see Yunes et al. 2007 for details). The first configuration on the list is the customer’s first choice; if available the customer will buy it. If that configuration is unavailable and there exists a second one on the list the customer will buy that, if available, and so on. If none of the configurations on a customer’s list are available, that customer buys nothing (i.e. goes to a competitor). As mentioned in Section 2, this is an enhanced localized choice model, in the spirit of Fisher and Vaidyanathan (2011).

One advantage of this methodology is that it is independent of the way migration lists are created; the only requirement is that there be one list, L_i , per customer i , consisting of a collection of configurations sorted in decreasing order of *preference*, where preference is defined by some *ranking function*. This ranking function could map configurations to utilities (as calculated by conjoint analysis (Hauser and Rao, 2004)), or to purchase probabilities (as in a multinomial logit model (Guadagni and Little, 1983)), or to any other quantitative measure of choice.

The configurations on customer i ’s migration list L_i could also be determined by sales history.

If customer i purchased machine M_i , L_i should contain configurations “similar enough” to M_i to satisfy i . There are several ways to define a *similarity function*. It could be as sophisticated as a formal metric in the space of configurations, or as simple as a conjunction of conditions. For example: their utilities and prices do not differ too much, *and* the number of features on which they differ is not too great, *and* they share compatible options for a few crucial features, etc. To illustrate the last condition, assume customer i needs a machine with large towing capacity (engine power is a crucial feature for i). All acceptable substitutes for M_i need to have an engine at least as powerful as M_i ’s engine. Once those machines “similar enough” to M_i are determined, they would be ranked and placed in L_i . In some settings, L_i may need to be truncated once its length reaches a certain threshold value to capture possible limits on customer willingness to substitute. Finally, if creating different ranking and/or similarity functions for each customer is too burdensome, customers can be clustered into market segments.

In summary, the following steps are repeated for each customer i in the optimization (assume, for the sake of illustration, we use the method based on sales history):

1. Let s be the customer segment to which i belongs (possibly unique for each customer);
2. Let g_{is} and h_{is} be, respectively, the *similarity* and *ranking* functions tailored for i and/or s ;
3. Apply g_{is} to M_i to obtain a list L_i of configurations that are acceptable substitutes for M_i ;
4. Sort the elements of L_i in non-increasing order according their h_{is} value;
5. Truncate L_i to a maximum acceptable length and save it for the optimization step.

4 Capturing Cost of Complexity

Our next task is to estimate how a line reduction might affect costs. Product variety affects many functional areas in heterogeneous ways, and in some areas, the impact on costs is not straightforward: Sales costs may increase as variety increases because a large line may overwhelm customers and sales personnel; on the other hand, sales costs may decrease in variety if it is easier to satisfy a demanding customer. As a result, complexity has to be understood in each functional area and individually modeled in different departments. We refer to all costs impacted by the variety of product offerings, i.e. number of features and options, as the *cost of complexity*.

We describe important elements of cost of complexity, propose a cost of complexity function that captures these elements, and derive a differential cost of complexity function in Sections 4.1,

4.2, and 4.3 respectively. The result of this process is used by our optimization model in Section 5.

4.1 Important Elements of Cost of Complexity

4.1.1 Option Effects

We distinguish between two main effects: Certain processes are impacted by the *number* of options offered for a feature, while other processes are impacted by the presence of *specific* options or combinations of options (within one feature or across features). For example, material planners need to calculate stocking requirements for each SKU offered. If one SKU is eliminated, the cost of complexity will go down proportionally, regardless of which SKU is eliminated. We refer to this effect as *Variety Based Complexity*, or *VBC*.

In contrast, other features may include simple and complex options; engineering cost for releasing a complex option may be much higher than that for releasing a simple option. Hence, the reduction in the cost of complexity will depend on the particular option eliminated. We refer to this effect as *Attribute-level Based Complexity* or *ABC*. ABC is not limited to single options; there may be cases in which a combination of options drives the cost of complexity.

4.1.2 Temporal Effects

Building the cost of complexity function also requires understanding the lagged impact of complexity on costs. For example, assembly cost today is impacted by the product complexity being built today, while warranty costs are affected by the complexity that was offered a certain time ago (positive time lag), and engineering and marketing costs may be impacted by the complexity that will be offered in the future (negative time lag). Some of these time lags may already be incorporated into the cost data, e.g. accounting may allocate costs for material write-offs to the month when an option was discontinued. In contrast, expenses paid to sub-contractors involved in development of a new set of options are likely to be recorded in the month when the work is being done, not in the months when the options will be added to the price list. Hence, it is important to talk to accounting about potential lags in data.

4.1.3 Volume Effects

Finally, the cost of complexity is impacted by different volume metrics. Not surprisingly, most processes are affected by sales volume: Costs increase as more items are produced and sold. However, costs are also driven by other volumes. For example, product support is impacted by the number of *unique* configurations *built*, because quality may decrease when employees have to work on many different configurations. Other processes, such as engineering, are impacted by

Table 1: Table of notation

$d \in \mathbf{D}$	Superscript used to represent attributes pertaining to department d , where \mathbf{D} is the set of all departments considered in the study;
$f \in \mathbf{F}$	Subscript used to represent attributes pertaining to feature f , where \mathbf{F} is the set of all features in the product line;
$\mathbf{F}^d \subset \mathbf{F}$	The set of all features identified as relevant for department d ;
$o \in \mathbf{O}$	Superscript used to represent attributes pertaining to option o , where \mathbf{O} is the set of all options in the product line;
$\mathbf{O}_f \subset \mathbf{O}$	The set of all options in feature f ;
N_f	Number of options in feature f : $ \mathbf{O}_f $;
\mathbf{N}^d	Set of cardinalities of all features relevant for department d : $\mathbf{N}^d = \{N_f : f \in \mathbf{F}^d\}$;
$VBC^d(\cdot)$	The cost of complexity caused by variety at department d ;
$ABC_o(\cdot)$	The cost of complexity caused by offering a specific option o ;
$DVBC^d$	The differential cost of complexity caused by variety at department d ;
$DABC_o$	The differential cost of complexity caused by offering a specific option o ;
$DCoC$	The total differential cost of complexity;
V	Total sales volume;
O	Number of configurations sold that contain option o ;
\mathbf{V}	Set of configurations sold for all options: $\mathbf{V} = \{V^o : o \in \mathbf{O}\}$;
U	Total number of <i>unique</i> configurations sold;
l^d	Time lag parameter at department d ;
ϕ^d	The size of the cost pool at department d relative to other departments;
α^d	The effect of the sales volume on the cost of complexity at department d ;
ξ^d	The effect of the number of unique configurations sold on the cost of complexity at department d ;
γ_f^d	The effect of the cardinality of feature f on the cost of complexity at department d ;
$\delta_o^d, \delta_o = \sum_{d \in \mathbf{D}} \delta_o^d$	One time cost incurred if option o is offered at department d and total across departments respectively;
a_o	Binary variable that indicates whether option o is offered or not;
$\omega_o^d, \omega_o = \sum_{d \in \mathbf{D}} \omega_o^d$	Cost incurred <i>each time</i> option o is produced at department d and total across departments respectively.

the complexity *offered*, as engineers have to prepare releases for all options and each associated feasible configuration, while sales volume is unlikely to have impact on the engineering cost.

4.2 Cost of Complexity Function

We estimate two separate components of the cost of complexity function: $VBC^d(\cdot)$ is the cost of complexity caused by variety and is specific to each functional area or department d , and $ABC_o(\cdot)$ is the cost of complexity caused by offering a specific option o . We summarize this and additional notation used in this section in Table 1 – we use small letters for superscripts and subscripts, bold letters for sets, and capital letters for numbers coming from collected data.

4.2.1 Estimation of VBC

We use the Cobb-Douglas log-linear function to estimate the variety-based effect on the cost of complexity for each department d . The Cobb-Douglas function is frequently used for estimating non-linear relationships (see Greene 2000); it can capture different returns to scale and has several attractive analytical properties. Two properties are of particular convenience for us: First, the log transformation of the Cobb-Douglas function is linear and hence can be estimated using linear regression. Second, a partial derivative of the Cobb-Douglas function has a simple form that is useful in the derivation of the *differential cost of complexity* function (Section 4.3).

Using lower-case Greek letters to represent estimated parameters, the Cobb-Douglas cost of complexity function at time t is given by:

$$VBC_t^d = \xi^d V_{t+l^d}^{\alpha^d} U_{t+l^d}^{\phi^d} \prod_{f \in \mathbf{F}^d} (N_{f,t+l^d})^{\gamma_f^d}. \quad (1)$$

The $\prod_{f \in \mathbf{F}^d} (N_{f,t+l^d})^{\gamma_f^d}$ term captures the complexity *offered*, by accounting for the number of options for each feature on the price list, and the $U_{t+l^d}^{\phi^d}$ term accounts for the complexity *built*.

We log-transform this function to use linear regression analysis to estimate needed parameters:

$$\text{Log}[VBC_t^d] = \text{Log}[\xi^d] + \alpha^d \text{Log}[V_{t+l^d}] + \phi^d \text{Log}[U_{t+l^d}] + \sum_{f \in \mathbf{F}^d} \gamma_f^d \text{Log}[N_{f,t+l^d}] + \epsilon_t. \quad (2)$$

Within our optimization model, we evaluate the effect of a one-time change to the product portfolio on the long-term cost. Hence we do not need the time dimension for further analysis and will suppress time subscript t , using a functional representation $VBC^d(\mathbf{N}^d, V, U)$ instead¹.

4.2.2 Estimation of ABC

Introducing a complex option may have different effects: 1) There may be a fixed cost associated with offering the option (e.g. part design cost); and 2) There may be variable costs incurred each time the option is produced (e.g. additional testing each time the part is installed). We model the total cost of offering and producing option o , ABC_o , as the sum of the one-time cost across all departments if option o is offered (complexity *offered*) and the incremental cost across all

¹Alternatively, one can refine this approach by keeping the time lags in the model and computing the net present value of future cash flows.

departments incurred each time a configuration with option o is built (complexity *built*):

$$ABC_o(a_o, O) = a_o \sum_{d \in \mathbf{D}} \delta_o^d + O \sum_{d \in \mathbf{D}} \omega_o^d.$$

4.3 Differential Cost of Complexity Function

To facilitate optimization, instead of computing the total cost of each offering, we build a cost of complexity function that starts with the current line and computes the *estimated change in cost* as the numbers of features and options change. For the variety-based component, we take the partial derivatives of $VBC^d(\mathbf{N}^d, V, U)$ with respect to all variables, and then combine them into an aggregate differential cost of complexity function.

For example, the change in VBC with the number of options for feature f is:

$$\frac{dVBC^d(\mathbf{N}^d, V, U)}{dN_f} = \frac{\gamma_f^d}{N_f} \xi^d V^{\alpha^d} U^{\phi^d} \prod_{i \in \mathbf{F}^d} N_i^{\gamma_i^d} = \frac{\gamma_f^d}{N_f} VBC^d(\mathbf{N}^d, V, U),$$

substituting the definition of $VBC^d(\mathbf{N}^d, V, U)$ given by (1).

This differential cost of complexity function includes the predicted value of $VBC^d(\mathbf{N}^d, V, U)$. This is problematic for two reasons: (i) The predicted $VBC^d(\mathbf{N}^d, V, U)$ contains an error term and hence may not give an accurate size of the cost pool; and (ii) Values of V , N_f , and U change during the specified time period. Therefore we approximate $VBC^d(\mathbf{N}^d, V, U)$ with the historical average cost at department d (\overline{D}^d) attributable to complexity, taken over an appropriate period:

$$\frac{dVBC^d(\mathbf{N}^d, V, U)}{dN_f} \approx \frac{\gamma_f^d}{N_f} \overline{D}^d. \quad (3)$$

Using Equation (3), the total change in cost of complexity due to a change in the number of options offered for feature f is then estimated as:

$$\overline{D}^d \frac{\gamma_f^d}{N_f} \Delta N_f,$$

where ΔN_f represents the change in the number of options in feature f after optimization; similarly we will precede V , U , O , \mathbf{N} , \mathbf{N}^d , and \mathbf{V} with Δ to represent change. These differential quantities depend on the market response to the options offered in the product portfolio, and will be computed within the optimization model by leveraging our customer migration model. We will formally define these functions in Section 5 when we introduce the optimization model.

In an analogous fashion we differentiate the cost of complexity function with respect to volume and number of unique configurations. Aggregating these differences yields the total variety-based *differential cost of complexity* $DVBC^d$:

$$DVBC^d = \sum_{f \in \mathbf{F}^d} \overline{D}^d \frac{\gamma_f^d}{N_f} \Delta N_f + \overline{D}^d \frac{\alpha^d}{V} \Delta V + \overline{D}^d \frac{\phi^d}{U} \Delta U. \quad (4)$$

To capture the differential ABC effect, we must account for the change in the number of options offered and sold. If we eliminate an option o from the price list the cost of complexity will decrease by $\delta_o = \sum_{d \in \mathbf{D}} \delta_o^d$ and if sales with option o deviate from O , the cost will likewise change:

$$DABC = \sum_{o \in \mathbf{O}} \delta_o (a_o - 1) + \sum_{o \in \mathbf{O}} \omega_o \Delta O, \text{ where } \mathbf{a} = \{a_o : o \in \mathbf{O}\}. \quad (5)$$

Finally, the total differential cost of complexity is:

$$DCoC = \sum_{d \in \mathbf{D}} DVBC^d + DABC. \quad (6)$$

Equation (6), with estimated parameters, becomes a part of the objective function in the optimization model of Section 5. Because the cost of complexity function is non-linear, this approach is accurate only for small changes. In practice, the accuracy of this approximation may be tested by calculating the actual change in the total cost of complexity for the original product line $VBC^d(\mathbf{N}^d, V, U)$ and for the optimized product line $VBC^d(\mathbf{N}^d + \Delta \mathbf{N}^d, V + \Delta V, U + \Delta U)$, and then comparing this to the result of the approximation $DVBC^d$.

5 The Optimization Model

With the migration lists from Section 3 and Equation (6) from Section 4, we are now ready to describe our optimization model. We use a mixed-integer linear program to select the set of options and configurations offered, and their prices, to maximize total profit from sales minus the change in cost of complexity (the objective function value increases when this change is negative).

In addition to the data defined in Section 4, our optimization model uses the following data:

- \mathbf{I} — The set of all customers;
- \mathbf{L}_i — The ordered set of configurations in the migration list of customer $i \in \mathbf{I}$. Each member of L_i is represented by a pair (j, k) , where k identifies the lane choice and j identifies the

specific options chosen for all remaining features. Therefore, the same j can appear in several pairs with different values of k , but only one (j, k) pair will appear in the final portfolio.

- \mathbf{J} — The set of all configurations appearing on migration lists ($\mathbf{J} = \bigcup_{i \in \mathbf{I}} \mathbf{L}_i$);
- \mathbf{O}_{jk} — The set of all options in configuration $(j, k) \in \mathbf{J}$;
- R_i — Reservation price of customer $i \in \mathbf{I}$; this could be configuration-dependent if desired (i.e. R_{ijk} instead of R_i);
- M — Maximum reservation price over all customers ($M = \max_{i \in \mathbf{I}} R_i$);
- C_{jk}, B_{jk}, P_{jk} — Cost, base price, and current sale price of configuration $(j, k) \in \mathbf{J}$, respectively. Base price is the starting price of an incomplete configuration before any of its options are included. The value of P_{jk} equals of B_{jk} plus the prices of all options contained in configuration (j, k) . We use this additive price structure to illustrate option pricing optimization, but it is not a requirement of our model, i.e., other price structures are also possible.

The decision variables (a_o was defined in Table 1, but we repeat it here for completeness):

- $a_o = 1$ if option $o \in \mathbf{O}$ is available, 0 otherwise.
- $q_{jk} = 1$ if configuration $(j, k) \in \mathbf{J}$ is bought by at least one customer, 0 otherwise;
- $x_{ijk} = 1$ if customer i buys configuration (j, k) , 0 otherwise ($i \in \mathbf{I}, (j, k) \in \mathbf{L}_i$);
- p_o — Price of option $o \in \mathbf{O}$ ($p_o \geq \delta_o$, where δ_o is defined in Table 1). These variables enable changing the prices of individual options; one could eliminate these variables entirely, or replace them with variables p_{jk} to price configurations. Using p_{jk} instead of p_o would result in simple changes to some of the constraints we present below;
- r_{ijk} — Profit if customer i purchases configuration (j, k) ($i \in \mathbf{I}, (j, k) \in \mathbf{L}_i$).

We now provide precise definitions to the following terms that appear in (4) and (5):

$$\begin{aligned} \Delta N_f &= \sum_{o \in \mathbf{O}_f} a_o - N_f, & \Delta U &= \sum_{(j,k) \in \mathbf{J}} q_{jk} - U, \\ \Delta V &= \sum_{i \in \mathbf{I}} \sum_{(j,k) \in \mathbf{L}_i} x_{ijk} - V, & \Delta O &= \sum_{i \in \mathbf{I}} \sum_{(j,k) \in \mathbf{L}_i \mid o \in \mathbf{O}_{jk}} x_{ijk} - O. \end{aligned}$$

Our optimization model is then:

$$\max \sum_{i \in \mathbf{I}} \sum_{(j,k) \in \mathbf{L}_i} r_{ijk} - DC_{oC} \quad (7)$$

$$q_{jk} \leq a_o \quad \forall (j,k) \in \mathbf{J}, o \in \mathbf{O}_{jk}, \quad (8)$$

$$x_{ijk} \leq q_{jk} \quad \forall i \in \mathbf{I}, (j,k) \in \mathbf{L}_i, \quad (9)$$

$$\sum_{k \mid (j,k) \in \mathbf{J}} q_{jk} \leq 1 \quad \forall j \mid (j,k) \in \mathbf{J} \text{ for some } k, \quad (10)$$

$$\sum_{(j',k') \text{ after } (j,k) \text{ in } \mathbf{L}_i} x_{ij'k'} + q_{jk} \leq 1 \quad \forall i \in \mathbf{I}, (j,k) \in \mathbf{L}_i, \quad (11)$$

$$r_{ijk} \leq B_{jk} + \sum_{o \in \mathbf{O}_{jk}} p_o - C_{jk} \quad \forall i \in \mathbf{I}, (j,k) \in \mathbf{L}_i, \quad (12)$$

$$B_{jk} + \sum_{o \in \mathbf{O}_{jk}} p_o \leq P_{jk}(1 + \text{MAX_INC}) \quad \forall (j,k) \in \mathbf{J}, \quad (13)$$

$$r_{ijk} \leq (\min\{R_i, P_{jk}(1 + \text{MAX_INC})\} - C_{jk})x_{ijk} \quad \forall i \in \mathbf{I}, (j,k) \in \mathbf{L}_i, \quad (14)$$

$$B_{jk} + \sum_{o \in \mathbf{O}_{jk}} p_o \leq R_i + (P_{jk}(1 + \text{MAX_INC}) - R_i)(1 - x_{ijk}) \quad \forall i \in \mathbf{I}, (j,k) \in \mathbf{L}_i. \quad (15)$$

The objective function (7) maximizes the total profit from sales minus the differential cost of complexity given by (6). The purpose of each constraint is as follows (where MAX_INC is the maximum allowed percentage price increase for any configuration): (8): If option o is not available ($a_o = 0$), no configuration that contains o can be bought ($q_{jk} = 0$); (9): If customer i buys configuration (j,k) ($x_{ijk} = 1$), then (j,k) must have been bought ($q_{jk} = 1$); (10): Configurations can be assigned to at most one lane; (11): If (j,k) is bought by someone ($q_{jk} = 1$), it must be available, therefore customer i cannot buy less desirable configurations (j',k') that appear after (j,k) in \mathbf{L}_i (all $x_{ij'k'} = 0$); (12): Profit cannot exceed price ($B_{jk} + \sum_{o \in \mathbf{O}_{jk}} p_o$) minus cost; (13): Configuration prices cannot increase by more than MAX_INC; (14): No purchase ($x_{ijk} = 0$) means no profit ($r_{ijk} = 0$) and, if a purchase happens ($x_{ijk} = 1$), the term in parenthesis is a ceiling on the value of r_{ijk} ; (15): If customer i buys configuration (j,k) ($x_{ijk} = 1$), (j,k) 's price must not exceed R_i .

We can turn off the price optimization aspect of this model by removing variables p_o and constraints (13)–(15), and changing the right-hand side of (12) to $(P_{jk} - C_{jk})x_{ijk}$. Lane assignment decisions can be removed by deleting constraint (10) and all occurrences of index k .

Table 2: Number of options in each feature of CAT’s BHL models.

Features	BHL Models			
	416E	420E	430E	450E
Sticks	2	2	2	2
Backhoe Hydraulics	3	3	3	6
Backhoe Controls	2	-	-	-
Loader Buckets	5	13	13	4
Loader Hydraulics	2	2	2	2
Cab/Canopy	5	5	4	2
Powertrain	4	3	2	-
Engine Cooling	2	2	2	-
Counterweights	4	4	4	-
Backhoe Aux Lines	3	3	3	3
Engine Coolant Heater	1	1	1	1
Product Link	1	1	1	1
Ride Control	1	1	1	-
Front Loader Mechanics	-	2	2	-

6 CAT-Specific Modeling Details

In this section we first provide some background on CAT’s BHL product families, in 6.1. We then detail the assumptions and decisions made to create the CAT-specific instantiation of the generic framework we describe in sections 3, 4, and 5, (in Sections 6.2, 6.3 and 6.4, respectively). CAT experts participated in the entire process, making sure they understood and, when necessary, validated, the inputs and outputs of each intermediate step.

6.1 BHL Product Families

Our product line simplification effort at CAT involved four models in the backhoe loader (BHL) family: 416E, 420E, 430E, and 450E. The 416E is their basic model, while the 420E, 430E and 450E provide progressively superior horsepower and capabilities. We refer to a complete machine as a *configuration*. Each configuration is composed of *features*; for each feature, a configuration specifies one of the *options* within that feature. For example, the feature *stick* has the options *standard* and *electronic*. In the marketing literature, what we call a feature is also known as an *attribute*, and what we call an option is also known as an *attribute level*. Table 2 summarizes the features and number of corresponding options present in each BHL model in our project. A dash “-” indicates that a feature is not present in a model or was not included in our analysis.

To create a complete configuration, a customer selects one option for each of its features, ensuring that these options are compatible. The number of such configurations is immensely

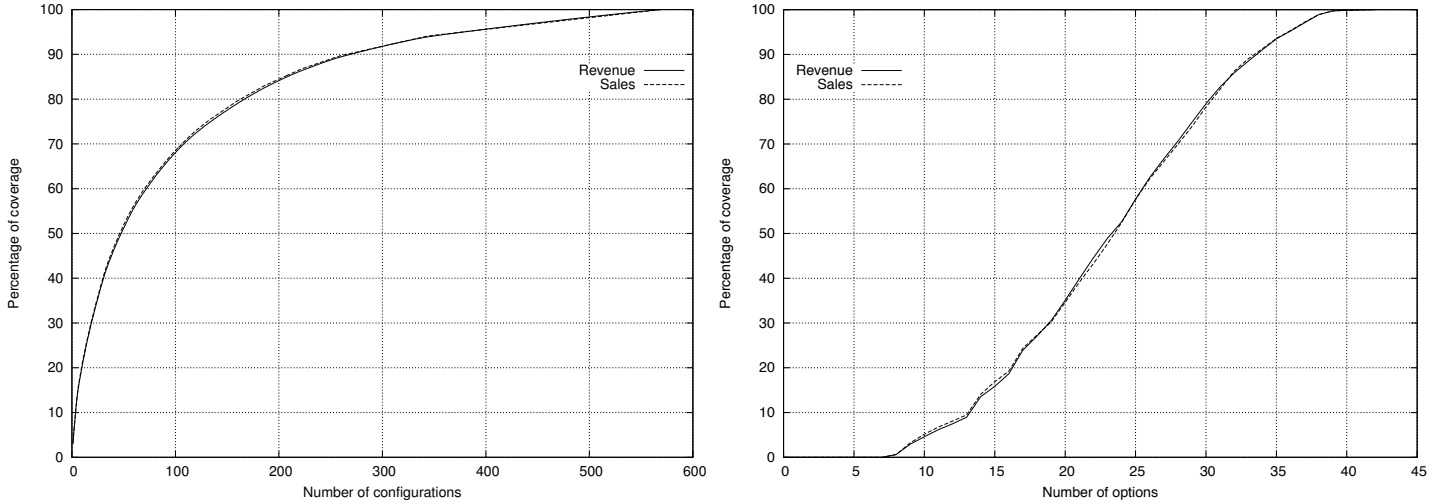


Figure 1: Minimum number of distinct configurations (left) and options (right) required to capture given percentages of revenue and sales volume for BHL model 420E.

large: For model 416E in its most basic version, there are 37,920 feasible configurations. Including choices for attachments yields 2,275,200 distinct feasible configurations. The vast majority of these configurations have never been, and most likely will never be, built. The mere fact that they *could* be purchased, however, creates overhead costs for CAT. Moreover, every unique option offered incurs a cost for CAT, due to the engineering and support costs it requires. We discuss this in detail in Section 4.

So how many configurations are actually built? Figure 1 depicts the minimum number of different configurations (left panel) and options (right panel) required to capture given percentages of revenue and sales, respectively, for eight month's worth of sales data for model 420E. Of the 569 built configurations, 400 were needed to capture about 95% of revenues and sales volume. Similarly, 36 out of the 42 available options were needed to capture at least 95% of revenues and sales volume. Therefore, to achieve the sought reductions in product offerings, it was imperative to steer purchases toward a considerably smaller subset of products and options.

As we will see in Section 8, CAT ultimately converged on a lane system in which configuration lead times depend on the lane to which they belong. Although not part of our original algorithm, our modeling framework can incorporate such a structure assuming the lanes and lead times are given. Specifically, lanes, with corresponding lead times, can be treated as options of an additional configuration feature, which we will call *Availability*. In addition, just as customers can be modeled as having tolerances for price and utility, there can be a maximum availability threshold per customer as well.

6.2 Modeling the Behavior of CAT's Customers

The pseudo-code below shows how migration lists were constructed for CAT; we use customer purchases in 2006 as the basis for generating our migration lists.

For each customer $i \in \mathbf{I}$ who bought a configuration over the past H months repeat:

1. Let M_i be the configuration (i.e. machine) bought by i .
2. Apply segmentation rules to M_i (Section 6.2.1) to place i in a customer segment S_i .
3. Based on the price and utility of M_i , and on characteristics of segment S_i , construct a *randomized* list of configurations, L_i , as acceptable alternatives to M_i (Section 6.2.2).
4. Sort L_i in non-increasing order of configuration utility, pruning it if it exceeds the maximum allowed length (Section 6.2.3).

6.2.1 Customer Segmentation

Customer segmentation is important in CAT's business because it affects customer flexibility. For example, customers who live in extreme weather conditions are unlikely to buy a configuration that does not include a cab with climate control, and customers who need to carry very heavy loads are not willing to sacrifice horsepower. We used focus groups composed of CAT experts and actual customers to identify the main customer segments and their characteristics: *performance extreme* (PE), *performance extreme versatility* (PEV), *performance mild* (PM), *performance mild versatility* (PMV), *commodity extreme* (CE), and *commodity mild* (CM). The performance category represents customers who are less price sensitive and need powerful machines. The extreme and mild categories refer to weather conditions, and the versatility category represents customers who need their machines to perform a variety of tasks. Based on historical sales data, the fraction of customers in each of the above six segments are approximately 20, 20, 25, 10, 5, and 20 percent.

A set of *segmentation rules* was created to classify each purchase: Given a configuration, its customer segment is determined by the presence and/or absence of certain options, represented as part numbers. For example, there are eight ways for a 416E loader to be placed in segment PE. One is: Two out of the options 2146913, 2099929, and 2139293 must be present (89HP powertrain and e-stick), *and* one out of the options 2044161, 2044162, and 2284602 must be present (cabs), *and* the option 2120206 cannot be present (6-function hydraulics), *and* neither option 2497912, nor option 2624213 can be present (one-way and combined auxiliary lines).

In addition to the segment-specific option utilities discussed in Section 6.2.2, segment-specific reservation prices and reservation utilities also affect migration list generation (Section 6.2.3).

6.2.2 Estimating Utilities

For each of the customer segments identified in Section 6.2.1, we calculate option utilities as follows. First, to estimate the importance of a model’s features, we asked a group of CAT employees with sales and manufacturing expertise to use the Analytic Hierarchy Process (AHP) (Saaty 1980). AHP asks experts to estimate the *relative importance* between every pair of features on a scale from 1 (equally important) to 9 (much more important). The pairwise scores are then transformed into absolute scores of relative importance for each individual feature. The same group of experts is then asked to rank the options within each feature on a scale from 0 to 100. These option scores are scaled so that the option receiving a score of 100 is assigned a value equal to its feature’s relative importance. These scaled scores represent the final option utilities. The utility of a complete configuration is estimated as the sum of the utilities of its options.

To validate the utility values calculated for each of the options—for every BHL model in all customer segments—we conducted a survey asking actual customers to choose among alternate configurations. Using t-tests, CAT determined that differences between the utilities derived from the survey results and those estimated by experts were not statistically significant.

6.2.3 Building Migration Lists

Although customers of a given segment tend to behave similarly, they are certainly not identical. To account for variations within each segment, we modify the migration list procedure in several ways. First, for each segment we randomly perturb the relative importance (and, consequently, the option utilities) of randomly selected features. The number of features to perturb is an input parameter (for CAT, this was around three). Given a perturbation factor θ (approximately ten), the change to a feature’s relative importance is randomly drawn from a uniform distribution over the interval $[-\theta\%, +\theta\%]$. CAT also did not want customers to have lists containing configurations too dissimilar from the one purchased. Therefore, a number called the *disparity factor* (around five) limits how many options an alternative configuration can have that differ from M_i . Finally, the model generates the customer’s reservation price and reservation utility; again, these values are randomly picked from a predetermined interval around the price and utility of M_i .²

²Note that randomness in our choice model is restricted to the generation of option utilities and reservation values, which influence the construction of customer migration lists. Once created, these (fixed) lists serve as input to a *deterministic* optimization algorithm. Thus we refer to our model as being “randomized,” as opposed to a *random choice model*, which typically has a different meaning.

We collect the above procedures into a Constraint Programming (CP) model (Marriott and Stuckey 1998) that finds feasible configurations for L_i . This CP model needs to know what constitutes a feasible configuration, i.e. which options are compatible. We use *configuration rules* to describe these interdependences. For example, for model 420E, one rule is: If a configuration has option 9R58666 and either option 2139272 or 2139273, then it cannot have option 9R5321. After all feasible configurations are found, those that exceed the generated reservation price or fall short of the reservation utility are pruned from the customer’s migration list.

Next, configurations are sorted in non-increasing order of total utility and L_i is truncated, if desired, while respecting two conditions. First, if L_i is truncated, M_i must always be retained. Second, we assume customers place M_i first, regardless of M_i ’s utility, with a certain probability (the β factor; for CAT it was between 0.3 and 0.7). This is an attempt to capture the fact that some customers are attracted to their M_i for reasons we cannot capture with utilities.

Migration across different models is also possible. In this case we apply a set of migration rules that map a purchased configuration M_1 of model m_1 (e.g. 416E) to its most likely counterpart M_2 , of a different model m_2 (e.g. 420E). Once M_2 is known, we generate alternatives as if it were the customer’s original purchase, and include them (together with alternatives to M_1) onto L_i . Because m_2 configurations may have higher utilities, when L_i is sorted it may contain almost no highly ranked m_1 configurations. Thus, to capture the fact that customer i originally preferred an m_1 configuration, we inflate the utilities of all m_1 configurations on L_i by a *preference factor* (between 10 and 20%). As a result, L_i ends up with configurations of both models, but it does not allow utilities to overemphasize the attractiveness of m_2 configurations. According to CAT, the plausible model migrations are from 416E to 420E and from 430E to 420E.

As was done for option utilities, we also conducted an extensive validation study with CAT experts to evaluate the quality of our migration lists. Throughout this process the experts provided valuable feedback that helped us fine tune our input parameters. After a few iterations, CAT experts agreed that our migration lists could be safely used by our optimization algorithm.

6.3 Estimating the Cost of Complexity at CAT

6.3.1 Understanding the Impact of Complexity at CAT

In conjunction with CAT experts, we identified nine functional areas impacted the most by complexity. Within each area we (i) identified up to three major processes most impacted by product complexity; (ii) found cost-measures that capture the impact of complexity for each

Table 3: Main business processes impacted by complexity and the corresponding cost measures. A dagger (†) indicates alternative cost measures used due to lack of data availability.

<i>Department</i>	<i>Processes impacted</i>	<i>Measure of complexity</i>
Purchasing	Capital tooling Supplier operations	Capital tooling cost Supplier delivery performance [†]
Customer acquisition	Ordering Forecasting Quoting and training	Cost of customer acquisition [†]
Marketing	Price list creation Training Publications	Budget expenditure [†]
Engineering	Drawing changes Original design & development	Cost of engineering changes Cost of product and component Cost of new releases
Order fulfillment	Attachment forecasting Sequencer work Grief resolution	Headcount cost [†]
Product support	Dealer solution network - calls Publications - manuals Warranty costs	Cost of service calls Cost of publications Cost of repairs (first 10 hours) Cost of repairs (during 11-100 hours) Cost of repairs (above 101 hours)
Material planning	Inventory management Schedule volatility Expedition	Inventory handling cost (prime product) Inventory handling cost (components) Headcount cost Freight cost
Operations	Initial process setup Assembly process	Person-hour cost, production planning Person-hour cost, assembly
Quality	Initial setup Hot test Cab test	Cost of initial setup Hot-test cost (person-hours) Cab-test cost (person-hours)

major process; and (iii) identified particular product features and/or options that have the largest impact on the cost of complexity. Table 3 lists the functional areas, processes impacted, and measures used (when alternate cost measures are used due to data unavailability they are denoted by a dagger †). Below we elaborate on several cost measures in Table 3.

Cost of supplier delivery performance refers to a program targeted towards improving availability, in which CAT contacts suppliers with low delivery performance to improve their processes. We use the cost allocated to this program as a proxy for the cost of supplier operations. In the customer acquisition department, CAT calculates sales variance cost by tracking all the discounts that go into making a sale: invoice, extended service, cost of free attachments, etc. We use this measure to approximate the *cost of customer acquisition*. We rely on CAT’s accounting system

Table 4: VBC/*ABC* classification of features.

<i>Department/Feature</i>	Backhoe Hydraulics	Loader Buckets	Auxiliary Lines	Quick Coupler	Hoe Buckets	Control Groups	Cab/Canopy	Sticks
Purchasing		VBC		VBC	VBC		VBC	VBC
Customer acquisition	VBC		VBC				VBC	
Marketing	VBC		VBC				VBC	
Engineering	<i>ABC</i>	VBC	VBC		VBC		<i>ABC</i>	
Order fulfillment	VBC	VBC	VBC	VBC	VBC	VBC	VBC	VBC
Product support	VBC		VBC				VBC	
Material planning	VBC	VBC			VBC		VBC	
Operations			<i>ABC</i>				<i>ABC</i>	<i>ABC</i>
Quality							<i>ABC</i>	

for cost estimates of engineering changes (primarily consisting of payroll to engineers working on changes) and engineering of new releases (primarily consisting of the payroll of developers and engineers who work on new parts, and costs of testing and design equipment).

6.3.2 Option Effects for CAT

The next step was to understand which features have VBC and/or ABC effects on identified processes. The results of VBC/ABC classification are summarized in Table 4.

Table 5: Summary of the time lags (in months).

<i>Department</i>	<i>Time Lag</i>	<i>Department</i>	<i>Time Lag</i>
Purchasing	0	Product support:	
Customer acquisition	3	Service calls	6
Marketing	0	Repairs in the first 10 hours	4
Engineering:		Repairs in 10-100 hours	9
Changes	-6	Repairs after 100 hours	9
Releases	-8	Material planning:	
Product and component costing	-7	Prime product inventory	2
Order fulfillment	0	Scrap of surplus materials	0
Operations	0	Inventory scrap	0
Quality	0		

6.3.3 Temporal Effects for CAT

We then estimated the time lags for different cost pools, some time lags are naturally incorporated into the cost data based on the accounting rules, but some needed adjustments. Table

Table 6: Main volume drivers.

<i>Department/Volume driver</i>	Sales volume	Complexity offered	Complexity built
Purchasing	✓		✓
Customer acquisition	✓	✓	
Marketing	✓	✓	
Engineering		✓	
Order fulfillment	✓		✓
Product support	✓		✓
Material planning	✓	✓	✓
Operations	✓		✓
Quality	✓		✓

5 summarizes our analysis of time lag parameters. We use this information in Section 6.3.5 to estimate the parameters of the cost of complexity function.

6.3.4 Volume Effects for CAT

Similar to the time lags, we collected initial estimates of the primary volume drivers for different cost pools from the experts in each department, summarized the results, and held a group discussion to come to consensus. Table 6 provides a summary of the results.

6.3.5 Estimation of the VBC Effect

With a good understanding of the costs, time lags, and volume drivers, we collected data to estimate parameters ξ^d , α^d , ϕ^d , and γ_f^d for all d and f . We collected data from January 2001 to December 2005, for all cost measures summarized in the third column of Table 3. We then collected data from the price lists from 2000 to 2006 to capture all changes in option offerings, which were used as independent variables. (We had to collect a larger range of data due to the time lags identified in Section 6.3.3.) Similarly, we collected monthly sales (V_t) and the number of unique configurations sold per month (U_t) from 2000 to 2006.

The nature of the data suggested that there may be serial correlation, hence, we examined partial auto-correlation function plots and checked for auto-correlation using the generalized Durbin-Watson statistics using the AUTOREG procedure in SAS. For those cost pools having autocorrelation (all three “Cost of repairs” measures), we used the autocorrelation order identified by SAS (all three were a lag of one) and used the Yule-Walker approach to fit the data (Greene 2000).

Next, we obtained statistical models for all departments by fitting collected data to (2). We evaluated our models using both graphical and numerical tests using standard statistical techniques (e.g. examined the plots of residuals for normality, heteroscedasticity, and influential outliers). Although our cost data exhibited seasonality, the seasonality in cost (our dependent variable) is

Table 7: Fit results. * indicates statistically significant parameters at 0.05 significance level. Subscripts HC , C , CW and H stand for hydraulic combinations, cabs, counterweights, and hydraulics.

Fn.	Cost pool (d)	R^2	RMSE	ξ^d	α^d	ϕ^d	γ_{HC}^d	γ_C^d	γ_{CW}^d	γ_H^d
CA	Customer acquisition	0.37	0.18	962771120.4*	0.19			-1.13*		-1.68*
MP	Inventory (components)	0.29	0.16	1087617.41*	-0.03			1.04*		
MP	Inventory (prime product)	0.38	0.4	297.68	0.91*	0.91				
ENG	Engineering changes	0.41	0.52	0.00342591			4.81712*	-0.54		
ENG	Engineering product & component	0.55	0.9	6.39E-04*			4.107*		3.14*	
PS	Repair costs in the first 10 hours	0.29	0.37	26238.94*	-0.21	0.59*				
PS	Repair costs during 11–100 hours	0.32	0.24	3055.00*	-0.32	1.28*				
PS	Repair costs above 101 hours	0.55	0.47	934.63*	-1.41*	2.78*				
P	Supplier delivery performance	0.33	0.58	36.65		1.5*				

driven mainly by the seasonality in the volume (an independent variable) and, hence, it is likely to be automatically taken into account by our model. We checked this by analyzing residuals: In each model we group the residuals for each month; F-tests show that there are no statistically significant differences between the means of the groups. We also checked for significance, and only accepted those factors with reasonable coefficients of determination and low RMSE. Hence, not all of the originally identified departments were included in the final cost of complexity function. Table 7 summarizes all models/departments included in the optimization model. The estimates that are statistically significant at the 0.05 significance level are marked with an asterisk.³

We comment on the coefficient of determination (R^2) of the models in Table 7. Some of the departmental costs are heavily impacted by factors *outside* CAT’s walls: For example, cost of customer acquisition is impacted by competitors’ actions, and cost of supplier delivery performance is impacted by suppliers’ operations. Hence, we expect the coefficients of determination to be lower for such departments. In running the optimization model, nevertheless, in order to help ensure that our results are robust, we evaluate ranges of parameters as described in Section 7.

Some other findings in Table 7 are noteworthy. Increasing the number of options for hydraulics and cabs decreases the cost of customer acquisition, i.e., $\gamma_H^{CA} < 0$ and $\gamma_C^{CA} < 0$: A large proportion of this cost consists of sales variance or discounts given to customers in order to attract business. Cabs and hydraulics are very important considerations for customers; having a large selection of options for these features makes it easier to make a sale, decreasing the sales variance. This finding is in line with the intuition of sales and marketing representatives from CAT. Nevertheless, this was the first time that CAT was able to quantify this effect.

Another observation is that sales volume has a negative impact on product support costs (i.e. repair costs): When volume goes up, CAT employees assemble more machines with the same options, and learning effects reduce the number of mistakes. This intuition again seemed

³Some data is masked to preserve confidentiality. The signs and magnitudes have been preserved.

Table 8: ABC costs.

<i>Option</i>	δ_o	ω_o
Any option	\$1,000.00	
<i>In addition:</i>		
IT	\$6,000.00	\$23.61
Cab	\$14,656.25	\$53.80
E-Stick	\$125.00	\$1.30
One Way Line	\$3,593.75	\$5.64
Ride Control	\$1,281.25	\$5.86

plausible to CAT, but had never been quantified. On the other hand, sales volume has a positive impact on the cost of inventory of prime product. As CAT subsidizes dealers for carrying final product inventory for time sensitive customers, and the lead time demand increases when volume increases, subsidies increase. Guided by our study, CAT subsequently has performed similar cost of complexity analyses in other product divisions and obtained comparable results.

Finally, we validated our model by using the first two years of data to fit the model, then, compared the predicted values for the next three years to actual data. A large majority (7 out of 9) of the actual cost pool values were within the 95% confidence interval around their predictions.

6.3.6 Estimation of the ABC Effect

From focus groups we identified that ABC effects were observed primarily in three departments: assembly, production planning, and engineering. The nature of work in these departments suggested a linear relationship between ABC cost and complexity: If it takes 5 extra minutes to install a particular option on a machine, this cost will apply to each machine that contains the option. This coincided with expert opinion, which posited minimal learning effects. This supported the form of our proposed ABC_o function having a fixed and variable cost component.

The cost parameters δ_o and ω_o were estimated using expert opinions, time studies, and accounting information from all functional areas that identified this option as important (see Table 8).

6.4 CAT's Optimization Model

CAT used the optimization model in Section 5 without its lane-assignment features. In addition to the constraints described therein, CAT's optimization model includes two constraints that deal with profit margins. Let MIN_MARG and MIN_AVGMARG be, respectively, the minimum required profit margin on each configuration sold, and the minimum required average margin over all

configurations sold. The margin constraint per configuration is written as

$$C_j(1 + \text{MIN_MARG}) \leq B_j + \sum_{o \in \mathcal{O}_j} p_o, \quad \forall j \in \mathbf{J} \quad (16)$$

CAT provided a specific formula that they use to enforce the minimum average margin over all configurations sold. Besides the configuration cost C_j defined in Section 5, CAT uses another cost figure, denoted $C'_j \leq C_j$, which they refer to as *variable cost*. The sole purpose of C'_j is to enforce this average margin constraint⁴:

$$\text{MIN_AVGMARG} \sum_{i \in \mathbf{I}} \sum_{j \in \mathbf{L}_i} (r_{ij} + C_j q_j) \leq \sum_{i \in \mathbf{I}} \sum_{j \in \mathbf{L}_i} (r_{ij} + (C_j - C'_j) q_j). \quad (17)$$

The resulting optimization models have around 850,000 variables and 1.8 million constraints, they are solved using ILOG CPLEX Optimizer with default parameters. Typical solution times range from 6 to 8 hours, including preprocessing.

7 Results from Our Analysis

CAT's goal was to make a drastic reduction in the number of configurations offered without significantly reducing customer satisfaction or market share. How to achieve such a goal, or whether it was even possible, was unclear at the outset of the project. Since this reduction would present customers with fewer configurations, CAT assumed each remaining configuration could be priced a little lower; the reduced cost of complexity would allow this while maintaining profit.

Throughout our analysis, to ensure that our recommendations were robust we ran the optimization model across a range of parameters and migration list lengths.

7.1 Stage 1: Focusing on Configuration and Option Reduction

As an initial benchmark for our optimization, we first sought to identify the set of configurations that maximized profit at the current option prices, assuming limited customer migration (no more than a dozen configurations on a migration list, and no migration across models). While profits did increase, we obtained very little reduction in the number of configurations, except when a reduction was explicitly enforced by the constraint $\sum_{j \in \mathbf{J}} q_j \leq (1 - \text{MIN_CONF_RED})|\mathbf{J}|$. Moreover, forcing a large reduction resulted in a significant decrease of sales revenue.

Given these results, we hypothesized that further reducing the cost of complexity would require

⁴CAT did not explain the rationale behind (17); they just asked us to enforce this requirement.

significant cuts in *options*. Hence, a new constraint was added to the model to force option reduction: $\sum_{o \in \mathcal{O}} a_o \leq (1 - \text{MIN_OPT_RED})|\mathcal{O}|$. We re-ran the optimization model forcing a reduction in the number of configurations and the number of options. Reducing options did succeed in reducing configurations, in some solutions by as much as 94%, and increased profit by generating a large cost of complexity reduction. But it also resulted in a drop in sales volume of up to 67%. This disturbed CAT team members since the company has always prided itself on its market share. Fulfilling customer demand therefore became an important new metric of the analysis.

7.2 Stage 2: Opening Up Choices

Thus in the next phase of our analysis we wanted to explore what results would be possible if customers were presumed to be significantly more flexible, possibly as a result of price incentives. We modeled this flexibility in two ways: We increased the migration list length (to 100 configurations) and we enabled model-to-model migration.

This approach started to generate encouraging results. A solution emerged with an increase in profit of 8.8%, less than a 2% reduction in sales volume, and a reduction in configurations equal to 65%. But further analysis showed that the number of options had not decreased significantly. The increase in profit came from a decrease in the number of configurations, and increases in price paid by customers who migrated to slightly more expensive machines. Performing sensitivity analysis confirmed this conclusion: Expecting a large reduction in options was not realistic. However, a large reduction in configurations was possible.

This resulted in a problem for CAT: Without a reduction of options, how would this new (and limited) set of configurations be presented to the customers? Restaurants can get by with a 3-4 page menu that lists all their entrees. But no customer would flip through a menu of 70-90 pages listing all the possible BHL configurations. A new scheme had to be devised.

7.3 Stage 3: Standardization and Options Packages

We decided to try two new strategies to concentrate customer demand on a manageable number of configurations. The first strategy was standardization: Could options such as High Ambient Cooler and Engine Heater be made standard across all configurations? Optimization models with these options forced into every configuration yielded cost reductions that justified a reduction in price large enough to make the standardized configurations attractive to customers, while maintaining sales volumes and profit. Other rarely used options were eliminated using a similar approach. For example, the Cab/Canopy options were cut from five to two.

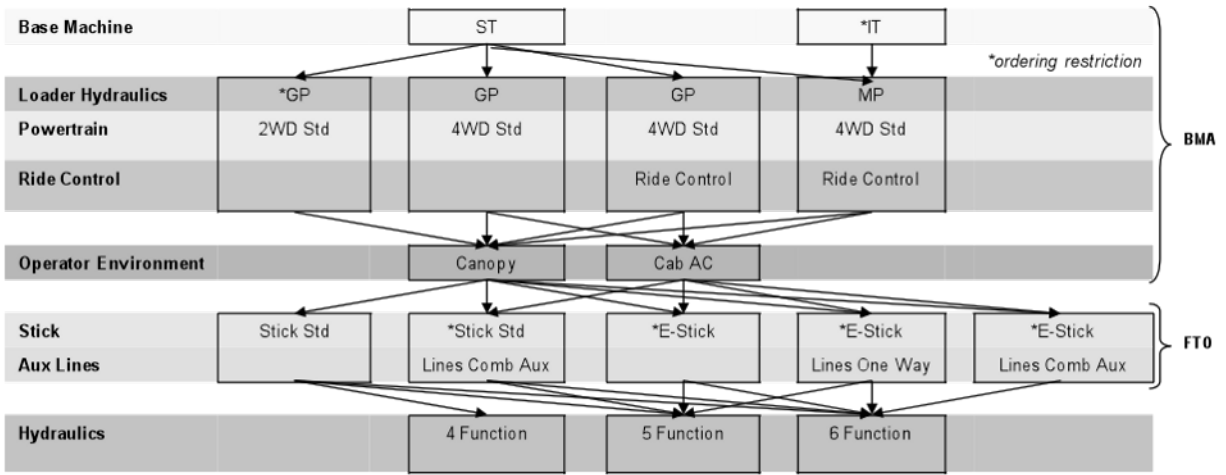


Figure 2: Final product hierarchy and packages for the BHL 420E series.

The second strategy was creating *packages* of options commonly found together. For example, guided by customer segment preferences, a single pair of loader hydraulics and powertrain options most likely to meet each segment's needs was proposed. Manual inspection and cluster analysis of the best solutions found so far led to the discovery of other options often found together.

The optimization was then run assuming standardization and option packaging, with constraints on the maximum price increase. This yielded the final product hierarchy for the 420E series, shown in Figure 2: It consists of 9 base-machine-assembly (BMA) packages, 5 finished-to-order (FTO) packages, and 3 hydraulics options, for a total of 135 possible configurations, some of them anticipated to be much more popular than others.

Pricing optimization showed that with these 135 configurations, revenue from sales could increase by almost 7%, and profit could increase by 15%, *with 99.6% demand fulfillment*. The standardization and bundling of options greatly reduced the universe of feasible configurations, which led to inventory and quality savings: 76% of the projected cost of complexity savings came from reductions in finished goods inventory and warranty costs (in particular, the cost of addressing failures in the first 100 hours of machine operation). Since the goal of the project was to maintain similar profit levels (rather than seeking increases), the team determined appropriate option price *reductions* to drive dealer behavior toward these configurations while maintaining profit. The proposed pricing policy resulted in an anticipated reduction in profit from sales of 4%, which was easily made up by the reduction in cost of complexity to yield a total profit increase of 4.8%. We had finally found the very small subset of configurations that we believed was broad enough to satisfy CAT's customers and dealers, but also focused enough to drive operational and

supply chain efficiencies. This final recommendation was presented to CAT, and was approved.

8 Implementation Details

8.1 Initial Implementation

CAT put an updated price list for the 430E product line (modified via dealer input to now feature 124 configurations) into effect in some regions in April 2010; but, in order to minimize the risk of lost sales, customers could still order a-la-carte machines according to the old price list. Concurrent with this new price list, CAT introduced a 3-lane strategy for order fulfillment:

- Lane 1 (the fastest lane): Orders on the four designated “most popular” fully configured machines would be satisfied within a few days;
- Lane 2: Orders on the remaining 120 choices, broken into the *Loader*, *Comfort and Convenience*, and *Excavation* packages, would be satisfied within a few weeks;
- Lane 3: Any a-la-carte machine would be satisfied within a few months, as previously.

Upon implementation, CAT experienced positive dealer feedback and large reductions in the number of unique configurations sold. This contrasted with previous attempts to reduce the size of the product line, based on a Pareto analysis of the top few dealers. These had likely been ineffective, in CAT’s opinion, because they did not effectively model the interaction of cost, customer preferences, and substitution, as we had.

It is worth noting that, to the best of our knowledge, neither standardization nor option packaging were being considered prior to the start of our project. These concepts, and the lane strategy they support, were only developed *after* our analysis indicated that CAT needed to find a way to limit the number of configurations while *not* eliminating too many options. Our analysis then gave CAT direction in implementing these new strategies, by indicating which sets of configurations were likely to be most successful.

This evolution of the solution strategy also affected how the results of our algorithm were applied. Specifically, we initially modeled a problem setting in which there would be a reduced set of products offered to customers differentiated by price, but not by lead time. The solution CAT implemented (in principle) still includes all possible configurations within Lane 3, and differentiates these from the first two lanes by lead time. Thus our forecasts of the savings in cost of complexity, which would be accurate if only the first two lanes were offered, are only an approximation given

the continued possibility of a-la-carte ordering. Caterpillar felt that the flexibility of retaining Lane 3 outweighed any reductions in cost of complexity savings. Likewise, although lead time differences could be incorporated into our framework as shown in Sections 3 and 5, Caterpillar felt comfortable enough with the main conclusions of the analysis to forgo additional analysis.

8.2 Moving Forward

In 2011 CAT released a single price list featuring options packages for use in all regions. They were able to capture over a quarter of their sales volume just in Lane 1; in contrast, the four top-selling configurations CAT offered before this project captured 11.3% of both sales and total revenue for the 420E model (Figure 1). In addition, CAT enjoyed a reduction in warranty costs, attributable to many factors, including this project. CAT has continued to focus their BHL offerings, for example reducing the 420E (now the 420F) Lane 1 offerings to three base machines by mid 2014. The other BHL lines have seen similar reductions.

The lane approach has become an integral part of CAT’s business strategy (Thomson Reuters (2014)). While the methodologies used to determine lane offerings in other divisions were somewhat simpler than the analysis done here, our work provided support for the corporate-wide lane strategy. In particular, our cost of complexity analysis approach has been applied to other divisions, such as Wheel Loaders, with the goal of capturing all the benefits and understanding all the consequences of proposed line changes. The detailed analysis and structured optimization approach reported in this paper allowed CAT to counteract skepticism toward the lane approach embarked upon by BCP prior to the diffusion of this strategy throughout the company.

9 Sensitivity Analysis and Insights

In order to generate additional insights from our model, we conducted an extensive sensitivity analysis of our framework by running a large number of experiments in which we vary the main input parameters over a range of values and record key output measures. Because of space limitations, we do not include all of the experiments here; they are available in a separate online supplement to this article. This section summarizes the main findings generated by this analysis.

9.1 Modeling Customer Behavior

9.1.1 Customer Flexibility: The More the Better

As customers become more flexible and willing to accept a wider range of configurations, the expected effects include a decrease in the number of configurations needed to satisfy them, a

reduced number of required options, decreased complexity costs, and higher profits. These trends are supported by our experiments; specifically, when any of the following parameters increases in value: maximum migration list length, reservation price, disparity factor, and percentage of novel configurations in the portfolio. Varying the number of features whose utilities are perturbed and/or the amount of perturbation around the point estimate (perturbation factor) seems to add some noise, but does not ultimately change the observed outcomes significantly.

9.1.2 Role and Effect of Migration Lists

While performance improves with longer migration lists, we note that the effects of greater list length is more pronounced for smaller lists—once lists are of moderate size (about forty) metrics are largely insensitive to further increases in length (see Figure 3, additional figures and discussion are provided in the Supplement, Section 1). Furthermore, more choices lead to purchases being more spread through migration lists, decreasing the number of top-ranked purchases, as these are knocked out of the portfolio in favor of more universally appealing configurations.

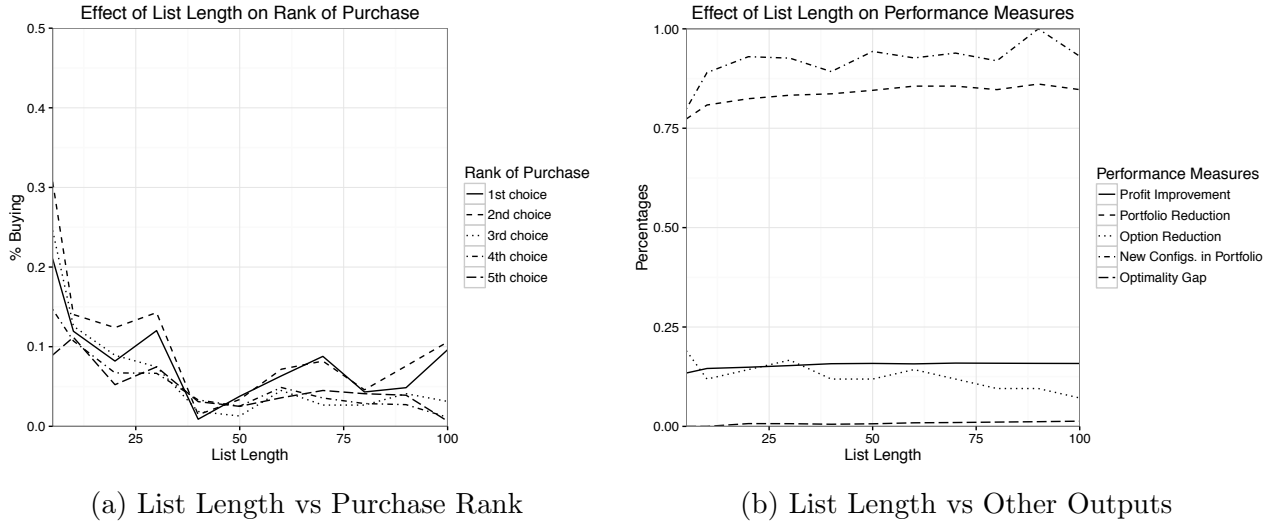


Figure 3: Effect of Migration List Length

To better explain the role of migration lists, Figure 4 shows how many times, out of 3825 customers, a customer’s original choice ended up at a given position on that customer’s (100 configuration long) migration list (note the log scale on the vertical axis). The rightmost, tallest bar indicates that for 2601 customers (68% of the time) a customer’s original choice would not have appeared anywhere in the first 100 positions of the customer’s list. For CAT this means that, for over two thirds of their customers, there are many products that provide them with higher utility than the first product they had in mind. Figure 4 emphasizes that, within the context of product portfolio reduction, the main purpose of migration lists is *not* to predict what a given customer

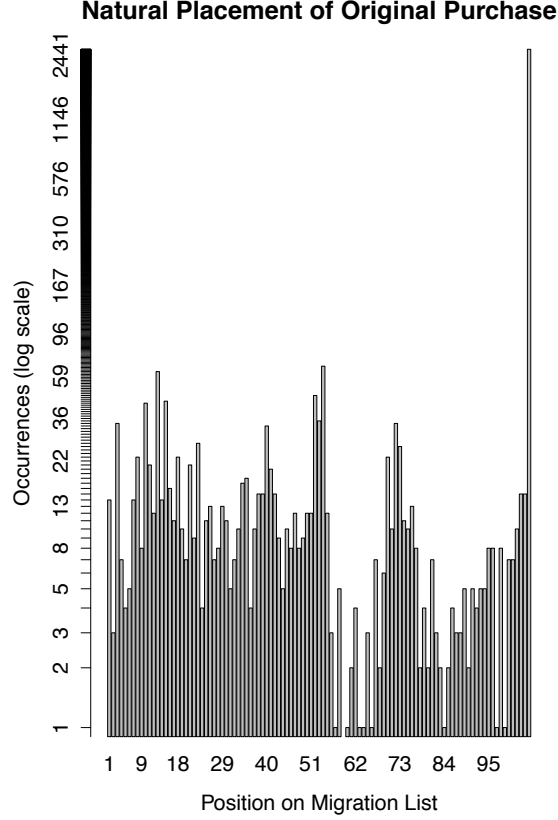


Figure 4: Number of migration lists, out of 3825, containing original purchase.

would buy. Rather, the migration list’s job is to determine which products would provide high utility to each customer, thus forming a pool of configurations from which to select the ultimate product portfolio.

To explore this point further, consider the following thought experiment: Assume a company had a perfect forecasting algorithm that could always guess exactly what any customer’s first choice of product would be. Despite being useful for several things (such as targeted advertising, as well as production and inventory planning), if the universe of customers’ first choices were very heterogeneous, this algorithm would not allow the company to reduce the size of its product portfolio because it would not provide *any* information about customers’ flexibility and willingness to substitute. Instead, a migration list, as defined in our framework, tries to predict, *given* a customer’s first choice of product, what *other* products would likely be acceptable to that customer. In doing so, if a large number of customers happen to like the same not-so-large collection of products, there is a chance that significant savings can be achieved by focusing the portfolio on that smaller collection, even if some of those products are not the first choice of many, or even any, of the original customers.

9.1.3 Varying the β Factor

As β goes up, the probability of buying the first configuration on the migration list goes down. This is likely because many of the originally purchased configurations are pruned from the portfolio in an effort to concentrate customers. As for profit, it is largely insensitive to β , even though the composition of the portfolio may change.

9.1.4 Reservation Price vs. Reservation Utility

As expected, when customers are willing to pay more, everything improves for the company. Customers whose β factor does not force their originally purchased configuration to appear first on the list are more likely to buy their top choice, as it will likely be a high-utility, high-price machine. The remaining customers are more likely to purchase configurations further down their migration lists, as their top, lower utility/lower priced choices get pruned. In contrast, having customers willing to accept lower utility machines is not as impactful as their becoming less price sensitive. That is because accepting machines with lower utility does not remove the higher utility machines from consideration (which the company would typically prefer to sell anyway), and the latter get placed ahead of the lower utility machines on the migration lists.

9.1.5 Role and effect of the number of lanes

The concept of lanes introduced in Section 8.1 adds another dimension to a configuration's attractiveness: *Availability*, that is, how quickly a customer can obtain it. Availability can be treated as a feature whose options are fastest, second fastest, etc. As a feature, availability would then have a relative importance and its options would have utilities. And, given these utilities, the company may choose to either charge a premium/markup for configurations in faster lanes or keep the prices the same. We ran a set of experiments to understand what happens to customers' migration lists and to several output measures related to the optimal product portfolio when different numbers of lanes are present (e.g. the company decided to offer 3 lanes instead of 2). For the optimization algorithm, what matters are the configurations on each migration list, as well as the order in which they appear. Hence, we simulate the presence of the availability feature by exploring the composition of migration lists under different lane scenarios. One benefit of this approach is that it does not require us to choose exact values for different costs and profits related to different lanes, as would be required to run a full-blown optimization.

For our experiments we first need to characterize customers based on two dimensions: willingness-to-pay (low, medium, high) and the utility they assign to the availability feature (low, medium, high). Note that these are two different things: A customer may prefer to have high availability,

but be unwilling to pay extra for it. As an example of how we categorize customers, consider a case with three lanes where Lane A is the fastest and Lane C is the slowest. A customer who has low willingness-to-pay (wtp) and low utility for availability ($avail$), will only have the slowest lane on his migration list (C), and a customer who has low wtp and medium $avail$, may have a migration list comprised of configurations in lanes C and then B (CB). For our experiments, we consider 10 different values for the Utility of Availability (specifically, from 1 to 10), such that: a customer with low $avail$ is mapped to value of 1, medium to 5, and high to 10. The values in between capture the gradual transition in the proportion of customers switching to the type of migration list: namely, for customers with low wtp , when the utility of availability equals 1, all customers have only one migration list (C), but as the values increase from 1 to 5, the proportion of customers switching to list CB increases. The exact proportion of customers corresponding to each utility value depends on the selected percentage of customers having different wtp ⁵. In addition, we consider cases with and without a price markup on faster lanes as some companies may have no markup and use the lanes solely to incentivize customers to purchase certain subsets of products, while others may put a markup on the configurations offered in faster lanes.

In this section, we report the results on the effect of utility of availability for an experiment without price markup, where the customers are split equally between the wtp values⁶. We provide additional results from multiple experiments in the Supplement in Section 2. In Figure 5(a), we demonstrate what percentage of profit is attributable to each lane as the utility of availability changes. Notice that as the utility of availability increases, Lane 3 (the slowest lane) gets utilized less and eventually disappears. This observations persists across scenarios with different markups and different distribution of customers across wtp values. In Figure 5(b), we compare the profit of five replications of the portfolio with 2 lanes vs the portfolio with 3 lanes and show that when the utility of availability is low, the 2-lane portfolio provides a higher profit than a 3-lane portfolio, but the opposite is true when the utility of availability is greater.

This arises because the overall portfolio size of the two lane portfolio stays comparatively more stable than the three lane portfolio for smaller utilities – there is just less room for movement. But once availability becomes more valuable, there is greater dispersion among the customer base, and thus the three lane portfolio has an easier time accommodating this: While both portfolios grow,

⁵We describe this process in great detail in the Supplement in Section 2

⁶Please note that even without markups, we enforce the 9 types of lists, taking wtp into account. We do this assuming that the company would first solve this no-markup version of the problem to find a solution with a higher complexity cost reduction; and once the optimal portfolio and lane assignments are decided, they can increase prices because they already know people are willing to pay a little premium.

the two lane portfolio grows more abruptly, lowering the profits⁷. Note that these experiments were run for the case of zero markup – as expected, when markup is permitted profit increases with the utility of availability, as customers are steered toward more expensive, more available, machines. Note finally, that these effects remain present, but are mitigated, if greater dispersion of customers is assumed.

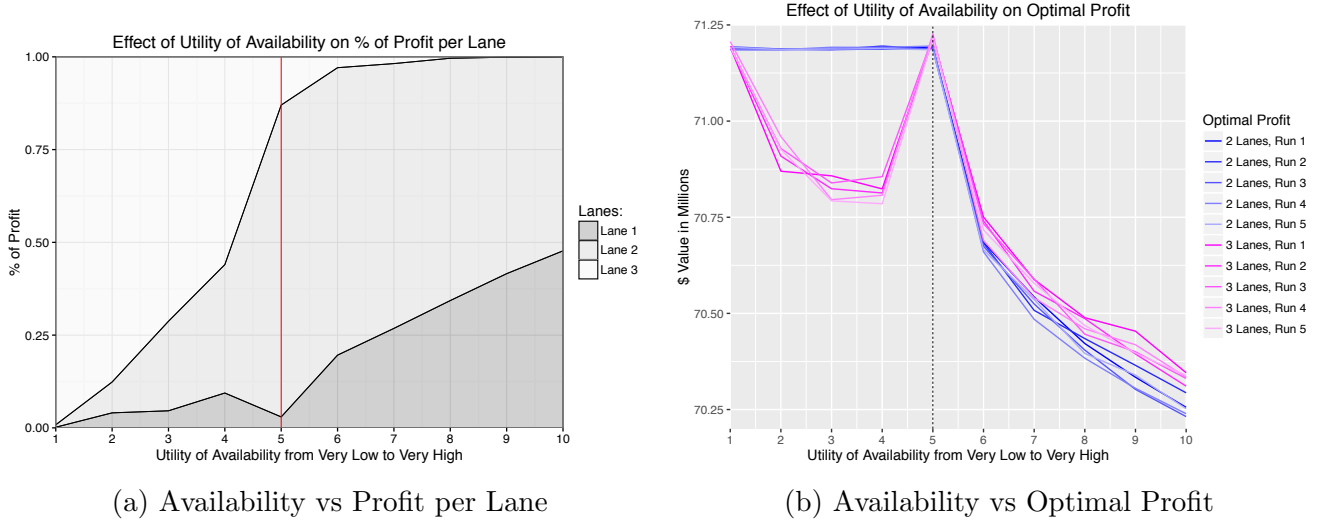


Figure 5: Effect of the Utility of Availability

9.2 Constraints on Sales Volume and Unique Configurations

For CAT, our experiments indicated that the best course of action would be to lose as much sales volume as permitted (see Figure 6). (Recall that early profit maximizing solutions decreased sales volume by as much as 67% and were thus rejected.) This was driven primarily by lower volumes creating savings in inventory cost and quality control: quality level is impacted highly by the total volume (not so much by options) as you have more time to spend on machines when the volume is low. Inventory cost is also reduced based on volume because lower volume implies less inventory held at dealers and, hence, CAT has to subsidize less. In CAT’s specific case, these two cost pools have a large enough impact to explain the change in profit. This is why the strategic constraint on market share is so important—CAT wants to hold the line on market share, which restricts the portfolio reductions they will tolerate. In contrast, the number of unique configurations can be easily reduced by up to 80% in CAT’s case, without negative effects on other performance measures, indicating it is not nearly as important as the minimum market share requirement.

⁷Please refer to Section 2 of the Supplement for the plots illustrating this and following observations

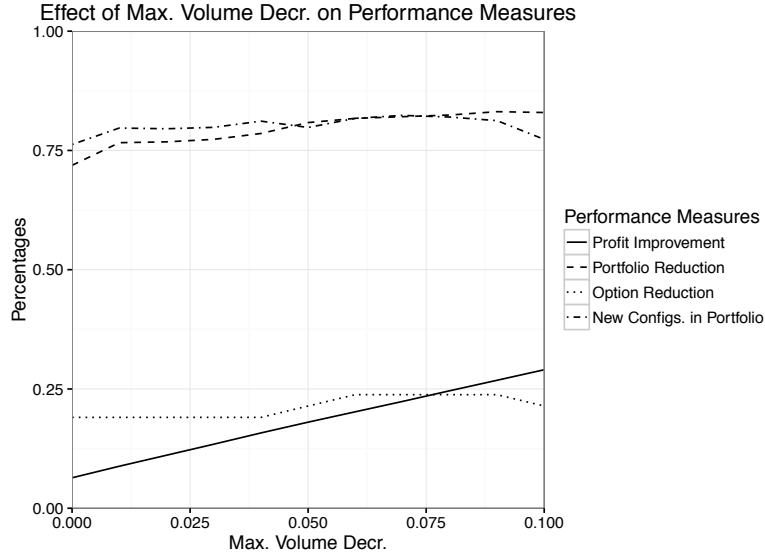


Figure 6: Maximum Sales Volume Decrease vs. Other Outputs

9.3 Insights Related to Cost of Complexity

Section 6.3.5 provided evidence for two insights that, although intuitive, had never been empirically verified by CAT. First, for some features, increasing the number of options decreases the costs of customer acquisition. Second, sales volume has a negative impact on product support costs (i.e. repair costs) and, at the same time, a positive impact on the cost of prime product inventory. In addition, we emphasize three complexity-related takeaways from our analysis: (i) There are two types of impact of product portfolio complexity on cost, which need to be modeled separately: complexity driven by the number of options offered (VBC) and complexity driven by specific options (ABC); (ii) There are time lags and leads on the impact of complexity that may or may not be reflected in cost data is depending on the accounting methods used; (iii) Costs can be impacted in either direction (increase or decrease) by complexity.

9.4 Data Collection

As we collected data to implement and run our models, we found some data particularly useful, while there were other instances when we wished we had certain types of data that were not available, forcing us to go with the next-best thing. Hence, as an added insight, we believe that tracking the following data, whenever possible and practical, would enhance a company's ability to improve their product portfolio: (i) Whether or not the product purchased by a customer was the one she originally had in mind to buy and, if not, record both the original choice and the final purchase; (ii) Record what options/features the product support costs are associated with.

Having this data could have revealed that some options had an ABC effect on the cost pool.

10 Conclusion

We present a three-step procedure to restructure a product line, demonstrating its successful application on the Backhoe Loader line at CAT. Our methodology hinges on (i) the construction of migration lists to capture customer preferences and willingness to substitute; (ii) explicitly capturing the (positive and negative) cost of complexity of a specific product line across different functional areas; and (iii) integrating these tools within a mathematical programming framework to produce a final product line. One of the greatest strengths of our methodology is its flexibility—each step can be tailored to a company’s particular setting, data availability and strategic needs, so long as it produces the necessary output for the next step.

CAT’s lane strategy has continued to evolve, for example, Thomson Reuters (2014) discusses a new variation in which lanes may contain only partially completed machines, which can be finished to customer order as needed. Ideas such as these offer opportunities to extend our work to new problem domains—analytically characterizing the performance of such a delayed differentiation strategy is an exciting and challenging problem.

Outside of CAT, our work can be extended in several directions. First, explicit experimental validation of our empirical models—in particular our migration list approach—would be of value. While this approach has been successfully applied in the construction equipment industry, further study could help establish how it could be applied in other sectors, and what changes might be necessary. For example, we set the list lengths based on consultation with CAT executives. A better understanding of how long such lists really are, and how willing customers are to substitute (and whether there is any sort of explicit cost to this) would be of interest.

Second, as our methodology is applied to other settings, new constraints might need to be incorporated into our mathematical program. One of the benefits of our procedure is that our optimization model is flexible enough to accommodate such constraints. Nevertheless, how it performs in other settings needs to be established.

Finally, the central thrust of this paper has considered the trade-off between cost of complexity and product line breadth. CAT has found what they believe to be the correct trade-off, which entailed a dramatic reduction in their product line. Different companies, in different industries, will have to answer this question for themselves. It is our hope that the methods we present in our paper can likewise help them find the answers they seek.

References

- A. Alptekinoglu and C. Corbett. Mass customization vs. mass production: Variety and price competition. *Manufacturing and Service Operations Management*, 10(2):204–217, 2008.
- S. Anderson. Measuring the impact of product mix heterogeneity on manufacturing overhead cost. *The Accounting Review*, 70(3):363–387, 1995.
- R. Banker, G. Potter, and R. Schroederr. An empirical analysis of manufacturing overhead cost drivers. *Journal of Accounting and Economics*, 19:115–137, 1995.
- A. Belloni, R. Freund, M. Selove, and D. Simester. Optimizing product line designs: Efficient methods and comparisons. *Management Science*, 54(9):1544–1552, 2008.
- G. Bitran and J.-C. Ferrer. On pricing and composition of bundles. *Production and Operations Management*, 16(1):93–108, 2007.
- S. Broniarczyk, W. Hoyer, and L. McAlister. Consumers’ perceptions of the assortment offered in a grocery category: The impact of item reduction. *Journal of Marketing Research*, 35:166–176, 1998.
- K. Chen and W. Hausman. Technical note: Mathematical properties of the optimal product line selection problem using choice-based conjoint analysis. *Management Science*, 46(2):327–332, 2000.
- Y. Chen, A. Vakharia, and A. Alptekinoglu. Product portfolio strategies: The case of multifunction products. *Production and Operations Management*, 17(6):587–598, 2008.
- Y. Chen, J. Carillo, A. Vakharia, and P. Sin. Fusion product planning: A market offering perspective. *Decision Sciences*, 41(2):325–353, 2010.
- J. Chong, T. Ho, and C. S. Tang. Product structure, brand width, and brand share. In T. Ho and C. S. Tang, editors, *Product Variety Management: Research Advances*, pages 39–64. Kluwer Academic Publishers, 1998.
- M. Fisher and C. Ittner. The impact of product variety on automobile assembly operations: Empirical evidence and simulation analysis. *Management Science*, 45(6):771–786, 1999.

- M. Fisher and R. Vaidyanathan. An algorithm and demand estimation procedure for retail assortment optimization. *Working Paper, University of Pennsylvania*, 2011.
- G. Foster and M. Gupta. Manufacturing overhead cost driver analysis. *Journal of Accounting and Economics*, 12:309–337, 1990.
- G. Fruchter, A. Fligler, and R. Winer. Optimal product line design: Genetic algorithm approach to mitigate cannibalization. *Journal of Optimization Theory and Applications*, 131(2):227–244, 2006.
- W. H. Greene. *Econometric Analysis*. Prentice Hall, New Jersey, fourth edition, 2000.
- P. M. Guadagni and J. D. C. Little. A logit model of brand choice calibrated on scanner data. *Marketing Science*, 2(3):203–238, 1983.
- J. R. Hauser and V. Rao. Conjoint analysis, related modeling, and applications. In J. Wind and P. Green, editors, *Advances in Market Research and Modeling: Progress and Prospects*, pages 141–168. Kluwer Academic Publishers, Boston, MA, 2004.
- G. Kok and M. Fisher. Demand estimation and assortment optimization under substitution: Methodology and applications. *Operations Research*, 55(6):1001–1021, 2007.
- G. Kok, M. Fisher, and R. Vaidyanathan. Assortment planning: Review of literature and industry practice. In N. Agrawal and S. Smith, editors, *Retail Supply Chain Management*, pages 1–55. Kluwer Academic Publishers, Amsterdam, The Netherlands, 2009.
- K. Marriott and P. Stuckey. *Programming with Constraints: An Introduction*. MIT Press, 1998.
- E. Rash and K. Kempf. Product line design and scheduling at Intel. *Interfaces*, 42(5):425–436, 2012.
- T. L. Saaty. *The Analytic Hierarchy Process*. McGraw-Hill Book Co., New York, 1980.
- C. Schoen. On the optimal product line selection problem with price discrimination. *Management Science*, 56(5):896–902, 2010.
- C. Tang. A review of marketing-operations interface models: From co-existence to coordination and collaboration. *International Journal of Production Economics*, 125(1):22–40, 2010.

- C. Tang and R. Yin. The implications of costs, capacity, and competition on product line selection. *European Journal of Operational Research*, 200(2):439–450, 2010.
- Thomson Reuters. Caterpillar Inc. 2014 Analyst Meeting at ConExpo edited transcript, 2014.
- X. Wang, J. Camm, and D. Curry. A branch-and-price approach to the share-of-choice product line design problem managing product variety. *Management Science*, 55(10):1718–1728, 2009.
- J. Ward, B. Zhang, S. Jain, C. Fry, T. Olavson, H. Mishal, J. Amaral, D. Beyer, A. Brecht, B. Cargille, R. Chadinha, K. Chou, G. DeNyse, Q. Feng, C. Padovani, S. Raj, K. Sunderbruch, R. Tarjan, K. Venkatraman, J. Woods, and J. Zhou. HP transforms product portfolio management with operations research. *Interfaces*, 40(1):17–32, 2010.
- T. H. Yunes, D. Napolitano, A. Scheller-Wolf, and S. Tayur. Building efficient product portfolios at John Deere and Company. *Operations Research*, 55(4):615–629, 2007.

Online Supplement for the Manuscript Entitled

Product Portfolio Restructuring: Methodology and Application at Caterpillar

Authors: M. Shunko, T. Yunes, G. Fenu, A. Scheller-Wolf, V. Tardif, and S. Tayur

1 Sensitivity Analysis with One Lane

In this section we investigate how sensitive a number of key outputs produced by our framework are with respect to its main input parameters, for a single lane. In doing so, we bring to light several managerial insights that can be useful to companies considering reducing their product portfolio. We consider sensitivity across multiple lanes in Section 2.

1.1 Experimental Settings

Table 1 shows the default values of the main parameters in our experiments. In each of our sensitivity analysis tests, parameters whose values are not being altered for the test are set to their default values. Optimization models were run with a time limit of two hours, except for the tests in Section 1.4, which ran for up to 24 hours each, which was necessary to obtain meaningful output with the longer migration lists in that section. When a positive optimality gap is shown, we report the results of the best solution found within the given time limit.

Parameter	Default Value
β factor: Probability original purchase is first on migration list	50%
Maximum migration list length	5
Customers' reservation price: max. acceptable price increase	2%
Customers' reservation utility: max. acceptable utility decrease	10%
Disparity factor: max. # features differing from original purchase	5
# features to have importance score perturbed	3
Perturbation factor for features' importance score (\pm)	10%
Maximum decrease in sales volume	3%
Minimum decrease in # unique configurations sold	0%
Maximum percentage of novel configurations in portfolio	100%
Maximum price increase per configuration (when price optimization is on)	0.5%
Minimum margin per configuration	2%
Minimum weighted average margin over all configurations	15%

Table 1: Default values of the main list-generation and optimization parameters.

1.2 Placement of Original Purchase on Migration Lists

Figure 1 shows how many times, out of 3825 customers, a customer's original choice ended up at a given position on that customer's (100 configuration long) migration list (note the log scale on the vertical axis). The rightmost, tallest bar indicates that for 2601 customers – 68% of the time – a customer's original choice would not have appeared in the first 100 positions of the customer's migration list. For Caterpillar this means that, for over two thirds of their customers, there are

many products that provide them with higher anticipated utility than the first product they had in mind.

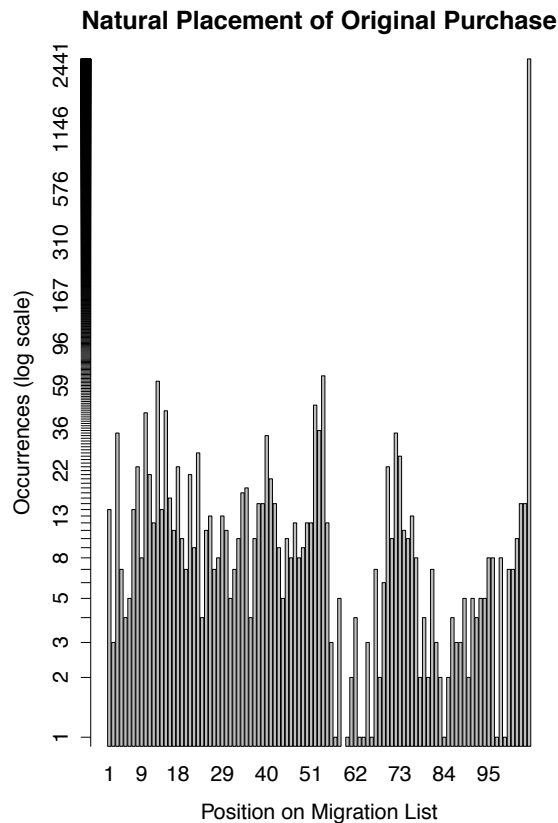


Figure 1: Number of migration lists, out of 3825, containing original purchase.

Figure 1 emphasizes that within the context of product portfolio reduction, the main purpose of migration lists is *not* to predict what a given customer would buy. Rather, the migration list's job is to find out which products would provide high utility to each customer, thus forming a pool of configurations from which to select the ultimate product portfolio. To see why, assume a company had a perfect forecasting algorithm that could always guess exactly what any customer's first choice of product would be. Despite being useful for several things (such as targeted advertising, as well as production and inventory planning), if the universe of customers' first choices were very heterogeneous, this algorithm would not allow the company to reduce the size of its product portfolio because it would not provide *any* information about customers' flexibility and willingness to substitute.

Instead, a migration list, as defined in our framework, tries to predict, *given* a customer's first choice of product, what *other* products would likely be acceptable to that customer. In doing so, if a large number of customers happen to like the same not-so-large collection of products, there is a chance that significant savings can be achieved by focusing the portfolio on that smaller collection, even if some of those products had not been the first choice of many, or even any, of the original customers.

1.3 Varying the Beta Factor

We also investigated how the percentage of customers buying their 1st (2nd, 3rd, 4th, or 5th) choice changes with the probability that the purchased configuration is forced to be first (Beta Factor). We observed that as the Beta Factor goes up, the probability of buying the first configuration goes down, likely because many of the originally purchased configurations are pruned from the portfolio in an effort to concentrate customers. We also observed that the profit improvement is largely insensitive to the Beta Factor: Again, probably because most of the originally purchased configurations get pruned, it does not really matter much where they reside on the lists.

1.4 Varying Migration List Length

We next varied the length of the migration list from 1 to 100 to see the impact of such change on various performance measures. We observe from 2(a) that more choices lead to purchases being more spread through migration lists, decreasing the number of top-ranked purchases. Looking also at the portfolio size (no figure), we observe that longer lists are an indication of greater customer flexibility which, as expected, leads to optimal portfolios with fewer configurations. From 2(b), we observe that longer lists correlate, as expected, with improvements in profit, increased portfolio reduction, decrease in options needed, increased number of new configurations, etc. These all follow from the fact that longer lists correspond to more flexible customers. We note though that the increase in profit with list length is more pronounced for smaller lists — once lists are of moderate size (about forty) profit is largely flat with further increases in length.

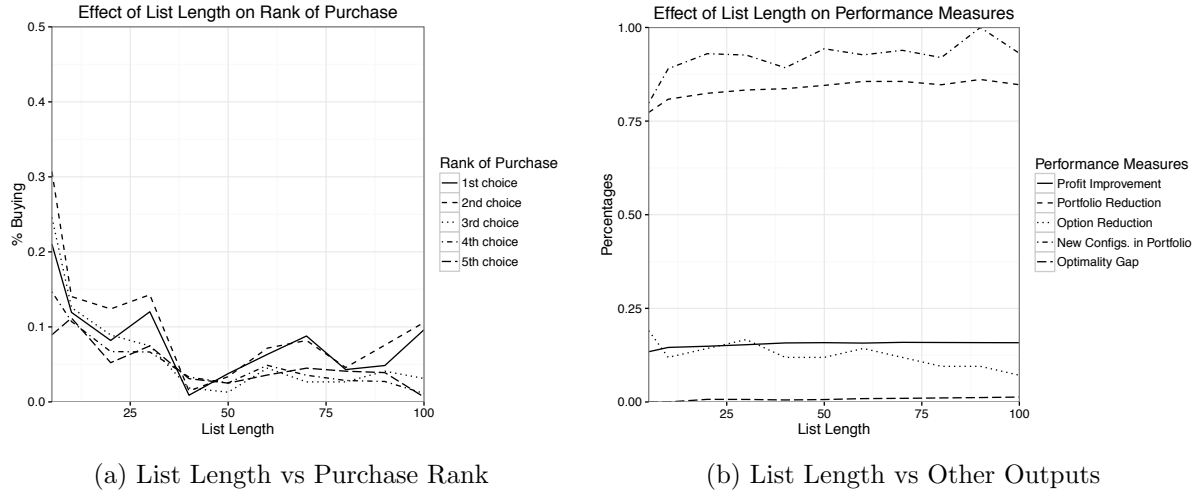


Figure 2: Effect of Migration List Length

1.5 Varying Customers' Reservation Price

Next, we look at the performance measures as the customers' reservation price changes. We observe from Figure 3(a) that as customers become more flexible, they are more likely to be steered away from their top choice (which with probability $\beta = .5$ is the purchased configuration) to a more expensive one. Customers whose β did not force their purchased configuration to appear first on the list are more likely to buy their top choice, as it will likely be a high-utility, high-price machine. Clearly, when customers are willing to pay more, everything improves for the company as is confirmed by the plot in Figure 3(b).

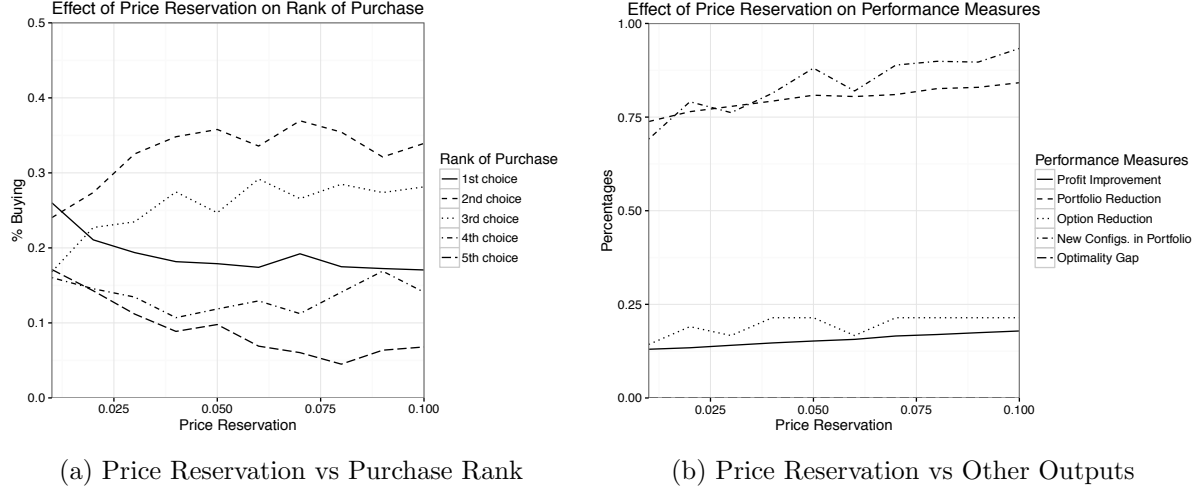


Figure 3: Effect of Customers' Reservation Price

1.6 Varying Customers' Reservation Utility

We also investigated the impact of the customer's reservation utility on our main performance measures and found that they are mainly insensitive to the changes. This insensitivity can be explained as follows: the fact that customers accept machines with lower utility does not remove the higher utility machines from consideration, and the latter get placed ahead of the lower utility machines on the migration lists as usual.

1.7 Varying the Disparity Factor

We now study the impact of disparity factor. We observe in Figure 4(a) that the first-choice purchase line is down dramatically, for reasons similar to those discussed in 1.5: For those customers for whom their purchased configuration is forced to be their first choice, greater flexibility gives us greater ability to steer them to other configurations, concentrating the portfolio. In Figure 4(b) we observe that as customers become more flexible in terms of disparity, lists become more concentrated on key configurations, driving up these measures. Relatively little disparity seems to go a long way.

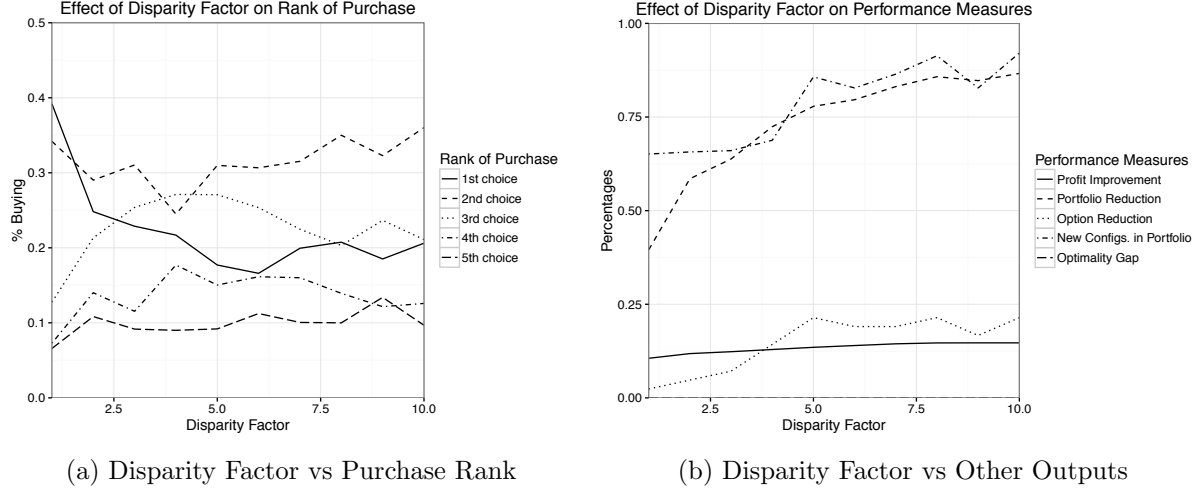


Figure 4: Effect of the Disparity Factor

1.8 Varying the Maximum Sales Volume Decrease

In this section, we look at the impact of the maximum sales volume decrease on the rank of purchase and the performance measures. While there seems to be no substantial impact on the rank of purchase (Figure 5(a)), there is a strong effect of the maximum allowed volume decrease (lost sales) on the profit increase (Figure 5(b)). The constraint on sales volume decrease was binding for all optimal solutions, indicating that the best course of action was to lose as much sales volume as possible, benefiting from reduced complexity costs stemming from inventory cost reduction and quality control savings: Quality level is highly impacted by the total volume (not so much by options) as there is more time to spend on machines when the volume is low. Inventory cost is also reduced based on volume, because lower volume implies less inventory held at dealers and hence, CAT has to subsidize less. In CAT's specific case, these two cost pools have a large enough impact to explain the change in profit.

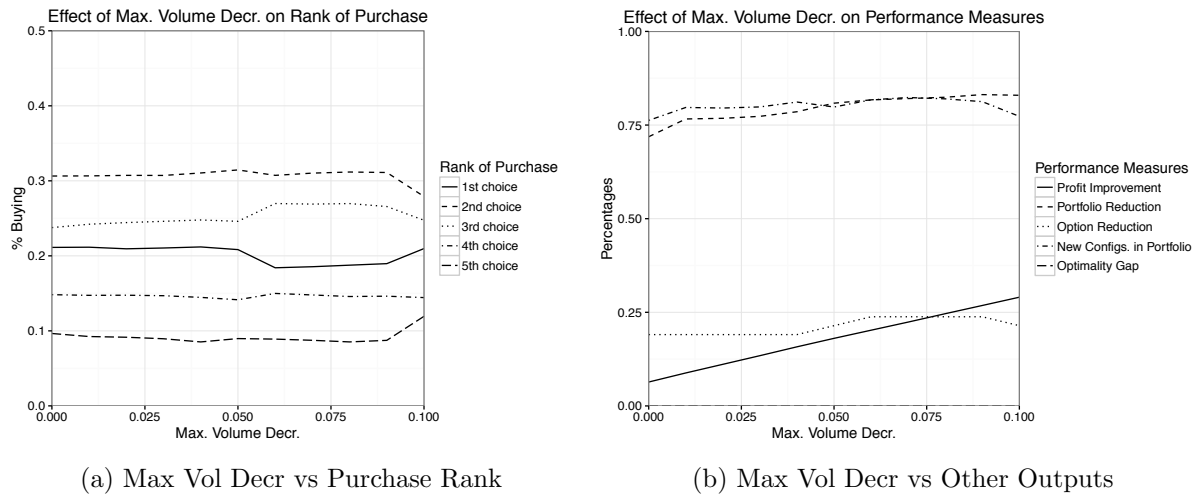


Figure 5: Effect of the Maximum Sales Volume Decrease

1.9 Varying the Minimum Decrease in Unique Configurations

Another potential constraint could be on the minimum number of configurations removed from the portfolio, which we study next.

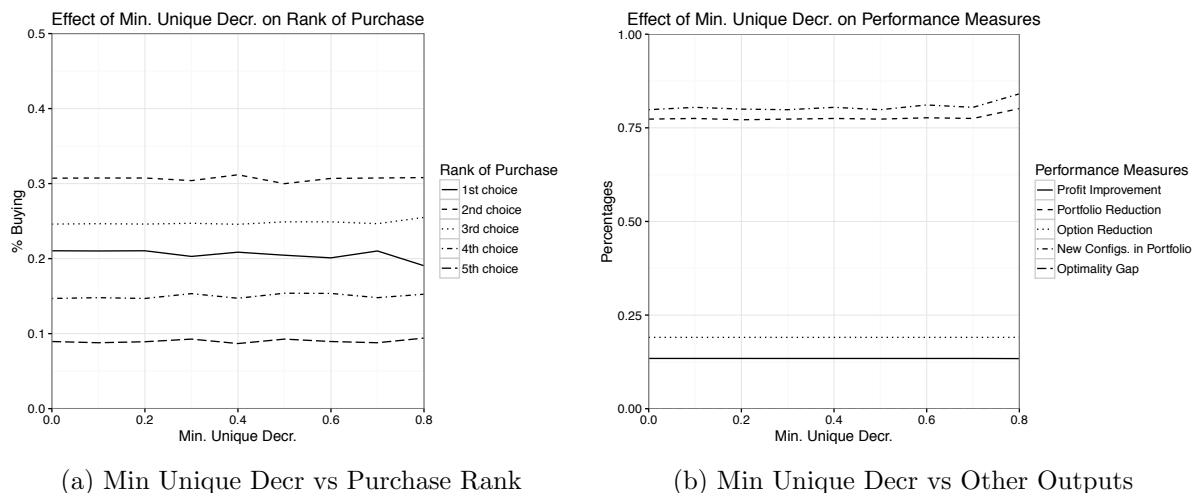


Figure 6: Effect of Minimum Decrease in Unique Configurations

In this case all curves are relatively flat, indicating that the number of unique configurations can be easily reduced by at least 80% in CAT's case, without side effects on other performance measures. Together with the previous graph this emphasizes that minimum market share is a much more important constraint than decrease in configurations.

1.10 Varying the Maximum Price Increase per Configuration, the Minimum Margin per Configuration, and the Minimum Average Weighted Margin over All Configurations

We next attempted to study the impact of the maximum price increase per configuration, the minimum margin per configuration, and the minimum average weighted margin over all configurations. However, the optimization problem becomes very hard to solve because the solution space expands dramatically as price becomes a more important lever, and in fact did not reach optimality. We therefore cannot draw conclusions from a collection of suboptimal solutions because they may not necessarily be comparable.

2 Sensitivity Analysis with Two and Three Lanes

The concept of lanes introduces one more dimension to a configuration’s attractiveness: *availability*, that is, how quickly a customer can obtain it. As explained in the main body of the paper, availability can be treated as a feature whose options are the different lanes: fastest, second fastest, etc. As a feature, availability would have a relative importance and its options would have utilities. Finally, the company may choose to either charge a premium/markup for configurations in faster lanes or keep the prices the same. In the absence of these specific details, we developed a set of experiments to better understand what happens to customers’ migration lists and to several output measures related to the optimal product portfolio when different lanes are present. Ultimately, what matters to the optimization step are the configurations on each migration list, as well as the order in which they appear. Therefore, we can simulate the presence of the availability feature by enumerating the different types of lists it would likely produce, without the need to artificially invent values for optimization-related parameters we currently do not have.

Note that our approach is valid regardless of whether faster lanes have markups. If the markup amount is known to the company, it can be used to determine whether or not a given configuration is dropped from a customer’s migration list by looking at the customer’s reservation price while keeping in mind that, in the presence of lanes, it might be possible that customers’ reservation prices increase to accommodate the markups if their utility of availability is high.

2.1 Customer and List Types as a Function of Availability

We think of customers in two dimensions: their willingness to pay for a faster lane (low, medium, or high), and the amount of utility they place in the availability feature (low, medium, or high). This leads to the two 3-by-3 matrices of customer types as shown in Tables 2(a) and 2(b), one for the two-lane case and the other for the three-lane case.

We explain how the three-lane case and Table 2(b) work; the two-lane case and Table 2(a) are analogous. For simplicity, we denote the different configurations by numbers and the different lanes by letters, with “A” being the fastest lane, “B” the next fastest, etc. Hence, 1A and 1B represent configuration 1 offered/bought in the first and second lanes, respectively. Recall from the optimization model that only one of these is possible because one of the decisions being made is choosing a single lane in which to offer each configuration. We refer to a customer type by a pair (w, u) , where w is their willingness to pay, and u is how much utility they place on availability (both ranging from low, to medium, to high). Consider a customer with three configurations on his original migration list, sorted in decreasing order according to their respective utilities, while ignoring availability. Let this list be 1, 2, 3. If this were a (low, low) customer (top left cell of Table 2(b)), his list would only include lane C, that is, 1C, 2C, 3C, because he neither cares about

		Utility of Availability		
		Low	Med	High
Willingness to Pay	Low	B	B	B
	Med	50% B 50% BA	50% B 25% BA 25% AB	50% B 50% AB
	High	BA	50% BA 50% AB	A

(a) Two-lane case.

		Utility of Availability		
		Low	Med	High
Willingness to Pay	Low	C	CB	B
	Med	CB	BC	50% B 50% AB
	High	CBA	50% BA 50% ABC	25% AB 75% A

(b) Three-lane case.

Table 2: Migration list types according to willingness to pay and utility of availability.

availability nor is willing to pay for it. If this were a (high, low) customer, his list would include all lanes (because he can afford them), but the slower lanes would come first (because availability is not important enough to make it worth spending more for the faster lanes). Therefore, his list would be: 1C, 2C, 3C, 1B, 2B, 3B, 1A, 2A, 3A. Similar reasoning can be applied to the remaining seven types of customers in each cell of Table 2(b). More than one type of list appears in a few of the cells because we wish to model some uncertainty or heterogeneity; the table indicates what percentage of those customers would have each of the lists shown. For example, if the sample customer above were (high, med) there is a 50% chance his migration list would be 1B, 2B, 3B, 1A, 2A, 3A, and a 50% chance it would be 1A, 2A, 3A, 1B, 2B, 3B, 1C, 2C, 3C.

2.2 Simulating a Sweep from Low to High Utility of Availability

Once again, we focus the explanation on the three-lane case (the two-lane case is analogous). To investigate the effect of availability, we explore what happens to migration lists as we execute a series of runs that give progressively more value to availability; we call this a *sweep*. By finding the optimal product portfolio characteristics and lane allocations for each of these runs and plotting the results, we try to understand how much effect the utility of availability has on the different outputs produced by our model.

Think of Table 2(b) as a three-step sweep that goes from low utility of availability, when only the customer types shown in its left column exist, to a medium-utility situation when only the customer types in its middle column exist, to a high-utility situation when only the customers types in its right column exist. We transform that into a ten-step sweep by adding seven more columns to that table: three columns between “Low” and “Med”, and four between “Med” and “High”. The resulting steps 1 through 5 and 6 through 10 are depicted, respectively, in Tables 3(a) and 3(b). Each column in those tables shows the proportion of customers of each type that exist in each step

		Utility of Availability				
		1	2	3	4	5
Willingness to Pay	Low	$\ell\%$ C	.75 $\ell\%$ C .25 $\ell\%$ CB	.5 $\ell\%$ C .5 $\ell\%$ CB	.25 $\ell\%$ C .75 $\ell\%$ CB	$\ell\%$ CB
	Med	$m\%$ CB	.75 $m\%$ CB .25 $m\%$ BC	.5 $m\%$ CB .5 $m\%$ BC	.25 $m\%$ CB .75 $m\%$ BC	$m\%$ BC
	High	$h\%$ CBA	.75 $h\%$ CBA .125 $h\%$ BA .125 $h\%$ ABC	.5 $h\%$ CBA .25 $h\%$ BA .25 $h\%$ ABC	.25 $h\%$ CBA .375 $h\%$ BA .375 $h\%$ ABC	.5 $h\%$ BA .5 $h\%$ ABC

(a) Steps 1 through 5 in the sweep.

		Utility of Availability				
		6	7	8	9	10
Willingness to Pay	Low	.8 $\ell\%$ CB .2 $\ell\%$ B	.6 $\ell\%$ CB .4 $\ell\%$ B	.4 $\ell\%$ CB .6 $\ell\%$ B	.2 $\ell\%$ CB .8 $\ell\%$ B	$\ell\%$ B
	Med	.8 $m\%$ BC .1 $m\%$ B .1 $m\%$ AB	.6 $m\%$ BC .2 $m\%$ B .2 $m\%$ AB	.4 $m\%$ BC .3 $m\%$ B .3 $m\%$ AB	.2 $m\%$ BC .4 $m\%$ B .4 $m\%$ AB	.5 $m\%$ B .5 $m\%$ AB
	High	.4 $h\%$ BA .4 $h\%$ ABC .05 $h\%$ AB .15 $h\%$ A	.3 $h\%$ BA .3 $h\%$ ABC .1 $h\%$ AB .3 $h\%$ A	.2 $h\%$ BA .2 $h\%$ ABC .15 $h\%$ AB .45 $h\%$ A	.1 $h\%$ BA .1 $h\%$ ABC .2 $h\%$ AB .6 $h\%$ A	.25 $h\%$ AB .75 $h\%$ A

(b) Steps 6 through 10 in the sweep.

Table 3: Ten-step sweep from low to high utility of availability.

of the sweep. In addition, to add a bit more variety to the sweep, in each step $\ell\%$ of the customers have low willingness to pay, $m\%$ have medium willingness, and $h\%$ have high willingness. In our experiments, we perform the ten-step sweeps under three scenarios having $\ell = m = h$: the 100% scenario ($\ell + m + h = 100$), the 75% scenario ($\ell + m + h = 75$), and the 50% scenario ($\ell + m + h = 50$). In scenarios where $\ell + m + h < 100$, the remaining customers are assigned uniformly at random to types of lists not yet represented in the step in question. For example, if we are in step 7 of the 75% scenario, column 7 of Table 3(b) covers list types A, B, AB, BC, BA, CB, and ABC. Hence, the remaining 25% of the customers get equally distributed between lists of type C and CBA.

While the specific percentages and orderings within and between cells of our tables, and the mechanism for *sweeping* across them are admittedly subjective, we maintain that our overall procedure captures the general characteristics of a customer population that grows increasingly more concerned about availability.

2.3 Experiments with and without Price Markups

We conduct experiments with and without price markups on faster lanes. There are plausible reasons for a company to choose either strategy, including but not limited to:

No Markups: Lanes are solely used as a tool to incentivize customers to buy a certain subset of the products, so price lists remain the same.

Markups: It is more costly to make configurations available on faster lanes, so a markup is nec-

essary.

With two lanes, we consider four cases: either no markups, or markups of 2%, 5%, or 10% for lane A only (lane B is never marked up). With three lanes, we consider three cases: either no markups, markups of 2% and 5% for lanes B and A, respectively, or markups of 5% and 10% for lanes B and A, respectively (lane C is never marked up).

From an optimization point of view, the optimal portfolio seeks to maximize the overall profit, which is the difference between revenue and costs. When faster lanes are marked up, one would expect higher profits to be achieved by having more of the sales come from more expensive lanes, at the expense of a slightly higher complexity cost (i.e. a larger portfolio). When faster lanes are not marked up, one would expect the optimal portfolio to focus more on complexity cost reduction in order to increase profits. Both of these behaviors have been confirmed by our experiments.

In the ensuing subsections, we present a collection of plots presenting the results of our sensitivity experiments related to the number of lanes included in the portfolio for different customer-proportion and markup scenarios and summarize the main results here:

In Section 2.4, we demonstrate that in the 100% scenario, an increase in availability value allows us to steer the customers away from their first choice. As availability grows larger, customers become more dispersed and the portfolio offered has more options. As we dilute the customer base and move towards the 50% scenario, these effects become less pronounced.

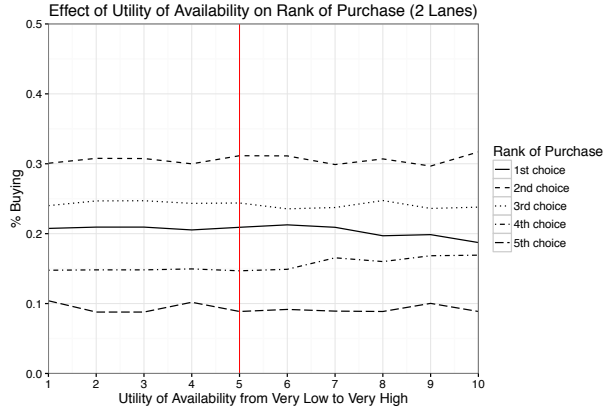
In Section 2.5, we show that as the availability value increases, customers are buying more from Lane 1 (the fastest lane) and hence, the size of Lane 1 increases, while the size of lane 2 stays relatively constant.

We observe similar results in Sections 2.6 and 2.7 in the case with 3-lanes, the plots in Section 2.7 demonstrate that at high enough values for availability, Lane 3 gets practically pushed out, suggesting that the company can switch to the 2-lane portfolio.

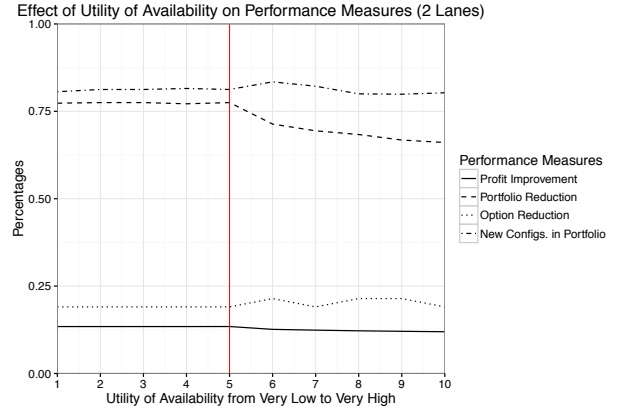
In Sections 2.8 and 2.9, we compare the 2-lane portfolio to 3-lane portfolio based on the profit and cost of complexity reduction. In general, we observe that 2-lane portfolio brings higher value than the 3-lanes, due to greater reductions in complexity.

In Sections 2.10 through 2.22, we repeat the same experiments for the case with price markups and observe similar insights; not surprisingly though, now profit tends to increase with utility of availability and portfolios grow larger at higher utility, to chase these profits. Relatedly, the plots in Section 2.22 demonstrate that when the markup is high enough, the 3-lane portfolio always dominates 2-lane portfolios regardless of the availability value.

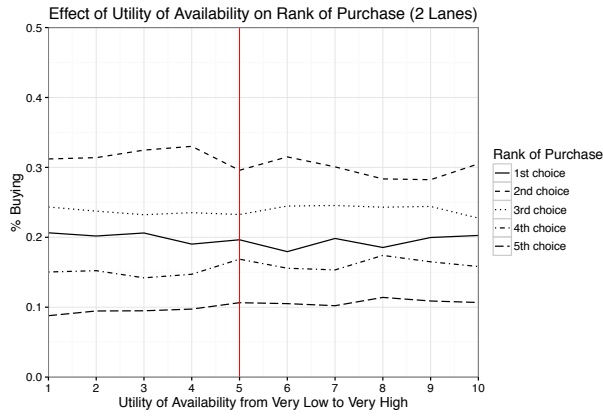
2.4 Performance Measures with Two Lanes and No Markup: Comparison across Scenarios



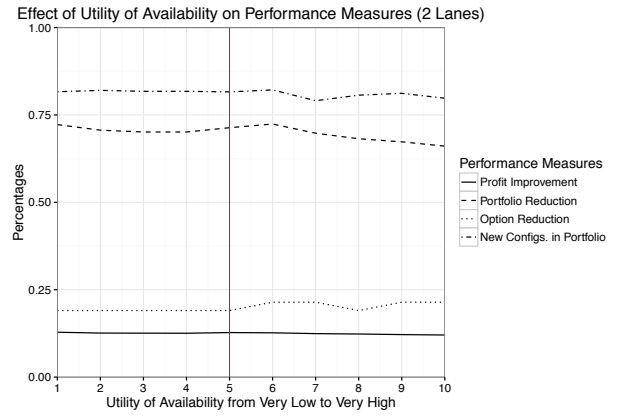
(a) Availability vs Purchase Rank, 100% Scenario



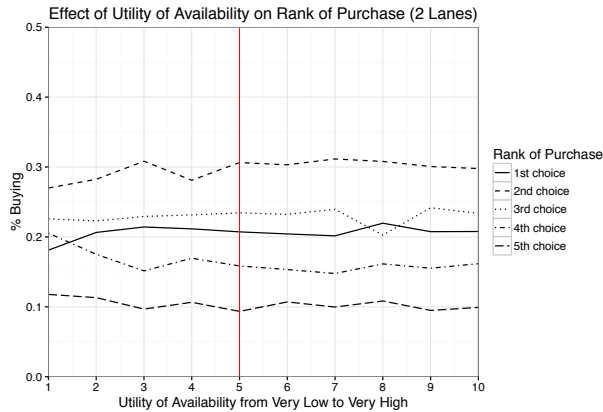
(b) Availability vs Other Outputs, 100% Scenario



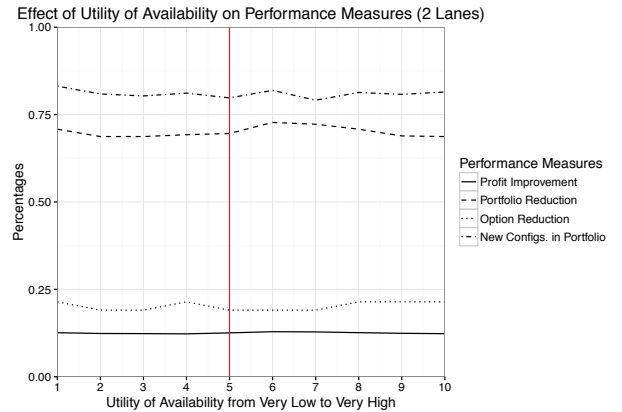
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

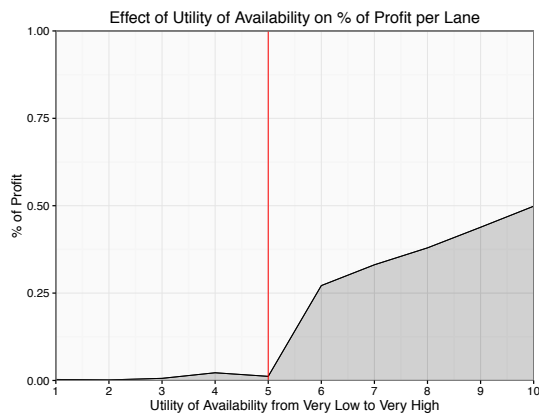


(e) Availability vs Purchase Rank, 50% Scenario

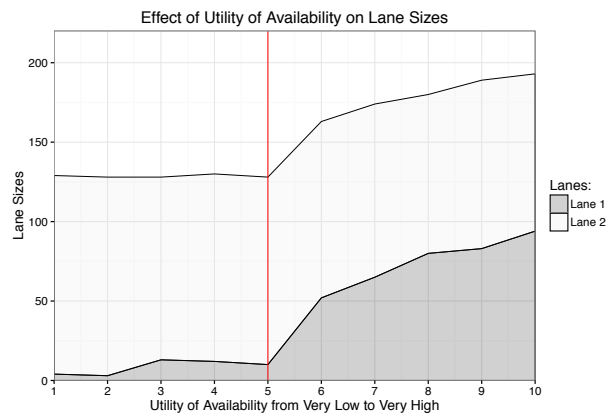


(f) Availability vs Other Outputs, 50% Scenario

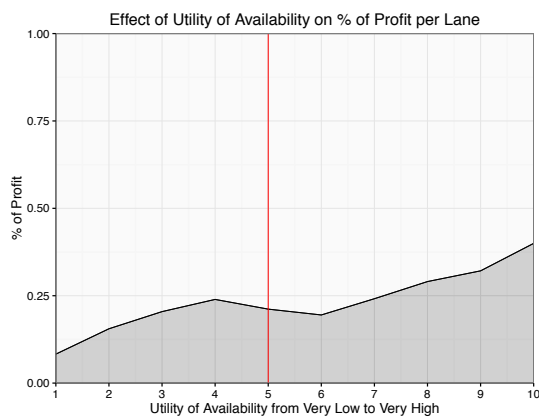
2.5 Relative Lane Performances with Two Lanes and No Markup: Comparison across Scenarios



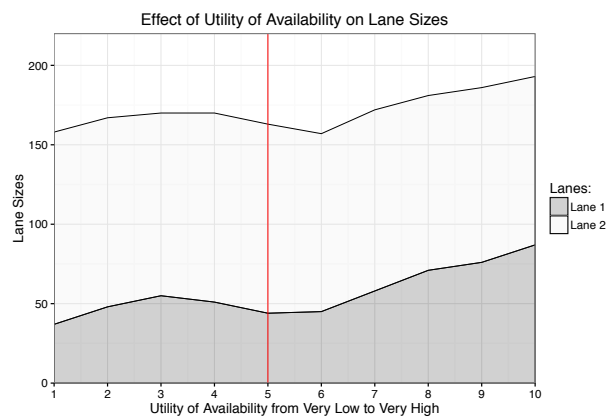
(a) Availability vs Profit per Lane, 100% Scenario



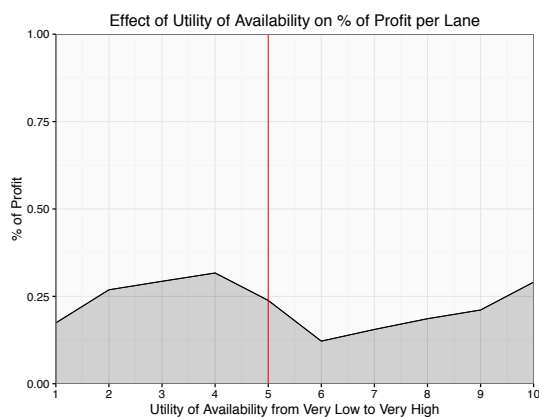
(b) Availability vs Lane Size, 100% Scenario



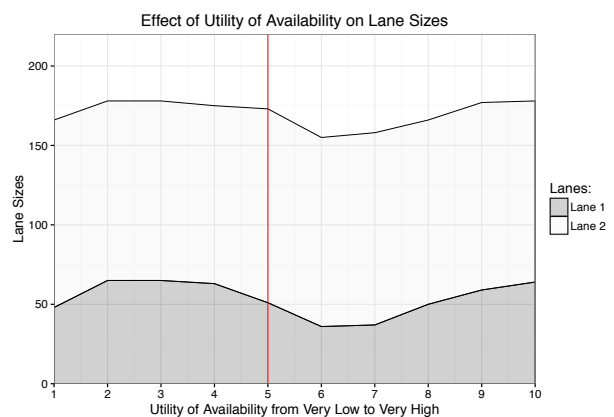
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

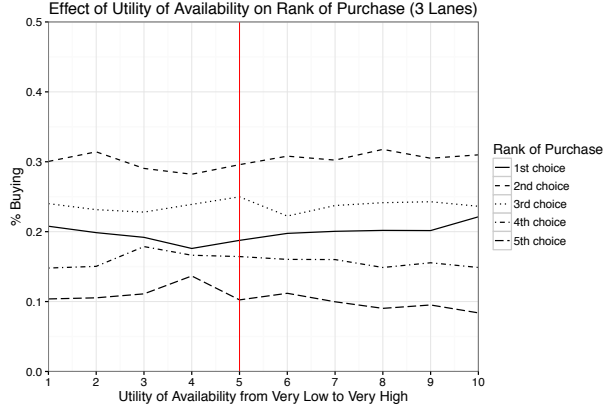


(e) Availability vs Profit per Lane, 50% Scenario

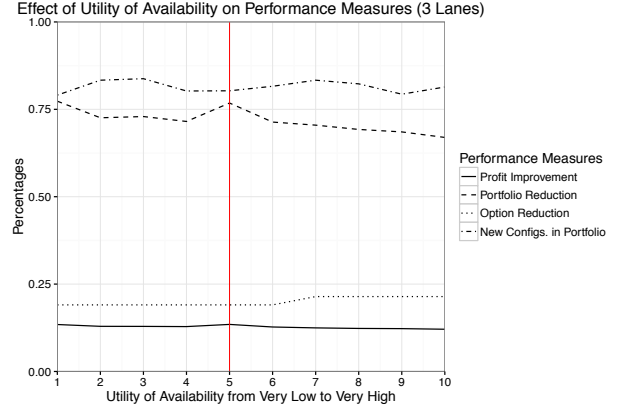


(f) Availability vs Lane Size, 50% Scenario

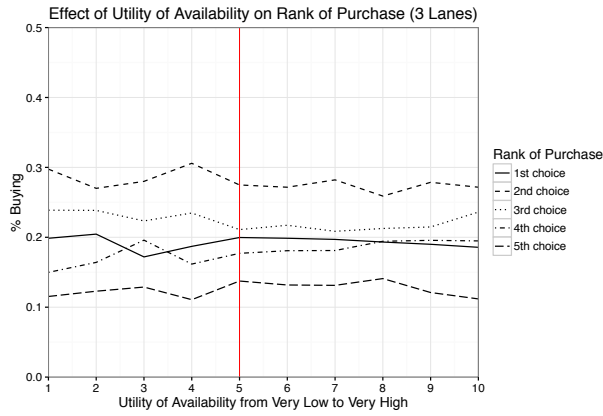
2.6 Performance Measures with Three Lanes and No Markup: Comparison across Scenarios



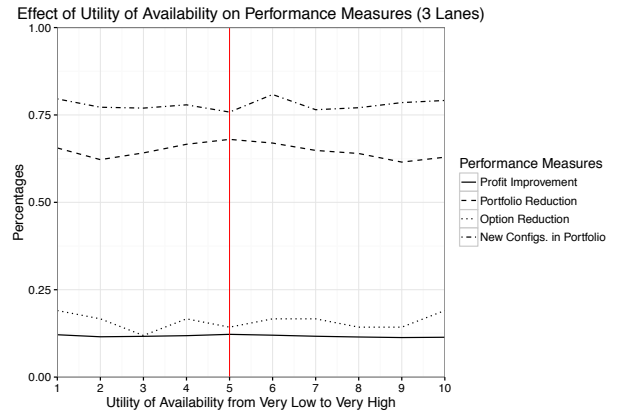
(a) Availability vs Purchase Rank, 100% Scenario



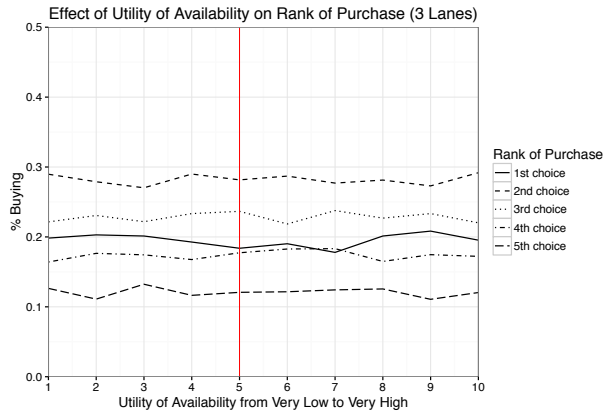
(b) Availability vs Other Outputs, 100% Scenario



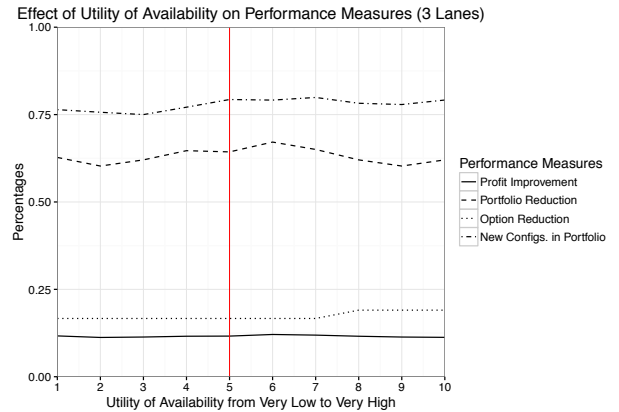
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

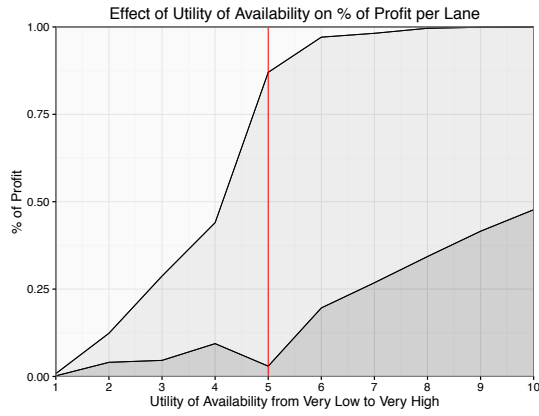


(e) Availability vs Purchase Rank, 50% Scenario

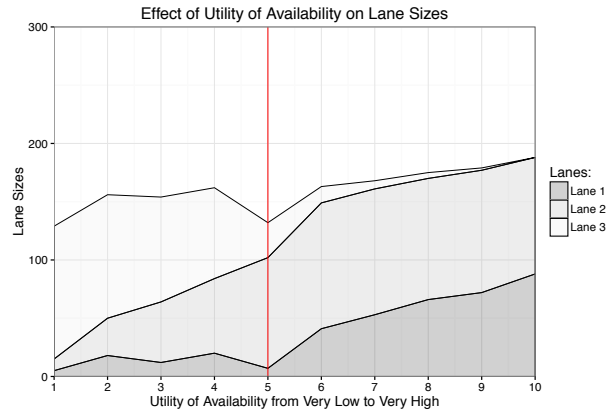


(f) Availability vs Other Outputs, 50% Scenario

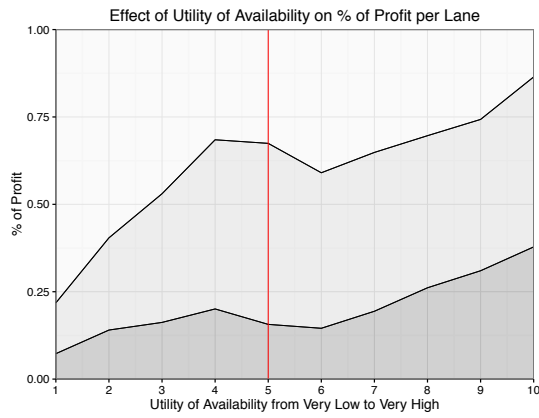
2.7 Relative Lane Performances with Three Lanes and No Markup: Comparison across Scenarios



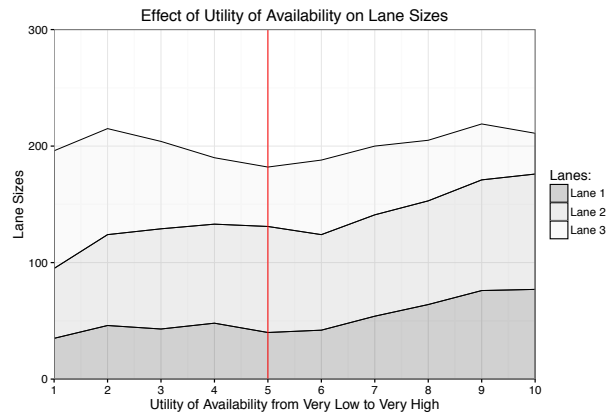
(a) Availability vs Profit per Lane, 100% Scenario



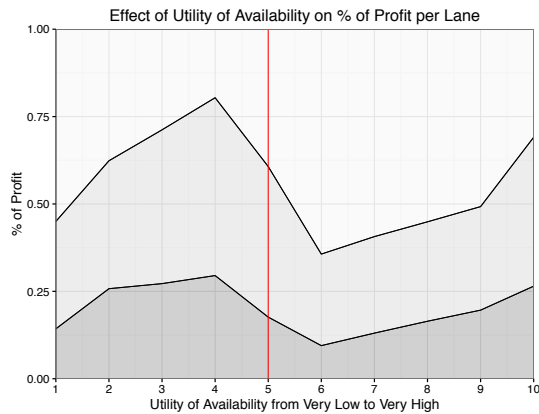
(b) Availability vs Lane Size, 100% Scenario



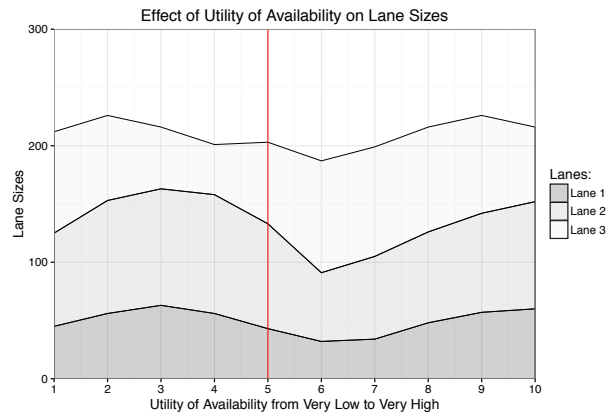
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

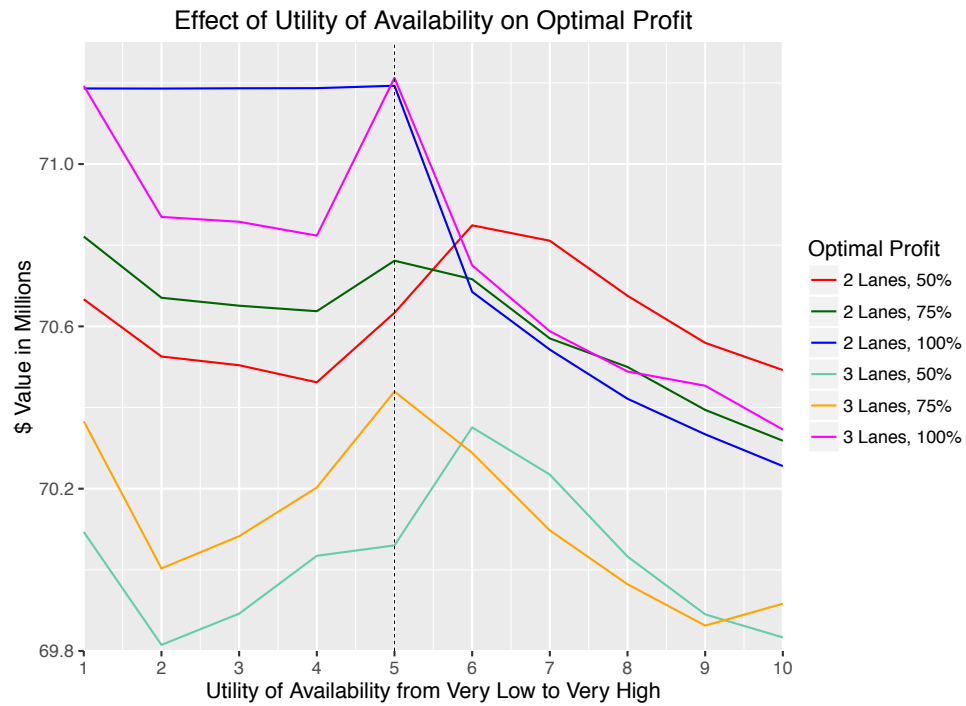


(e) Availability vs Profit per Lane, 50% Scenario

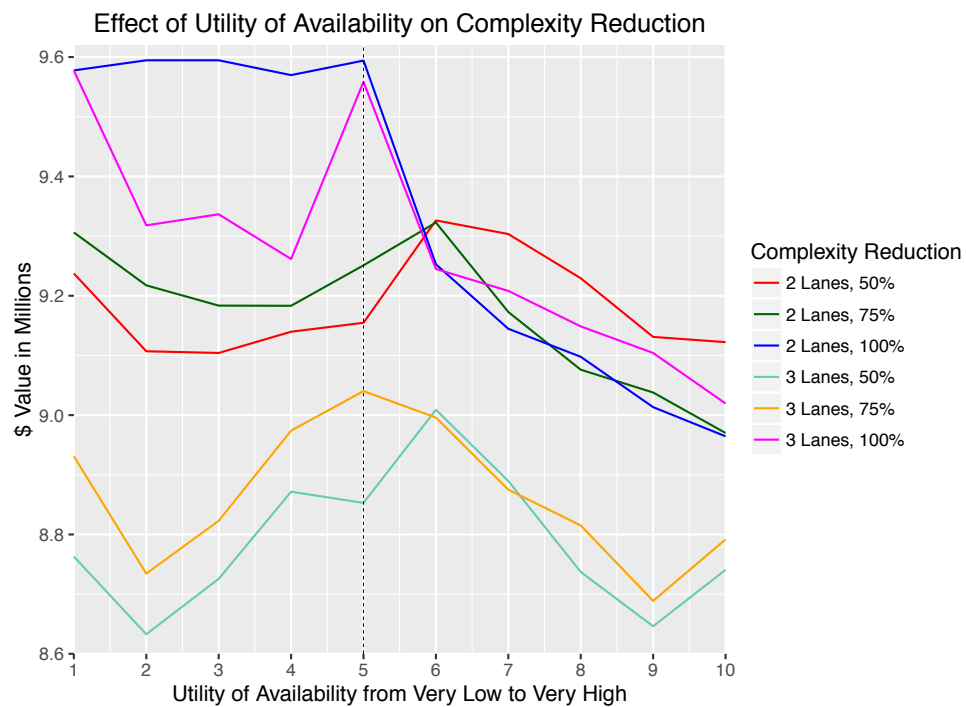


(f) Availability vs Lane Size, 50% Scenario

2.8 Comparing Two and Three Lanes across Different Scenarios on Profit and Complexity Reduction without Markups

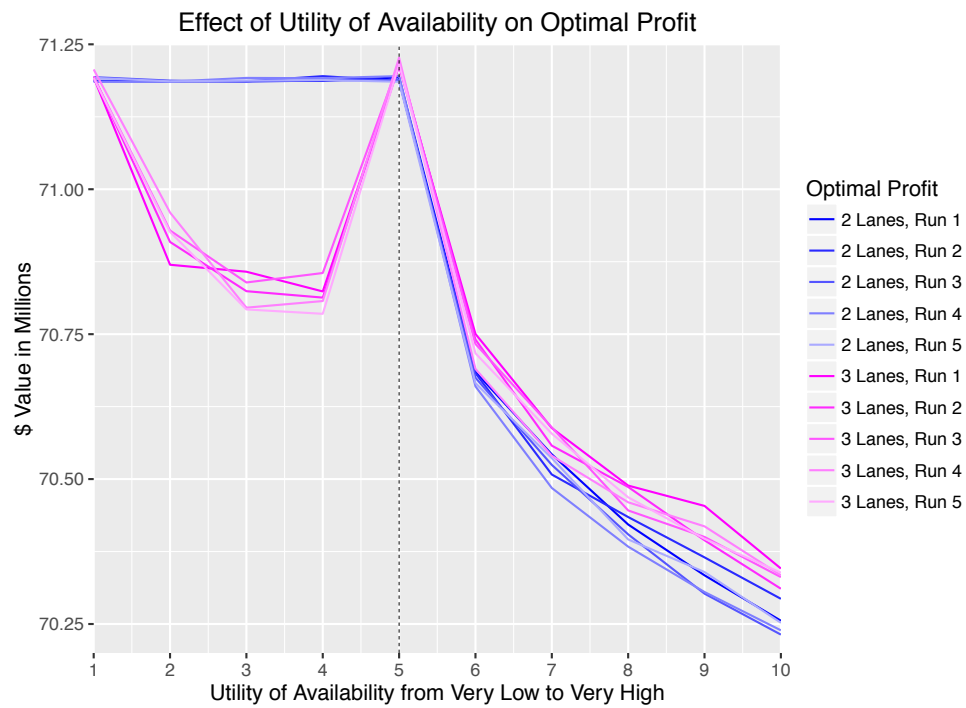


(a) Utility of Availability vs Optimal Profit

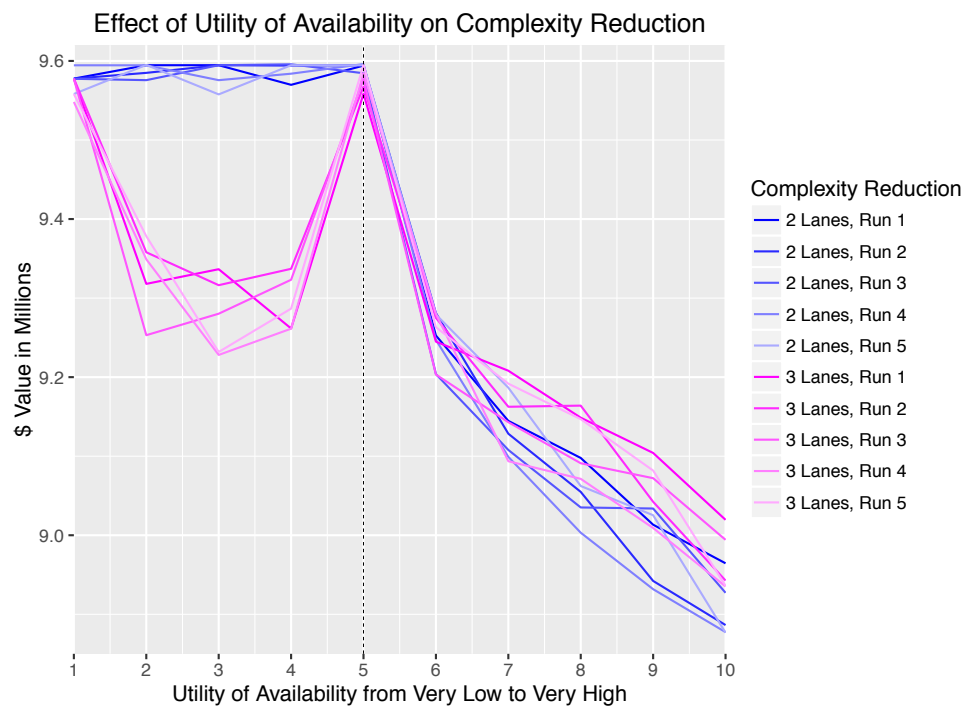


(b) Utility of Availability vs Complexity Reduction

2.9 Replications of the 100% Scenario for Two and Three Lanes to Compare Profit and Complexity Reduction without Markups

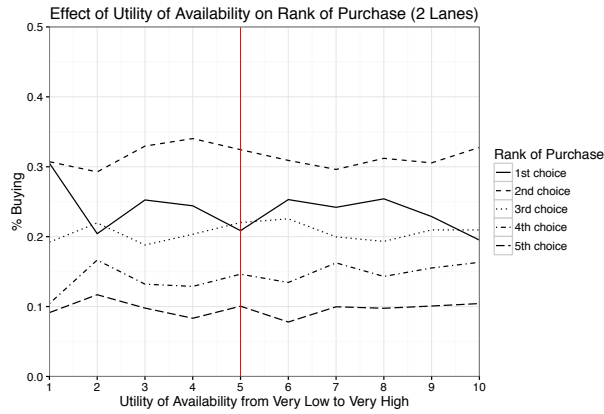


(a) Utility of Availability vs Optimal Profit

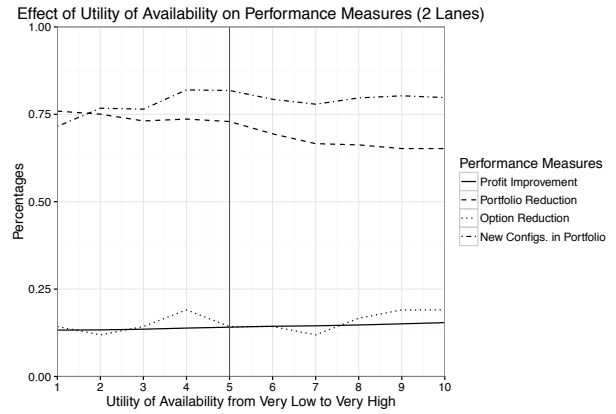


(b) Utility of Availability vs Complexity Reduction

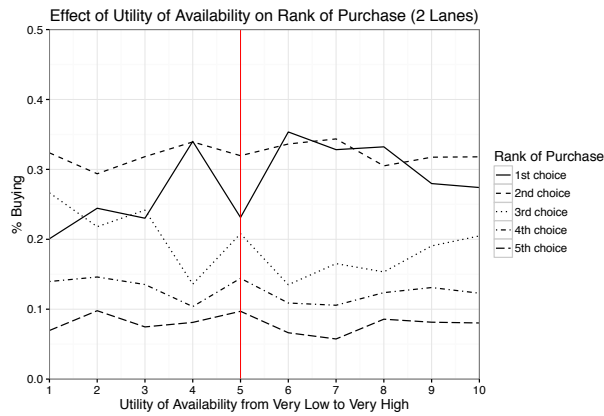
2.10 Performance Measures with Two Lanes and 2% Markup: Comparison across Scenarios



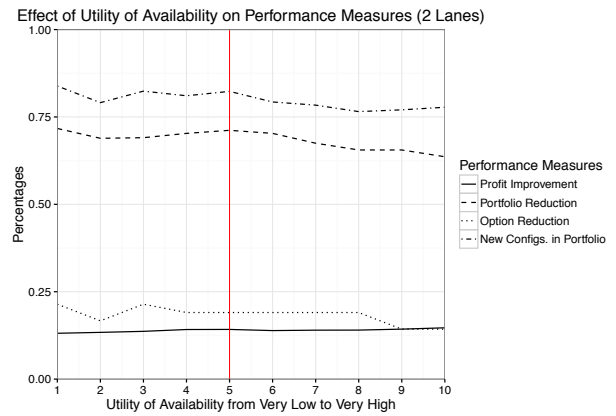
(a) Availability vs Purchase Rank, 100% Scenario



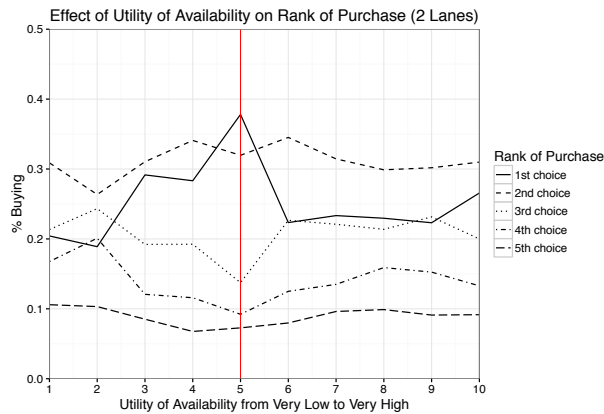
(b) Availability vs Other Outputs, 100% Scenario



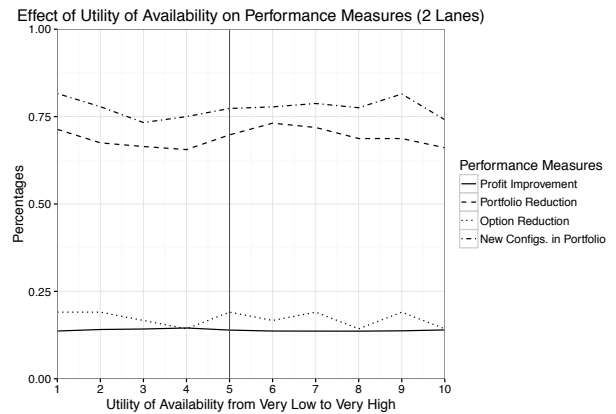
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

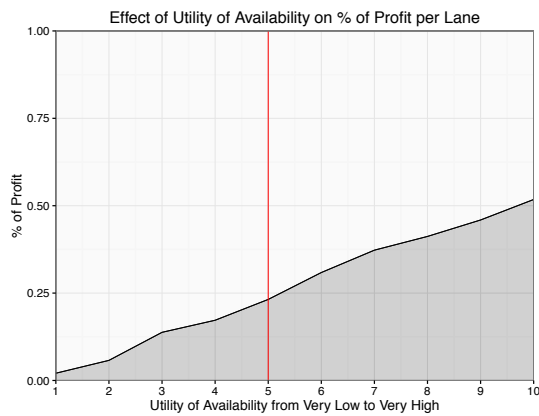


(e) Availability vs Purchase Rank, 50% Scenario

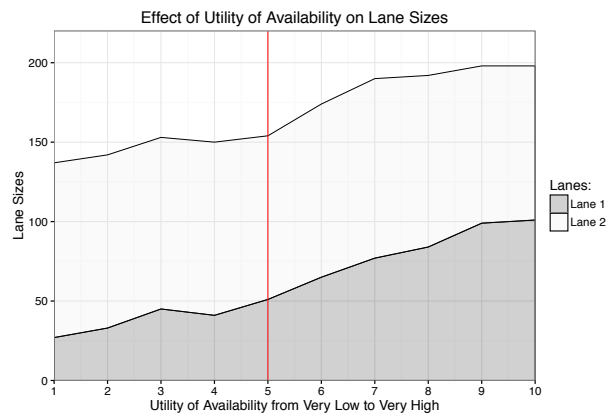


(f) Availability vs Other Outputs, 50% Scenario

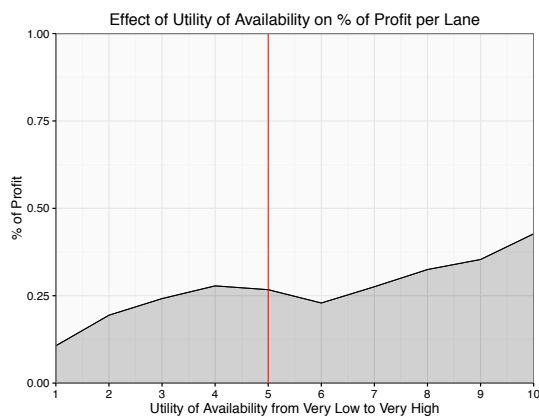
2.11 Relative Lane Performances with Two Lanes and 2% Markup: Comparison across Scenarios



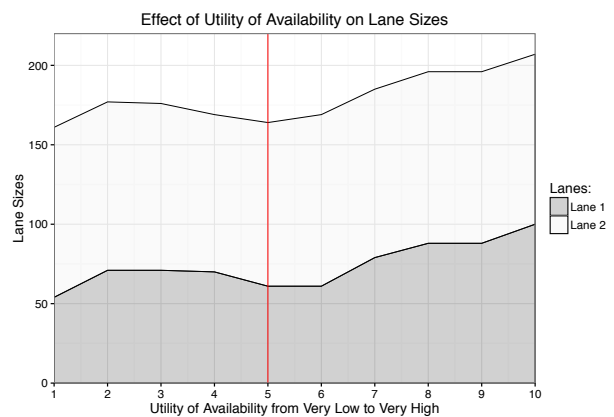
(a) Availability vs Profit per Lane, 100% Scenario



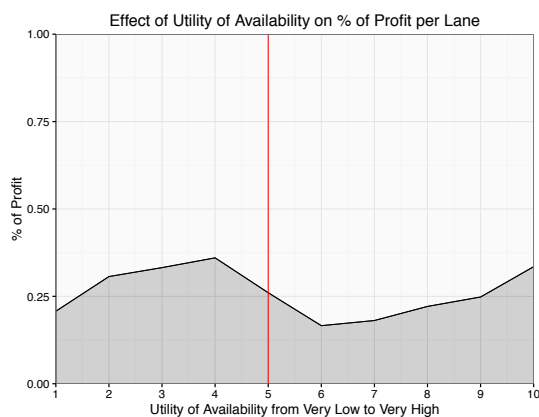
(b) Availability vs Lane Size, 100% Scenario



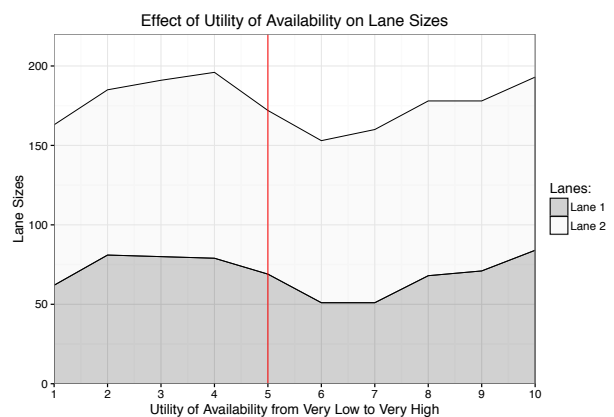
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

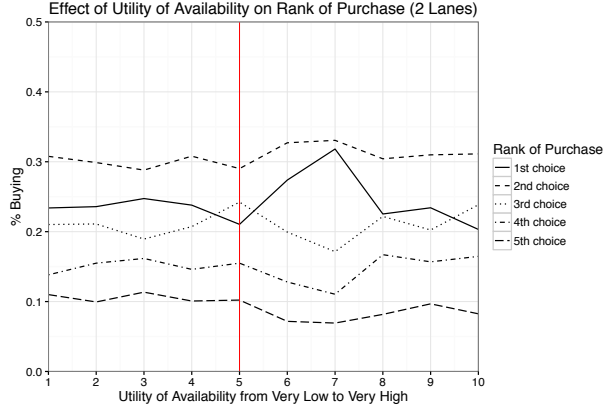


(e) Availability vs Profit per Lane, 50% Scenario

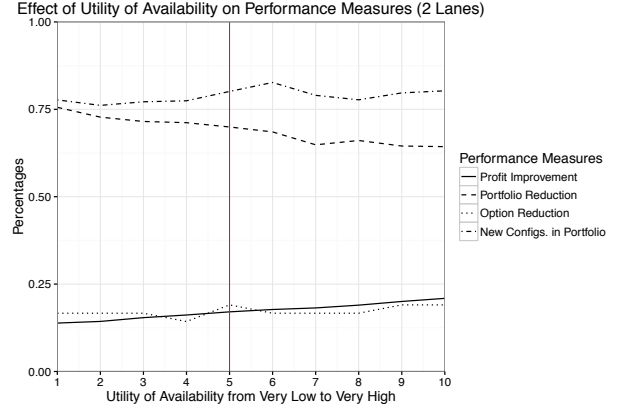


(f) Availability vs Lane Size, 50% Scenario

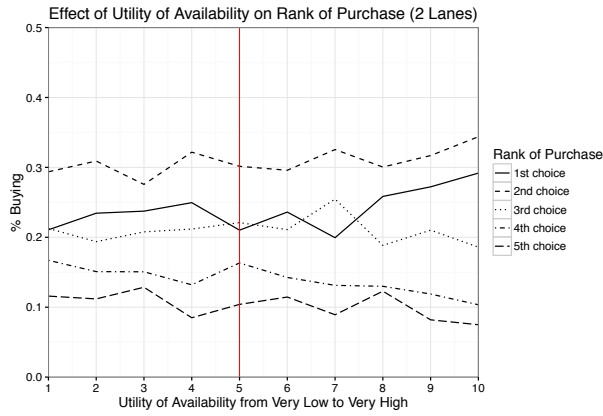
2.12 Performance Measures with Two Lanes and 5% Markup: Comparison across Scenarios



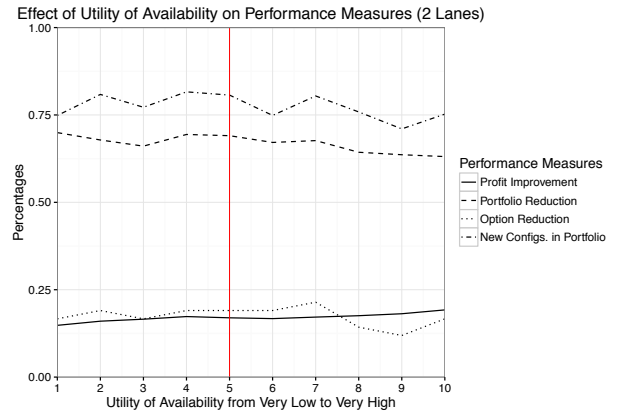
(a) Availability vs Purchase Rank, 100% Scenario



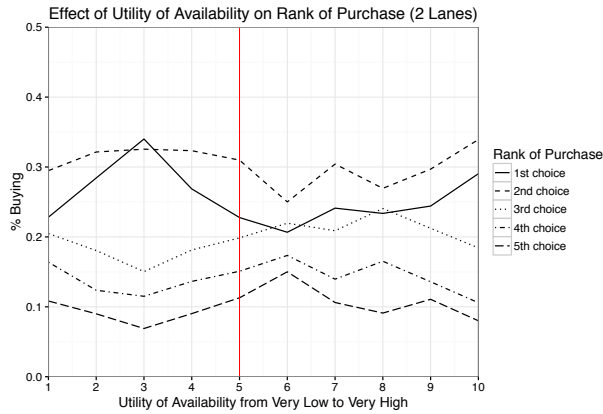
(b) Availability vs Other Outputs, 100% Scenario



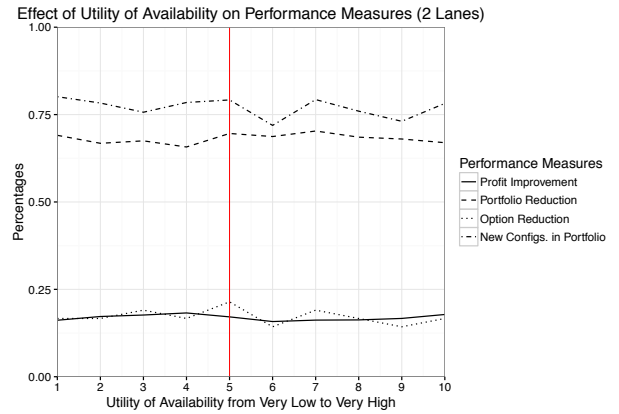
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

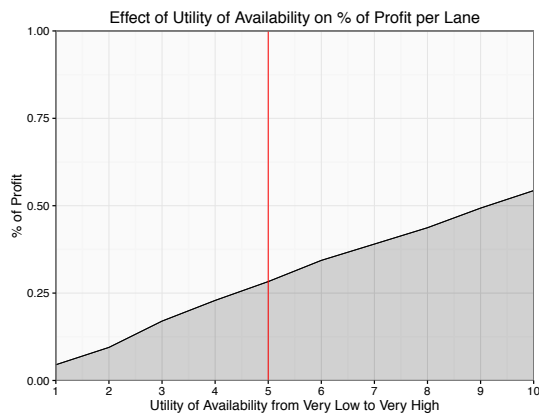


(e) Availability vs Purchase Rank, 50% Scenario

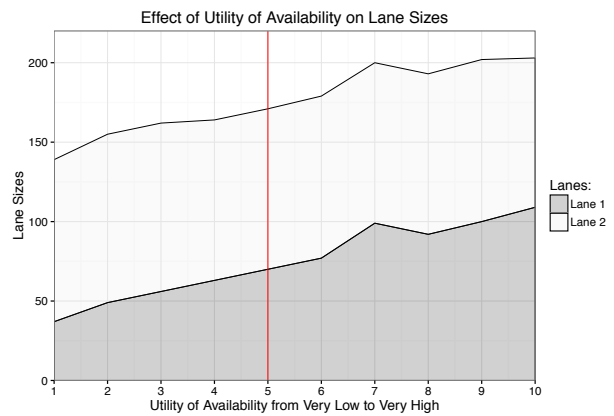


(f) Availability vs Other Outputs, 50% Scenario

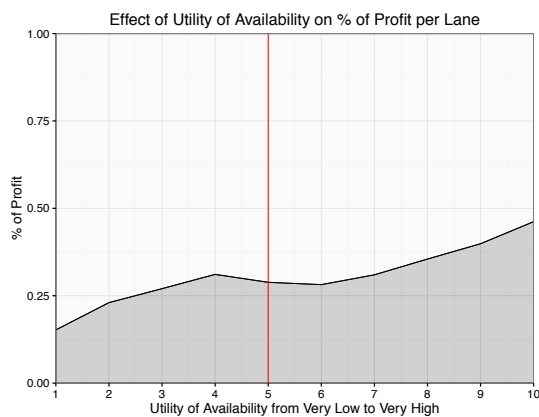
2.13 Relative Lane Performances with Two Lanes and 5% Markup: Comparison across Scenarios



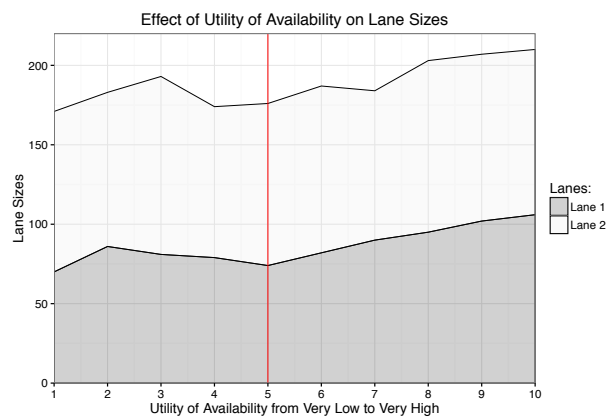
(a) Availability vs Profit per Lane, 100% Scenario



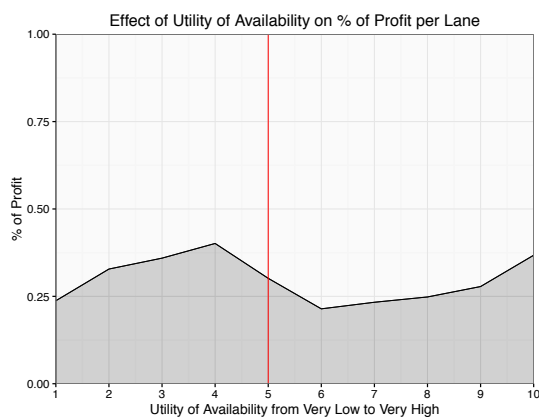
(b) Availability vs Lane Size, 100% Scenario



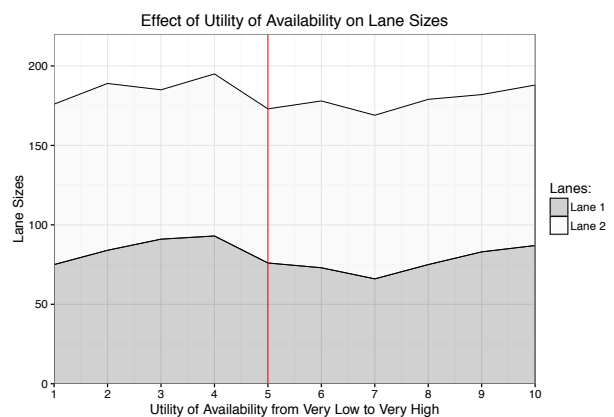
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

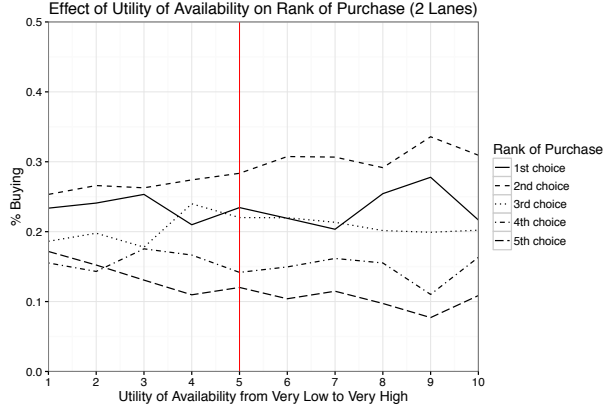


(e) Availability vs Profit per Lane, 50% Scenario

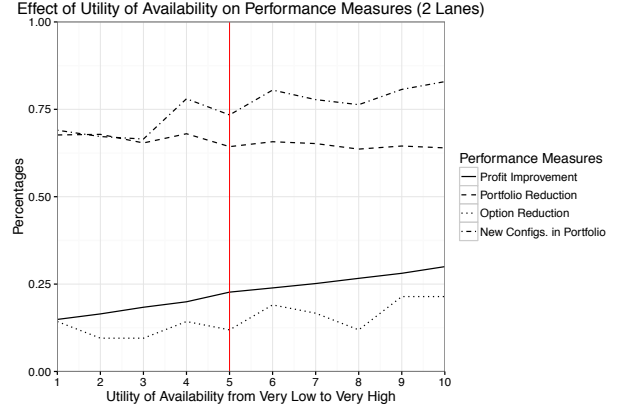


(f) Availability vs Lane Size, 50% Scenario

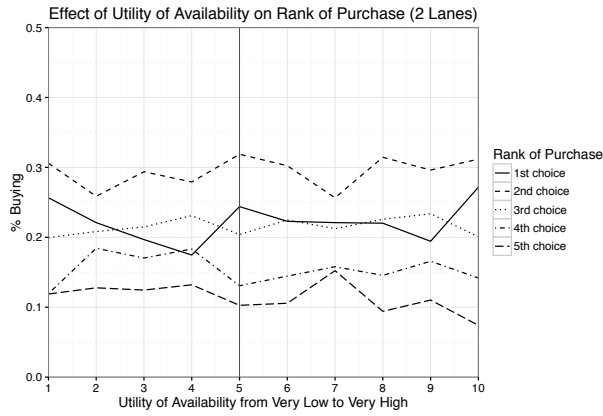
2.14 Performance Measures with Two Lanes and 10% Markup: Comparison across Scenarios



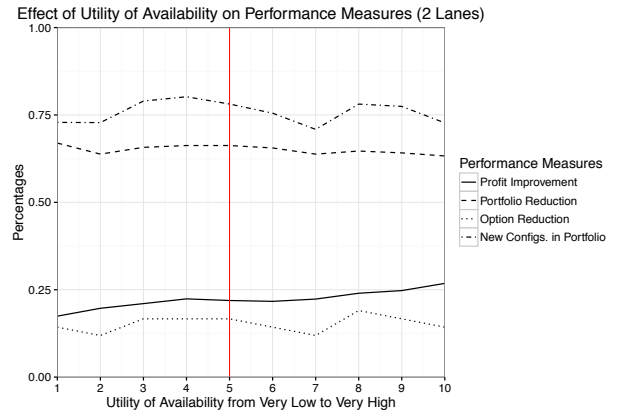
(a) Availability vs Purchase Rank, 100% Scenario



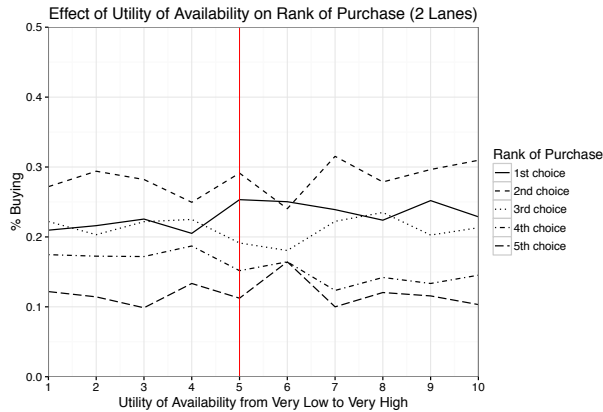
(b) Availability vs Other Outputs, 100% Scenario



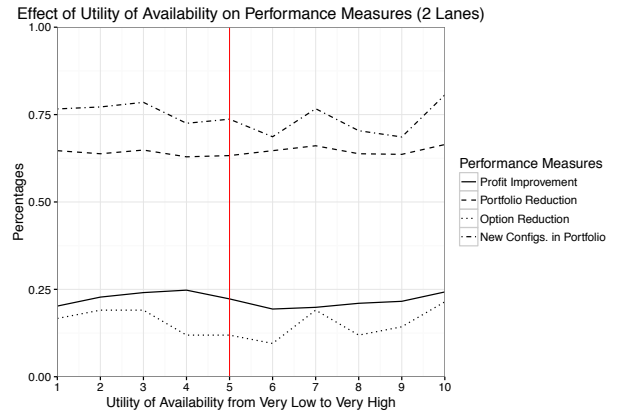
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

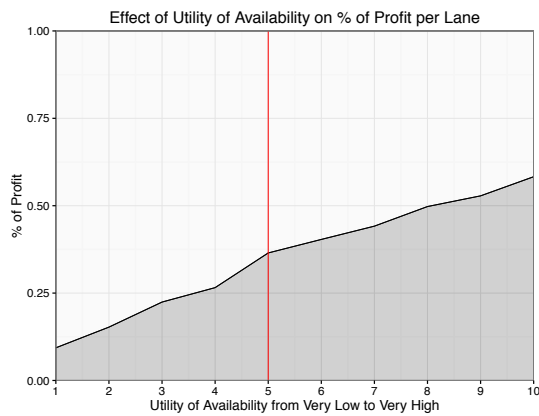


(e) Availability vs Purchase Rank, 50% Scenario

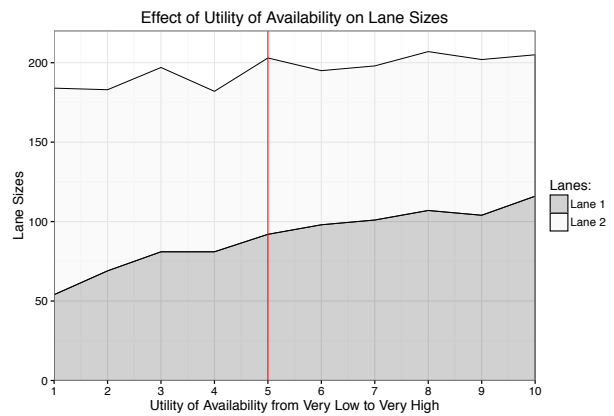


(f) Availability vs Other Outputs, 50% Scenario

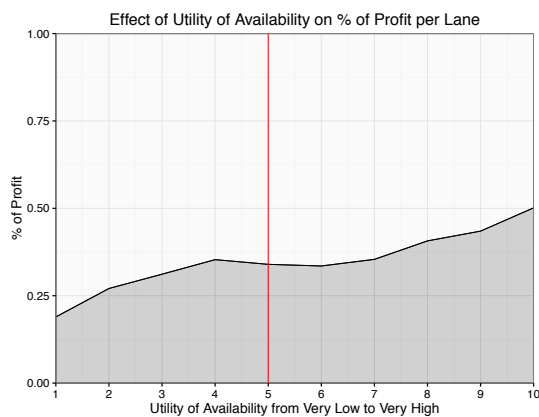
2.15 Relative Lane Performances with Two Lanes and 10% Markup: Comparison across Scenarios



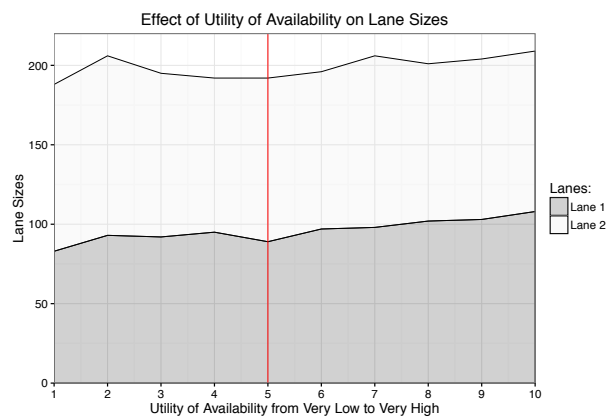
(a) Availability vs Profit per Lane, 100% Scenario



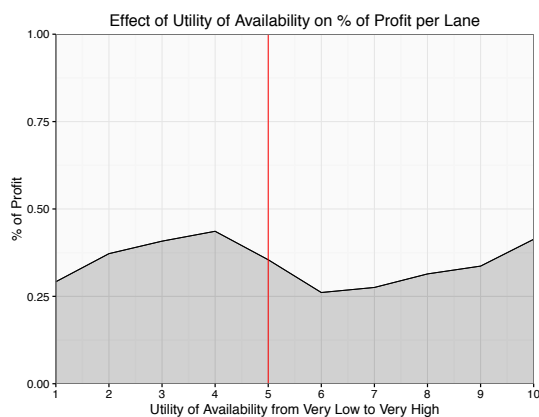
(b) Availability vs Lane Size, 100% Scenario



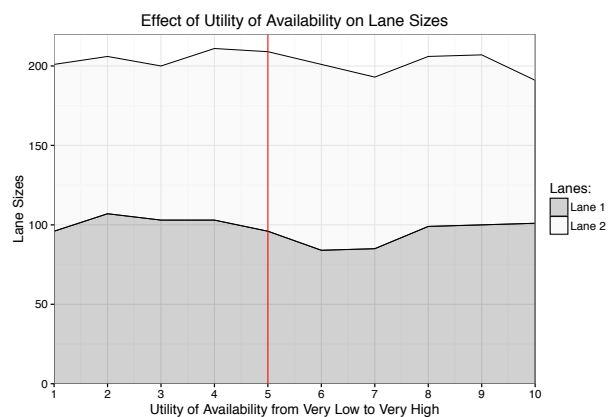
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

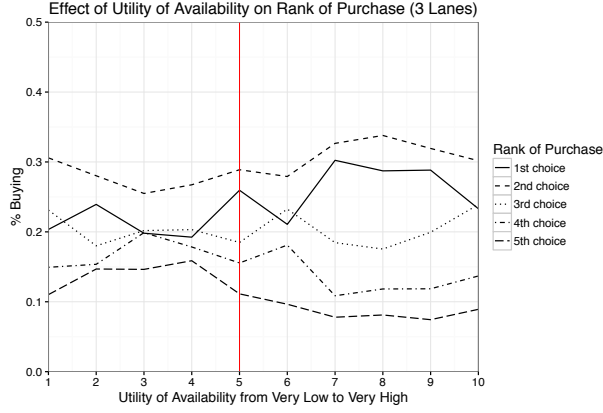


(e) Availability vs Profit per Lane, 50% Scenario

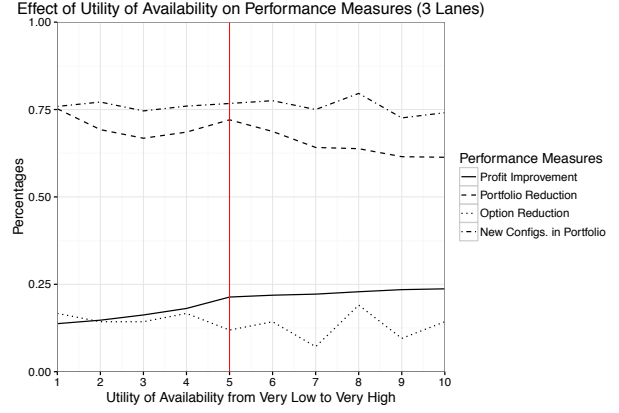


(f) Availability vs Lane Size, 50% Scenario

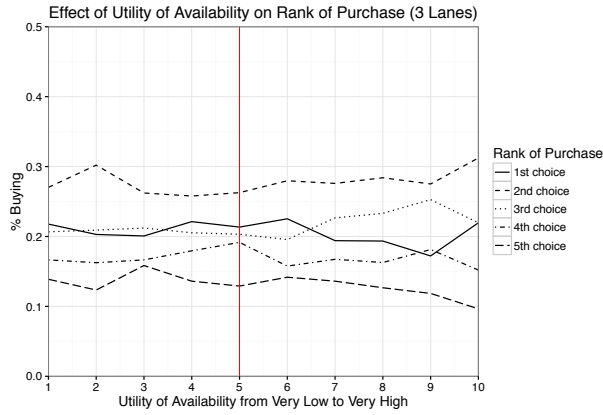
2.16 Performance Measures with Three Lanes and (2%, 5%) Markup: Comparison across Scenarios



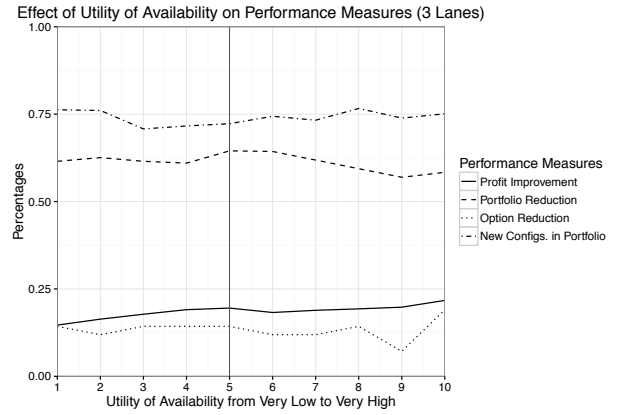
(a) Availability vs Purchase Rank, 100% Scenario



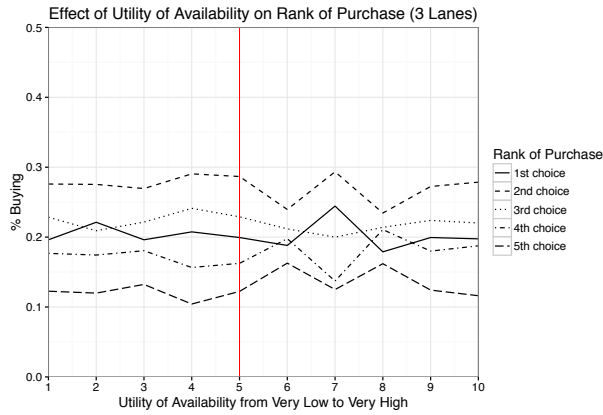
(b) Availability vs Other Outputs, 100% Scenario



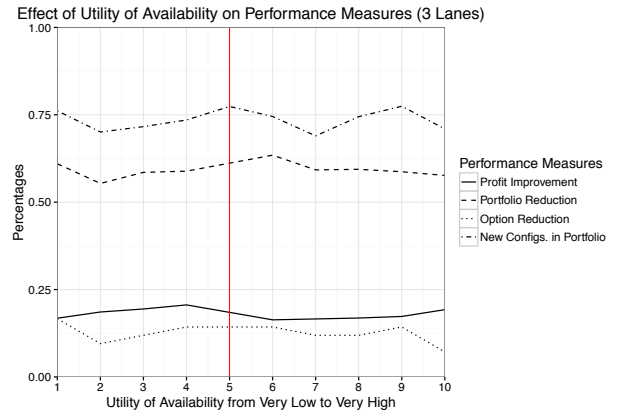
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

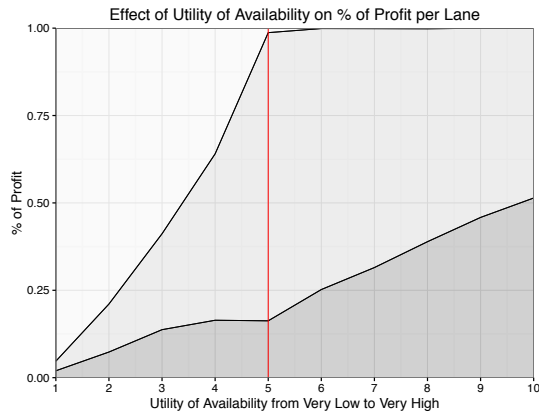


(e) Availability vs Purchase Rank, 50% Scenario

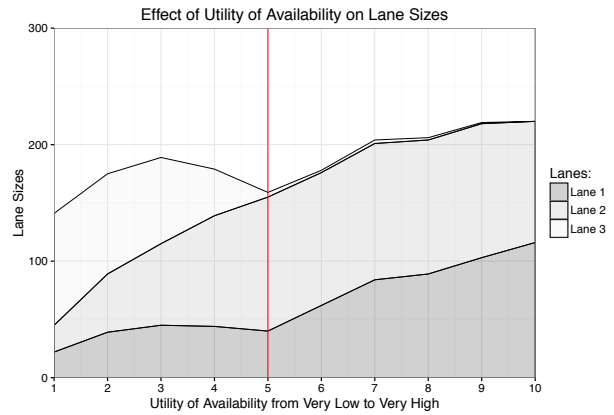


(f) Availability vs Other Outputs, 50% Scenario

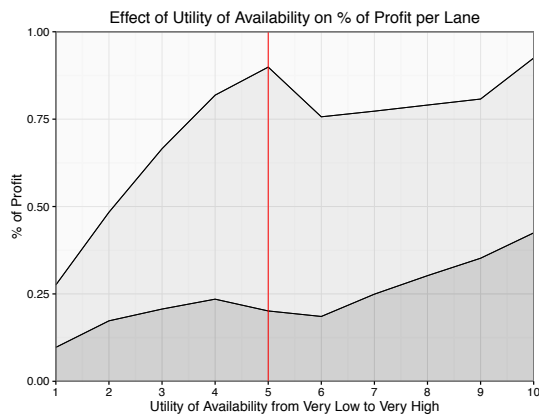
2.17 Relative Lane Performances with Three Lanes and (2%, 5%) Markup: Comparison across Scenarios



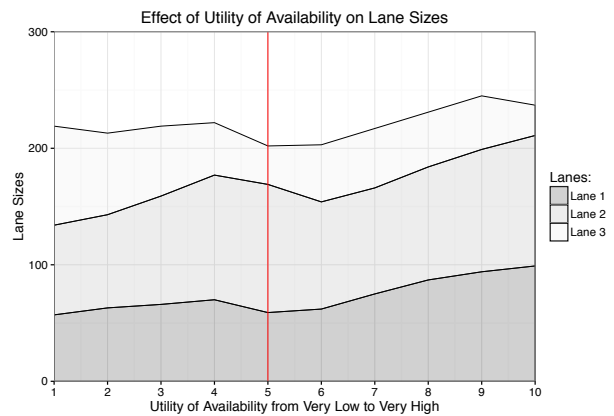
(a) Availability vs Profit per Lane, 100% Scenario



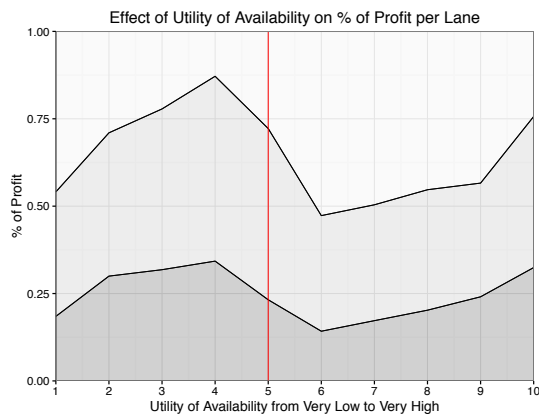
(b) Availability vs Lane Size, 100% Scenario



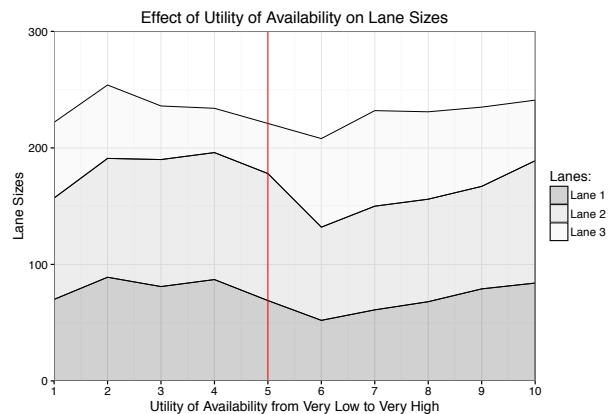
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

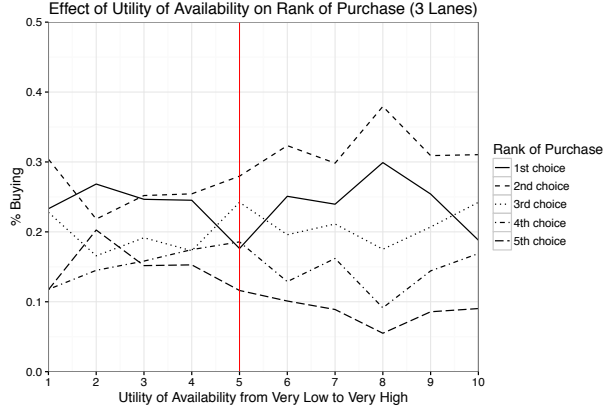


(e) Availability vs Profit per Lane, 50% Scenario

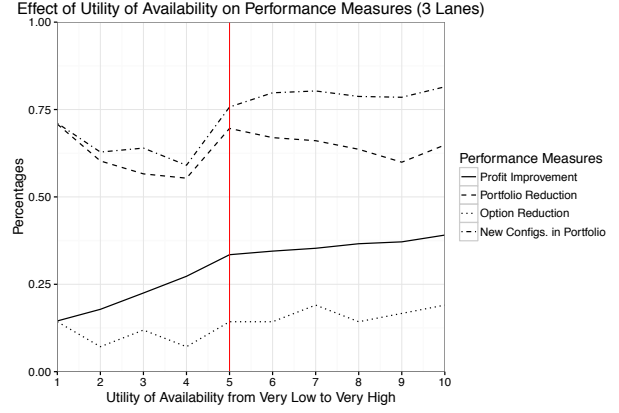


(f) Availability vs Lane Size, 50% Scenario

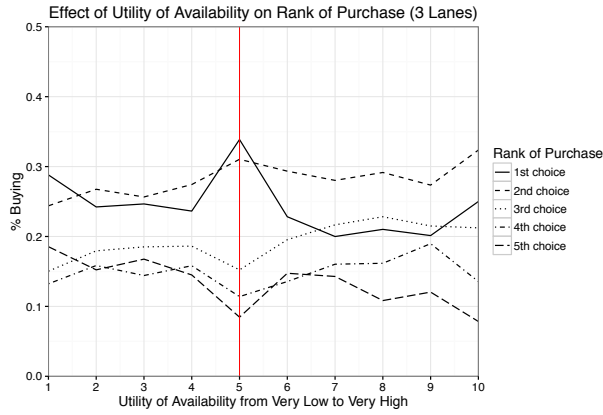
2.18 Performance Measures with Three Lanes and (5%, 10%) Markup: Comparison across Scenarios



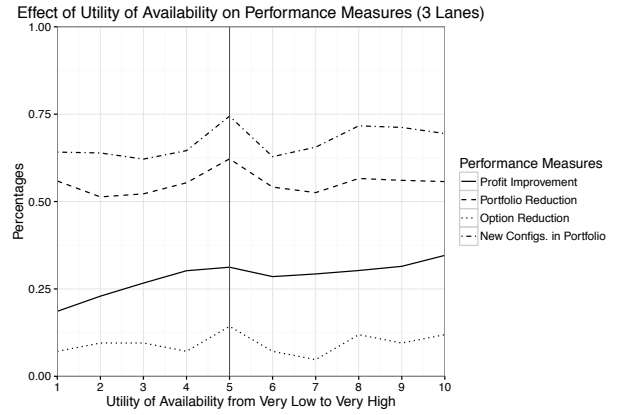
(a) Availability vs Purchase Rank, 100% Scenario



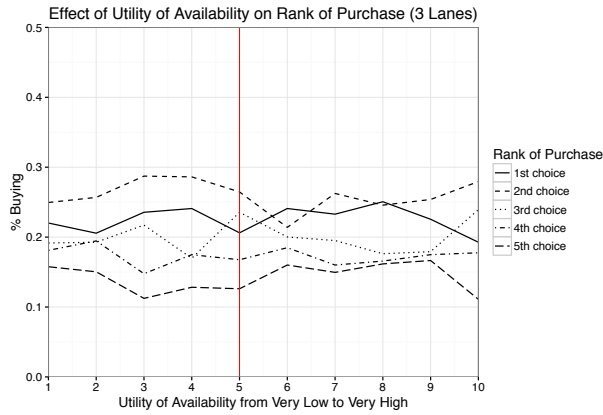
(b) Availability vs Other Outputs, 100% Scenario



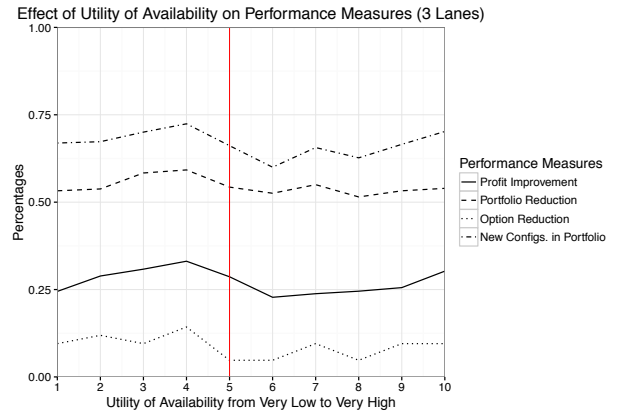
(c) Availability vs Purchase Rank, 75% Scenario



(d) Availability vs Other Outputs, 75% Scenario

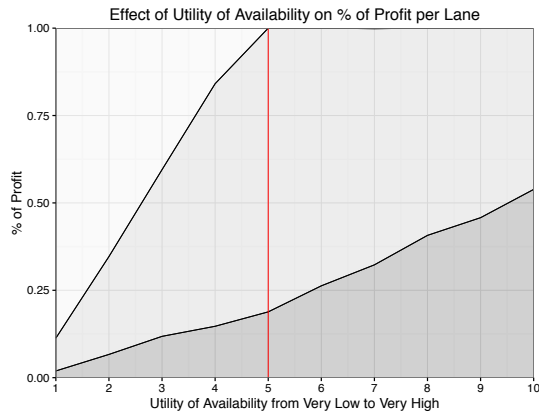


(e) Availability vs Purchase Rank, 50% Scenario

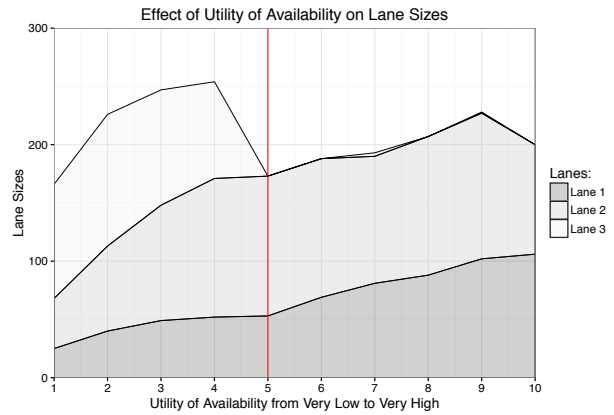


(f) Availability vs Other Outputs, 50% Scenario

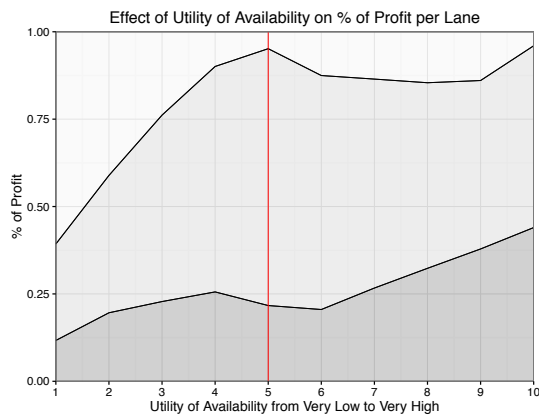
2.19 Relative Lane Performances with Three Lanes and (5%, 10%) Markup: Comparison across Scenarios



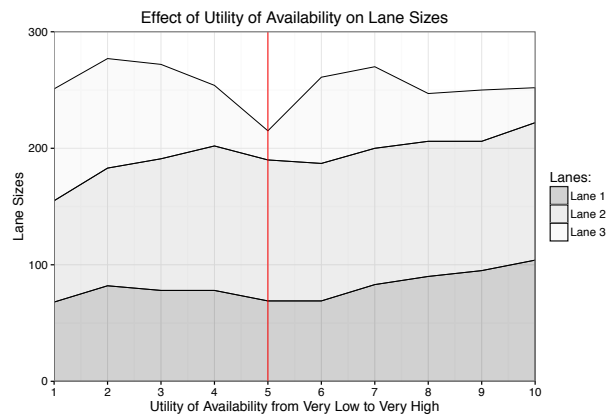
(a) Availability vs Profit per Lane, 100% Scenario



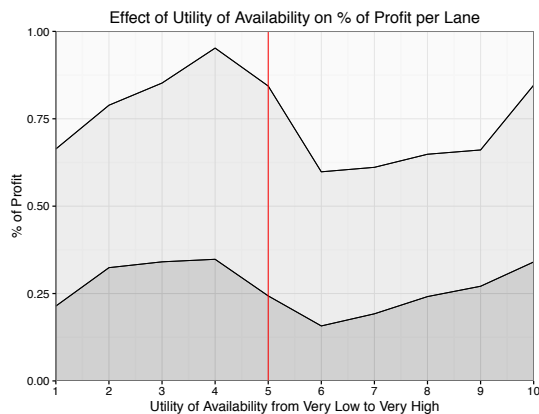
(b) Availability vs Lane Size, 100% Scenario



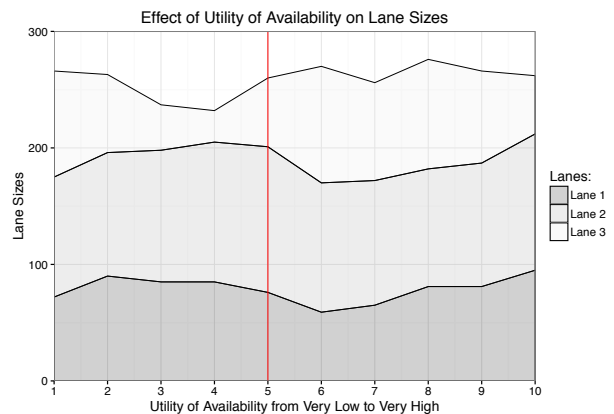
(c) Availability vs Profit per Lane, 75% Scenario



(d) Availability vs Lane Size, 75% Scenario

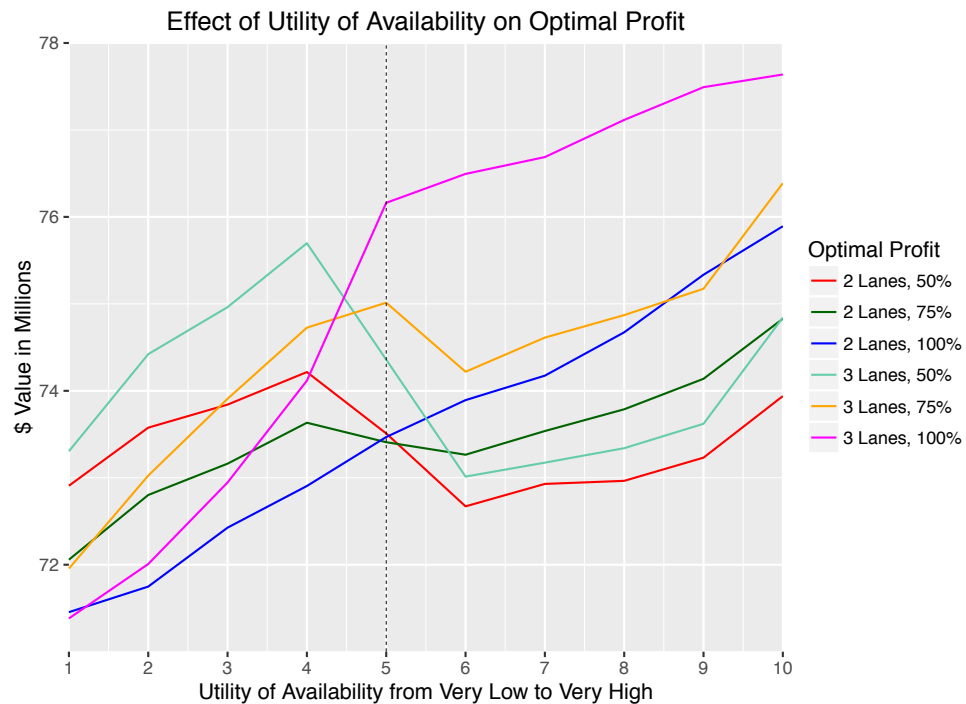


(e) Availability vs Profit per Lane, 50% Scenario

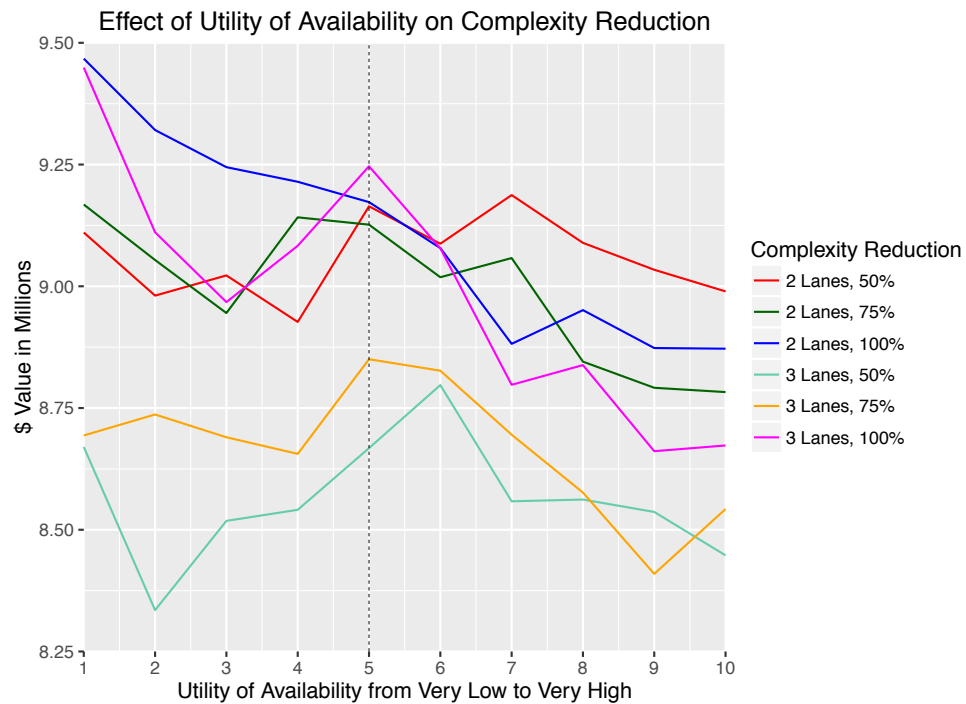


(f) Availability vs Lane Size, 50% Scenario

2.20 Comparing Two and Three Lanes across Different Scenarios on Profit and Complexity Reduction with Markups: 5% for Two Lanes, and (2%, 5%) for Three Lanes

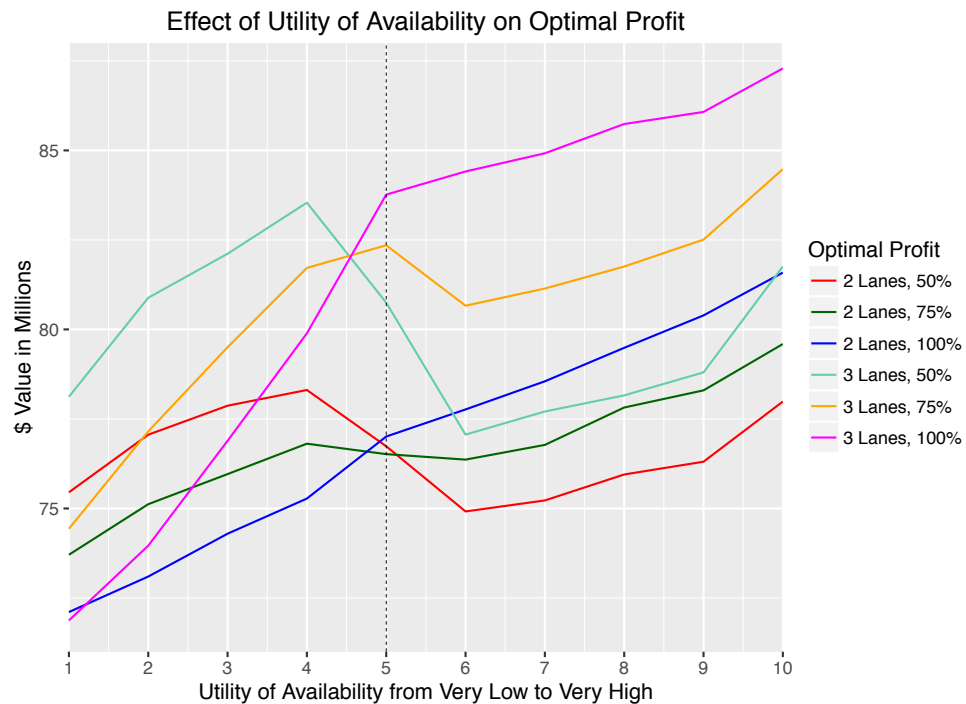


(a) Utility of Availability vs Optimal Profit

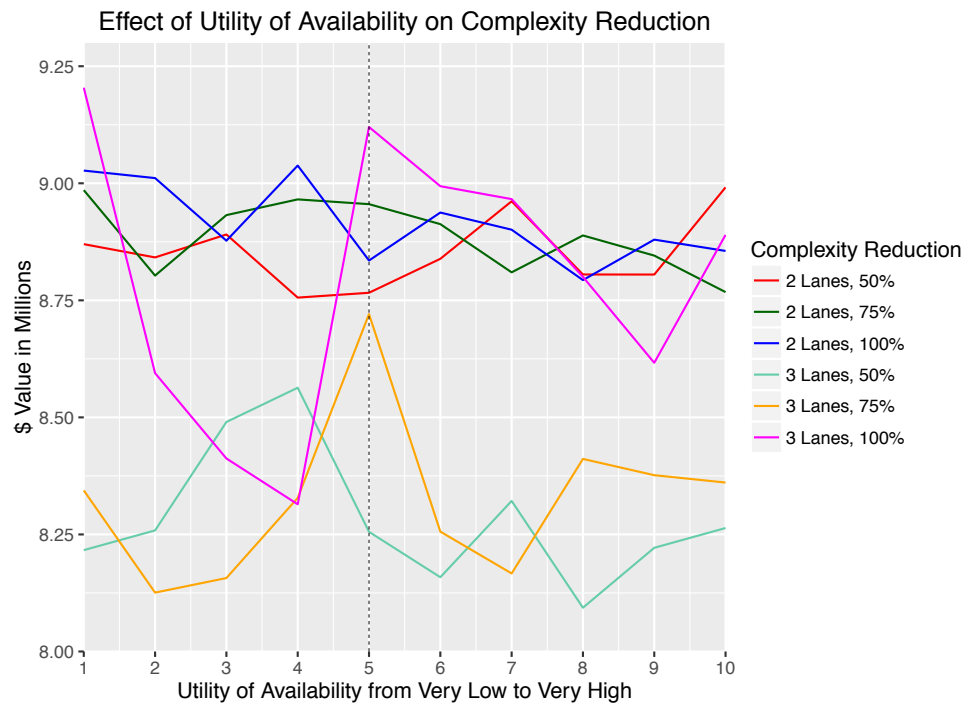


(b) Utility of Availability vs Complexity Reduction

2.21 Comparing Two and Three Lanes across Different Scenarios on Profit and Complexity Reduction with Markups: 10% for Two Lanes, and (5%, 10%) for Three Lanes

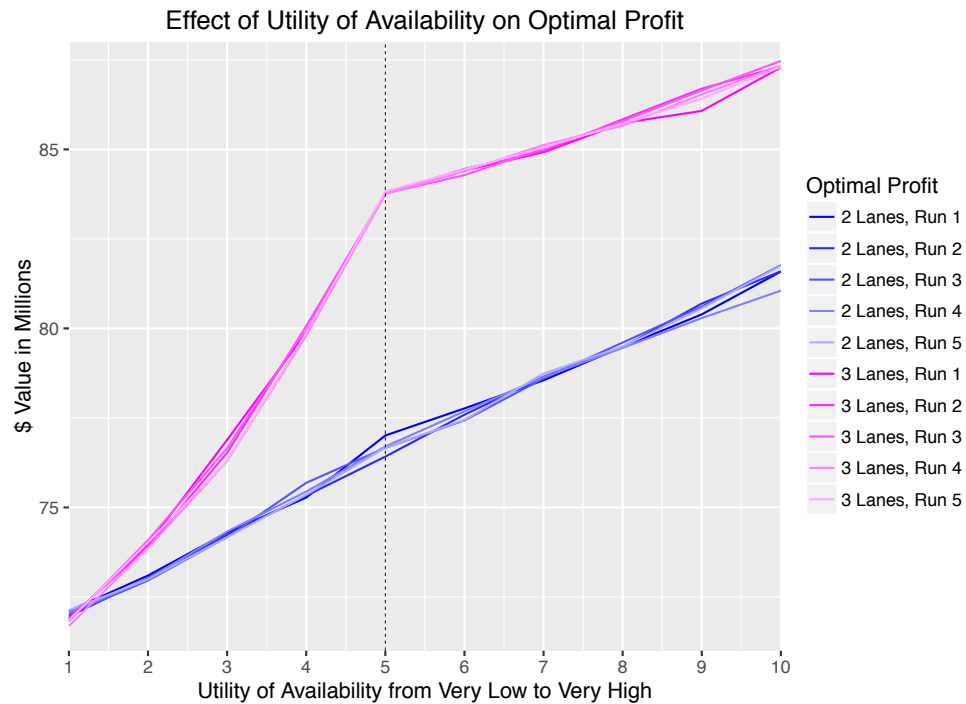


(a) Utility of Availability vs Optimal Profit

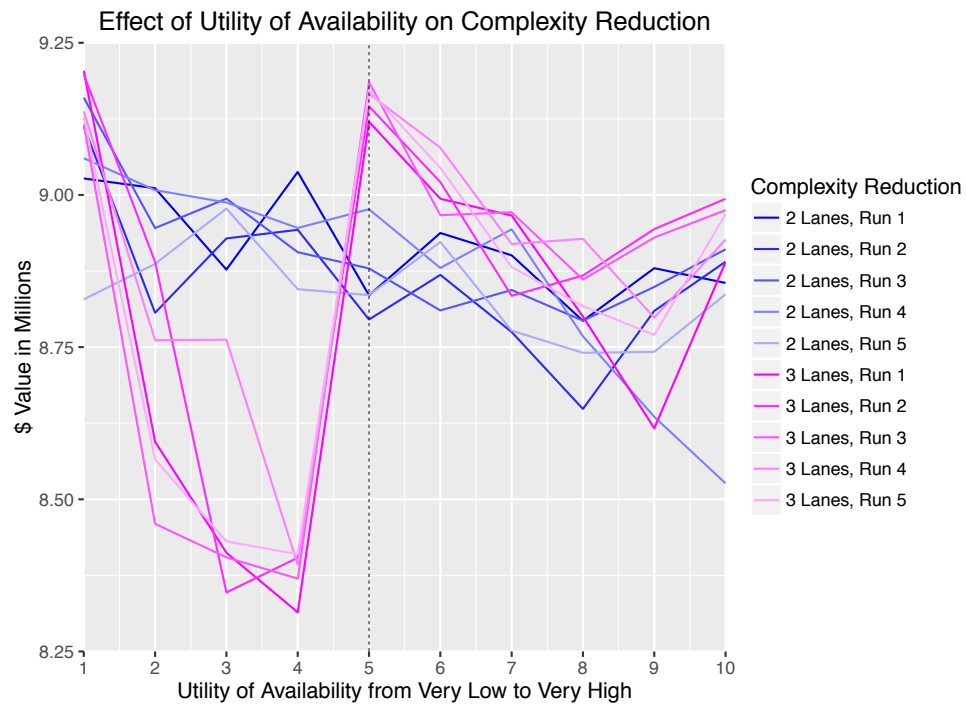


(b) Utility of Availability vs Complexity Reduction

2.22 Replications of the 100% Scenario for Two and Three Lanes to Compare Profit and Complexity Reduction with Markups: 10% for Two Lanes, and (5%, 10%) for Three Lanes



(a) Utility of Availability vs Optimal Profit



(b) Utility of Availability vs Complexity Reduction