TUHH Hamburg University of Technology

Hamburg University of Technology

Institute of Digital Economics

Prof. Dr. Timo Heinrich

Simulation-Based Optimization of Dynamic Pricing and Advertising Strategies for Online Retailers

Master Thesis

Author:	Rishon Noel Saldanha
Matriculation Number:	564720
Degree Program:	Mechanical Engineering and Management
First Examiner:	Prof. Dr. Timo Heinrich
Second Examiner:	Prof. Christian Thies

Hamburg, the 22.05.25

I, Rishon Noel Saldanha (student of M. Sc. Mechanical Engineering and Management at the TUHH, matriculation number 564720). Hereby assure that I have written this Thesis independently without outside help and have not used any other sources and aids other than those stated. Every passage taken literally or analogously from other works is marked with an indication of the source.

Place, Date Signature

I, Rishon Noel Saldanha (student of M. Sc. Mechanical Engineering and Management at the TUHH, matriculation number 564720), hereby grant TUHH and the members of TUHH the non-exclusive right, free of charge, to publish, reproduce, distribute, edit and publicize the contents of this thesis in whole or in part for the purposes of research and teaching. Any use beyond this is not permitted.

Place, Date Signature

I declare that this thesis was written independently and is the result of my own work. In the course of its development, OpenAI's ChatGPT was utilized as a supportive tool. This included assistance with refining the language, clarifying technical content, and helping to debug and troubleshoot code used within the thesis. The use of AI tools was strictly limited to supportive functions and in no way replaced or compromised the academic or scientific integrity of the work.

Place, Date Signature

Abstract

Online retail has become increasingly dynamic and competitive, making it more challenging for retailers to set effective prices and manage advertising budgets. Strategic consumer behavior, such as delaying purchases in anticipation of future discounts, and the unpredictability of customer preferences further complicate decision-making. This thesis investigates how retailers can dynamically coordinate pricing and advertising bids to maximize profit in the presence of consumer uncertainty and behavioral variability.

The study examines two strategic approaches: a consistent single-mode bidding strategy and a more adaptive multi-mode bidding strategy. These are evaluated under three consumer reservation price distributions: Gamma, Weibull, and Beta. To focus solely on the impact of different customer behavior patterns, all three distributions were adjusted to have the same average reservation price. This ensures that any variation in performance is due to differences in the distribution’s shape and spread rather than the average willingness to pay

To support this analysis, a simulation-based dynamic programming model was developed. The model evaluates retailer performance over fifteen time periods, measuring bid levels, pricing decisions, and profits across repeated iterations. This allows for a robust assessment of each strategy's effectiveness under varying market conditions.

Findings show that the distributional characteristics of consumer valuations play a critical role in determining which strategy is optimal. Single-mode strategies perform best in stable or price-sensitive markets, providing consistent outcomes with minimal volatility. Multi-mode strategies, while more variable, can generate higher profits in markets with premium consumer segments and skewed demand patterns. The simulation framework offers practical insights for retailers, helping them adapt pricing and bidding policies to market dynamics, inventory levels, and strategic time horizons in sponsored search environments.

Table of Contents

List of Figures

1 Introduction and State of Art

In today's digital economy, the success of many businesses depends on their ability to implement effective advertising strategies and allocate resources efficiently. The rapid advancement of internet technologies has significantly elevated the role of online advertising, making it a dominant channel for reaching consumers. As a result, a growing number of online retailers are leveraging internet-based platforms to promote their products and services. For many of these businesses, a substantial share of sales is directly influenced by investments in online advertising and sponsored search marketing. However, these businesses face a fundamental tradeoff: investing in advertising to increase consumer reach versus adjusting pricing strategies to manage revenue and demand. Finding the right balance between these two levers is essential for maximizing profitability in competitive and dynamic marketplaces.

Understanding this tradeoff is increasingly important as global spending on online advertising reaches unprecedented levels. Even marginal improvements in advertising efficiency or pricing precision can translate into meaningful gains in revenue and market share. Existing research, such as the model developed by Ye et al. (2015), provides a theoretical framework to analyze the interplay between dynamic pricing and advertising strategies. Their model frames the problem as an optimization task, in which firms must allocate limited resources between bidding for visibility and setting appropriate prices to maximize overall profit.

Building on this foundation, the thesis introduces a simulation-based model to examine how pricing and advertising bids jointly influence profit, inventory depletion, and consumer engagement over a multi-period selling horizon. By integrating different probability distributions for consumer reservation prices, the model captures varying demand patterns and evaluates strategy performance under uncertainty.

This work contributes in two key ways. First, it develops a dynamic simulation framework that models probabilistic consumer behavior in response to pricing and bidding decisions. Second, it offers actionable insights for online retailers by demonstrating how these strategies can be optimized over time and across inventory levels. The model effectively bridges theoretical constructs with real-world applications, supporting strategic decision-making in sponsored search advertising.

The chapters that follow are organized as follows. Chapter 2 outlines the theoretical underpinnings of dynamic pricing and bidding strategies and presents a review of relevant literature. Chapter 3 describes the simulation methodology and the model's implementation. Chapter 4 presents the results of the simulation study, highlighting key trends and actionable insights. Finally, Chapter 5 concludes with a summary of findings and outlines directions for future research.

1.1 Key Terms

This section defines and explains the fundamental concepts and terminologies used throughout the thesis. Understanding these key terms is essential for interpreting the models, simulations, and analysis presented in later sections. By clarifying these terms, this section establishes a foundational understanding that connects theoretical frameworks with practical implementations in the simulation model.

Dynamic Pricing: Dynamic pricing is a strategy that continuously updates prices in response to changing factors like consumer demand, market competition, and available inventory. It is widely used in industries like e-commerce, hospitality, and airlines, where demand elasticity significantly impacts profitability (Kopalle et al., 2023).

Sponsored Search Marketing: According to Jansen et al. (2013) sponsored search refers to a type of online advertising in which businesses pay to have their products or services prominently displayed in search engine results. It is also commonly known as paid search advertising, pay-per-click (PPC), or search engine marketing. Fain and Pedersen (2006) noted that sponsored search advertising has long served as the core business model for major search engines.

Fain and Pedersen (2006), further explain the basic elements of sponsored search, which include keywords, bids, the search engine, content, and the processes that gather data and charge advertisers based on their displayed content.

Consumer Reservation Price: Consumer reservation price is defined as the maximum amount a consumer is willing to pay to buy a product. It is a critical variable in dynamic pricing. Wang et al. (2007) give several definitions of consumer reservation price, including:
a) The lowest price at which a consumer decides against purchasing the product.
b) A price at which the consumer feels indifferent about making the purchase.
c) A threshold price at or below which the consumer is willing to buy one unit of the product.

Distribution Function: In probability theory and statistics, a probability distribution is a mathematical model that describes how likely different outcomes are in a given experiment (Everitt & Skrondal, 2006).

Two essential ways to describe a probability distribution are through the cumulative distribution function (CDF) and the probability density function (PDF). The probability density function provides the likelihood that the value of a random variable will fall within a specified range (Cuemath, n.d). The cumulative distribution function (CDF) is derived from the probability density function for a continuous random variable and gives the probability that the variable is less than or equal to a specific value (DeCross & Liou, n.d).

Bid Optimization: Bid optimization focuses on determining ideal bid amounts to maximize advertising visibility and profitability. Bids can be optimized by regularly monitoring performance, adjusting bids, and refining targeting strategies as needed (BuzzBoard, 2024).

According to LeadsView (2024), bid optimization involves strategically adjusting bids to target high-performing keywords, audience segments, and time periods that drive the best results. By leveraging automated bidding strategies, bid adjustments, and performance analytics, advertisers can improve advertising placements without overspending. Continuous monitoring and refinement of bids ensure that campaigns remain competitive and cost-effective, ultimately boosting return on investment (ROI) and driving higher-quality traffic to their websites.

Inventory Management: Inventory management is integral to dynamic pricing, where pricing decisions must account for inventory levels. According to Dadge et al. (2014), effective inventory management is essential for maintaining seamless business operations and achieving profitability.

On-and-off Bidding: On-and-off bidding alternates between aggressive and conservative bidding strategies to manage advertising budgets effectively. Ye et al. (2015) provided the theoretical basis for on-and-off bidding strategies. In simple terms, when demand is high, retailers place higher bids to maximize profits during peak times. When demand is low, they often choose not to bid. This bidding strategy helps retailers take full advantage of high-demand periods, allowing them to generate more revenue when people are willing to spend more.

Conversion Rate: Conversion rate measures the percentage of users who complete a desired action, such as a purchase, after interacting with advertising. In the context of sponsored search marketing, conversion rate is defined as the fraction of clicks that lead to the purchase of a product (Ye et al., 2015).

Click-Through Rate (CTR): Click-through rate (CTR) is a critical performance indicator in online advertising, representing the ratio of advertising clicks to impressions. According to Klipfolio (2025), CTR measures how often a keyword generates a click compared to the number of times it appears in search results. Google Ads has released benchmark CTRs by industry: B2B: 2.55%, E-commerce: 1.66%, Health & Medical: 1.79%, Real Estate: 2.03%, Technology: 2.38%, and Consumer Services: 2.40% (Irvine, 2024).

1.2 Literature Review

This section provides a focused review of existing research on dynamic pricing, bidding strategies, and inventory management, with particular attention to studies that employ simulation-based methods in online marketplaces. It begins by exploring broad theoretical frameworks and gradually narrows down to studies closely aligned with the focus of this thesis. The discussion progresses toward simulation-driven models that are directly relevant to the scope of this thesis. Special emphasis is placed on how simulation has been used to model advertiser behavior, auction dynamics, and the integration of pricing with inventory and budget constraints

Early studies such as Gallego and van Ryzin (1994) laid the theoretical groundwork by exploring revenue management models for perishable goods. Their work introduced optimization techniques aimed at maximizing revenue by aligning pricing strategies with inventory constraints. These foundational models were later extended to incorporate consumer behavior, demand uncertainty, and competitive dynamics.

The study also demonstrated that fixed-price policies can be optimal when sales volume increases or when the time horizon approaches zero. Numerical examples further showed that fixed pricing remains effective even when expected sales volume is moderate. Although this body of literature extensively examines dynamic pricing strategies under stochastic demand and finite time horizons, it often focuses on pre-determined price paths rather than continuously adaptive real-time pricing mechanisms.

Studies by (Ferreira et al., 2016) presented the case of the online retailer Rue La La as an example of how customer data and machine learning can be used to optimize pricing decisions on a daily basis. Unlike the fixed-price assumptions in earlier models, their approach involves dynamic pricing that adjusts based on real-time demand predictions. The authors highlighted the challenge of forecasting demand for first-exposure products, which make up the majority of Rue La La's sales. To address this, they developed a nonparametric demand prediction model and incorporated reference price effects into their price optimization algorithm. However, while this study provides valuable insights into dynamic pricing, it does not examine bidding strategies or interactive, real-time customer influence leaving a gap for further research in those areas.

The study by Smith et al. (2024) advances the discussion on dynamic pricing by highlighting how AI-driven price optimization enables real-time adaptability, customer segmentation, and market trend prediction that earlier models, such as Ferreira et al. (2016), did not fully address. While Ferreira et al. (2016) focused on daily price adjustments based on historical sales data, Smith et al. (2024) demonstrated how predictive analytics and artificial intelligence can inform pricing decisions proactively, resulting in more personalized and responsive strategies.

Earlier studies primarily concentrated on dynamic pricing in isolation, often overlooking the influence of advertising on consumer demand and sales outcomes. Integrating advertising strategies, particularly bidding for sponsored search placements along with dynamic pricing, creates opportunities for more comprehensive revenue optimization in online marketplaces.

Ye et al. (2015) provide an integrated framework in which dynamic pricing and advertising decisions, specifically through sponsored search marketing, are addressed simultaneously with the goal of maximizing profit. Unlike prior studies that treated pricing and advertising independently, this research highlights the strategic interdependence between a retailer's bid for search placement and its pricing decisions.

The central question is whether bids and prices should be used as complements or substitutes, depending on how changes in bidding affect customer traffic and willingness to pay. The authors demonstrated that optimal bidding and pricing policies can exhibit complex patterns, including smooth bid adjustments or sudden jumps, depending on customer arrival rates and reservation price distributions. They also introduced scenarios where on-and-off bidding patterns emerge, offering practical insights into how retailers can dynamically adjust both levers to optimize performance.

Theoretical models offer structural insights but often rely on restrictive assumptions regarding consumer behavior, demand distributions, or advertiser rationality. Simulation-based studies help relax these constraints by enabling researchers to explore dynamic strategies in more realistic environments. Schwabe (2013) presents a simulation-based framework that models how advertisers adjust their bids over time in sponsored search auctions under the assumption of bounded rationality. Rather than assuming advertisers always behave optimally, the study acknowledges that real-world decision-makers often operate with limited information, computational constraints, or rely on heuristic learning.

They introduced a probabilistic model of bidder behavior and used Monte Carlo simulation to estimate the mixed strategies that such advertisers are likely to adopt in future rounds. By calibrating their model with real search engine log data, they demonstrated improved predictive performance in forecasting bid adjustments and revenue outcomes compared to traditional models based on perfect rationality or static best-response dynamics. This work underscores the value of incorporating realistic behavioral assumptions into simulation-based auction modeling

Tunuguntla et al. (2019)explored the integration of bidding and pricing strategies for a retailer selling a perishable product through sponsored search advertising. They developed a stochastic dynamic programming model, supported by simulation, to determine the optimal joint bidding and pricing policy over a finite selling horizon, explicitly accounting for inventory constraints.

Their results reveal a key insight: when inventory levels are low, it is optimal to bid more aggressively to capture limited demand quickly. Conversely, when inventory is high, lowering prices becomes more effective in stimulating sales. The study also finds that the mean and variance of consumer willingness to pay influence the optimal strategy. Greater uncertainty in consumer value justifies higher bids and prices to balance risk and reward.

By combining dynamic pricing, advertising expenditure, and inventory awareness in a unified framework, this work demonstrates how demand uncertainty and inventory status interact to shape optimal marketing decisions.

Salehi and Mirmohammadi (2023) extend existing research on dynamic pricing and bidding by introducing explicit budget constraints into the retailer's decision-making process. They propose a stochastic dynamic programming model that jointly optimizes advertising bids and selling prices for a product with finite inventory, while adhering to a fixed advertising budget over the campaign horizon. To solve this problem efficiently, the authors exploit structural properties of the value function and develop a tailored algorithm that significantly reduces computational complexity compared to traditional brute force methods.

Through simulation, they demonstrate that the algorithm consistently identifies optimal or near optimal solutions. Notably, increasing inventory size does not always increase the computational burden, contrary to common expectations. Their findings emphasize the importance of strategically distributing advertising spend over time, particularly when resources are limited, to maintain profitability and prevent premature budget exhaustion. This contribution highlights the operational value of incorporating financial constraints into tactical marketing decisions.

Continuing from the review of earlier studies on dynamic pricing and advertising Agrawal et al. (2023) offers a significant advancement by integrating advertising strategies within dynamic pricing models. This research explores how sellers can not only set prices but also use advertising schemes that influence buyer perceptions of product quality. By applying the Bayesian persuasion framework, the authors developed an online algorithm that learns adaptively to optimize both pricing and advertising strategies in real-time. This allows the seller to maximize expected revenue by strategically aligning advertising and pricing decisions based on consumer learning. The paper presents an innovative approach to combining information design with dynamic pricing, offering valuable insights into how real-time adaptations in both levers can significantly improve revenue outcomes.

While prior research has demonstrated the value of simulation in sponsored search marketing, most existing models focus either on auction mechanisms or on optimization under well-defined structural assumptions. Studies such as Tunuguntla et al. (2019) and Salehi and Mirmohammadi (2023) integrate dynamic pricing and advertising decisions in inventory-constrained environments, but often treat demand in a stylized manner, assuming known probabilistic models or simplified consumer responses. Similarly, Schwabe (2013) simulates bidder behavior under cognitive limitations, but does not incorporate inventory dynamics or joint pricing mechanisms. These models are typically designed to derive optimal policies or improve algorithmic efficiency, rather than to explore how different strategies perform under uncertainty in realistic retail settings.

In contrast, this thesis takes a behavioral and exploratory simulation approach to examine how alternative bidding and pricing strategies perform over time in a setting with strategic consumers and evolving inventory building on the integrated pricing and bidding framework proposed by (Ye et al., 2015).

2 Theoretical Model Framework

This section presents the theoretical model proposed by Ye et al. (2015), which serves as the foundation for the simulation analysis in this thesis. The model captures the joint decision-making process of pricing and bidding in a finite-horizon setting, with the objective of maximizing retailer profit.

Rather than treating pricing and advertising independently, the framework integrates both elements to reflect their interdependence in sponsored search environments. It provides a structured representation of how firms can respond dynamically to inventory levels, consumer behavior, and time-based constraints. This formulation forms the basis for the simulation conducted in later chapters, offering a realistic approach to understanding profitability under uncertainty.

2.1 Theoretical Model: Foundations and Assumptions

The retailer's problem is modeled as a dynamic program, where the objective is to maximize expected profit across a finite time horizon *T*. The model consists of three components: state variables, decision variables, and an objective function.

State Variables:

State variables describe the current condition of the system at any given time. In this model, the state variables include the inventory level and the maximum expected profit at each time period.

y_t: Inventory level at the beginning of period *t*.

$\Pi(y,t)$: Maximum expected profit when starting period *t* with *y* units of inventory.

Decision Variables:

Decision variables are the controllable inputs that the retailer actively selects in each time period to influence outcomes. In this model, they include the product price and the bid amount for advertising.

p: Price of the product.

b: Bid amount for sponsored search advertising.

Objective Function:

The objective function defines the retailer's goal is to maximize the total expected profit over the finite selling horizon by optimally choosing the pricing and bidding decisions at each time period.

The retailer's goal is to maximize profit over the selling horizon by selecting optimal pricing (*p*) and bidding (*b*) strategies:

The Objective function described in Ye et al. (2015) is as follows ,

$$\begin{aligned}\Pi(y,t) = max_{(b\geq 0\,,p\geq 0\,)}\{ & \big((1-\lambda(b)\big)\,\Pi(y,t-1) + \lambda(b)\,F(p|b)(\Pi\,(y,t-1)-b) \\ & + \lambda(b)\,\bar{F}(p|b)(p + \Pi(y-1,t-1)-b)\,\}\,for\ y>0\,,for\ t=1,\dots.T\end{aligned}$$

Where:

$\lambda(b)$: Probability of a consumer clicking on the advertising (depends on *b*).

$\bar{F}(p|b)$: Cumulative probability of a consumer purchasing at price *p*, given bid *b*.

The model relies on several key assumptions to simplify the complex decision-making process

1. The selling period is limited to a fixed number of time intervals, and decisions are made at the beginning of each period to reflect real-world retail constraints.
2. Consumers are assumed to have a reservation price influenced by both the product's price and its visibility through advertising, meaning higher bids can make higher prices more acceptable.
3. Inventory decreases only when a sale is made, and no new stock is added during the entire selling horizon, requiring careful planning of stock usage over time.
4. The probability that a consumer clicks on an advertisement depends on the bid amount, with higher bids improving visibility and increasing the chance of a click.
5. Whether a consumer makes a purchase after clicking is uncertain and modeled using probability distributions such as Beta, Gamma, or Weibull, reflecting varying demand patterns.

These assumptions formed the foundation for deriving the dynamic programming equations.

This equation can be broken down into three distinct cases, each reflecting different customer behaviors and their impact on profit:

First Term: No customer clicks

$$\left(1-\lambda(b)\right)\Pi(y,t-1)$$

This term represents the scenario where no customer clicks on the advertising. Here, the probability of no click is $\left(1-\lambda(b)\right)$, where $\lambda(b)$ is the likelihood of a click depending on the bid amount b. Since no purchase occurs, the inventory remains unchanged, and the expected profit remains $\Pi(y,t-1)$. which is the profit from the previous period. This means that the retailer incurs no cost or revenue in this case, as there's no interaction with the customer.

Second Term: Customer clicks but does not purchase

$$\lambda(b)\,F(p|b)(\Pi(y,t-1)-b)$$

This term covers the case where a customer clicks on the advertising but does not complete the purchase. Here:

The probability of a click is $\lambda(b)$, and the probability of no purchase is $F(p|b)$, which represents the cumulative distribution function (CDF) of the reservation price (i.e., the likelihood that the consumer's willingness to pay is below the price *p*). The retailer incurs the bidding cost *b* for the click but earns no revenue. The profit decreases by *b*, reflecting the incurred cost for the click. The expected profit to go remains as $\Pi\,(y,t-1)$, given the unchanged inventory and the continuation of the sales horizon.

The second term highlights the cost associated with attracting customer attention without securing a sale.

Third Term: Customer Clicks and Purchases

$$\lambda(b)\,\bar{F}(p|b)(p+\ \Pi(y-1,t-1)-b)$$

This term reflects the situation where the customer clicks the advertising and completes the purchase.

The probability of a click is $\lambda(b)$, and the probability of purchase is 1- $F(p|b)$, where 1- $F(p|b)$ indicates that the consumer's reservation price exceeds the product price p. The retailer earns revenue equal to price p. The profit to go is $\Pi(y-1,t-1)$ reduced by the bidding cost *b*, as one unit of inventory is sold. The third term captures the profitable scenario where both the advertising strategy and pricing align with consumer behavior.

The retailer selects the optimal bid and price to maximize the expected profit, which is computed as a weighted sum of three components. Each component represents a different outcome of consumer interaction no click, click without purchase, and click followed by purchase making the model responsive to changes in both consumer behavior and market conditions. By solving the dynamic program across all periods and inventory levels, the model provides strategic insights into how much a retailer should bid and what price to set. By evaluating these terms for each period and inventory level, the model provides insights about how much can a retailer bid. It's to be noted that Higher bids increase *λ(b)* attracting more clicks but at a higher cost. It's also providing insights about what price to set. Lower prices reduce *F*(*p*|*b*), increasing the likelihood of purchase but potentially reducing per-unit revenue.

2.2 Empirical Analysis: Graphical Validation and Insights

This section replicates and validates dynamic programming results from Ye et al. (2015), specifically focusing on two optimal bidding strategies: single-mode and multi-mode bidding. The objective is to assess their performance under varying inventory conditions and consumer behaviors modeled by different reservation price distributions. This gives a solid foundation for our simulation.

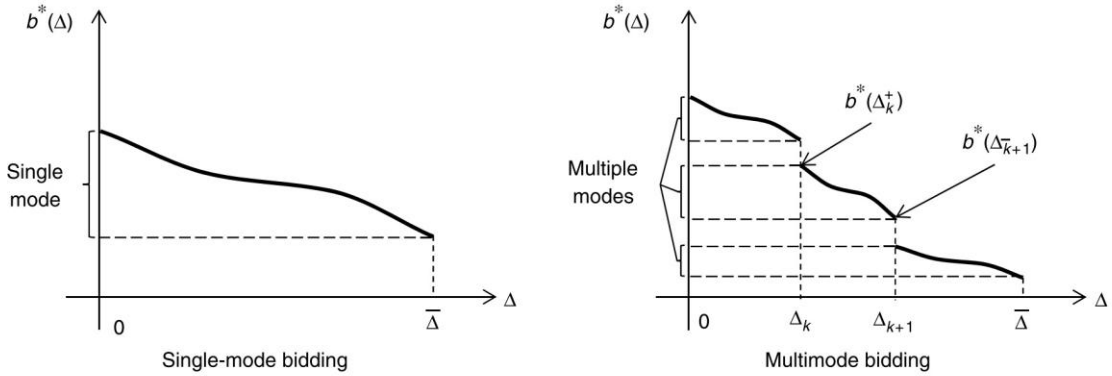

Figure 1 : Two types of bidding strategies

As seen in the figure 1, single-mode bidding represents a smooth, continuous approach to bid adjustments. In this structure, bids (b^*) decrease gradually and consistently as time progresses or inventory levels decline. At the beginning of the decision horizon, when inventory is abundant, bids are set at relatively high levels to attract consumer traffic and maximize sales opportunities. As inventory is depleted or as the selling period approaches its end, bids are progressively reduced, ensuring that advertising costs remain aligned with the diminishing need for traffic. This strategy is particularly effective in stable market conditions where demand patterns are predictable, allowing businesses to implement a straightforward and cost-efficient bidding policy. The smooth nature of single-mode bidding makes it easier to forecast advertising expenses and align them with overall revenue goals, thus offering simplicity and predictability.

As previously described in Section 1.2, multi-mode bidding adjusts bids discretely at predefined thresholds such as time milestones or inventory levels. This structure enables greater flexibility in responding to dynamic market changes, such as shifts in demand or competitive pressure, making it particularly relevant in volatile retail environments.

The choice between single-mode and multi-mode bidding depends largely on market conditions. Single-mode bidding is ideal for stable environments with predictable demand, offering simplicity, cost efficiency, and consistent engagement. Conversely, multi-mode bidding is more effective in dynamic settings, where flexibility is crucial to capitalize on sudden shifts in consumer behavior or inventory constraints.

Ye et al. (2015) demonstrate that while single-mode bidding minimizes complexity, multi-mode bidding offers enhanced adaptability. Their comparative analysis highlights how strategic alignment of bidding with inventory levels and demand variability can improve overall profitability in sponsored search advertising.

Below are the steps outlining how the theoretical model is implemented and solved in a way that closely replicates the approach presented in the original paper (follow appendix A). This step-by-step breakdown follows the logic of the model and translates it into a computational process that allows us to simulate, analyze, and visualize the behavior of optimal bidding strategies under different scenarios. Each step corresponds to a key part of the algorithm used in the code to bring the theoretical ideas into practice.

Step 1: Parameter Initialization

The model begins by defining the key parameters and probabilistic functions.

A Gamma distribution is used to model price uncertainty, with shape parameter α=6 and scale parameter β=0.5. The time horizon is set to T=15, representing discrete decision periods and the maximum inventory level is denoted by *Ymax*. Additionally, the following functions are defined to capture the dynamics of the bidding environment as stated in the paper. Other details are as follows.

- **Expected Price Function**:
  $\mu(b)$=0.6*b, this represents the expected selling price as a function of the bid amount b.
- **Price Variability Function**:
  $\sigma(b)$=1+0.1, the standard deviation of price which increases with the bid value.
- **Bid Success Probability**:
  $\lambda(b)$=0.8−0.7 this models the likelihood of winning a bid as a function of its value.

Step 2: Profit Matrix Initialization

A two-dimensional profit matrix $\Pi\,[y,t] \in (Ymax+1) \times (T+1)$ is initialized with zeros. Each entry $\Pi[y, t]$ represents the maximum expected cumulative profit when holding y units of inventory at time period T.

Step 3: Dynamic Programming Loop

The core of the algorithm uses dynamic programming to calculate the best possible profit at each time period from 1 to T, and checking every possible inventory level from 1 to the maximum (*Ymax*). At each of these points, the algorithm builds an objective function, this is a formula that helps decide what bid and price should be chosen to get the highest expected profit.

The function takes into account three possible situations: (1) no click, where the bid fails and profit from the previous period is carried forward with unchanged inventory, (2) click without purchase, where the bid incurs a cost but no revenue is gained, and (3) click with purchase, where revenue is earned, inventory is reduced by one unit, and profit is adjusted accordingly.The objective function is numerically minimized (since it is defined as the negative of profit) over the variables *b* and *p*. The resulting maximum expected profit is stored in Π [*y*, *t*], reflecting the best outcome after accounting for possible bid costs and inventory changes.

Step 4: Extracting Optimal Bids

Once the profit matrix is fully computed, the algorithm traces back the optimal bidding decisions over time.The optimal bid values (*b**) obtained are stored to analyze bidding behavior under different inventory constraints.

Step 5: Visualization of Bidding Strategy

The final step involves visualizing how the optimal bid evolves over time:

A line plot is generated with:

X-axis: Time periods (from *T* down to 1).

Y-axis: Optimal bid values *b**.

Legend: Separate curves for each inventory level analyzed (e.g., 2, 4, and 6)

This visualization highlights the impact of both inventory pressure and time urgency on bid aggressiveness. As inventory depletes or as the time horizon shortens, the optimal bid strategy adjusts accordingly to maximize expected profit.

Parameter settings used in example 1 are $\sigma(b)$ = 1+ 0.1b, it is distributed with Gama distribution with α=6 and β =0.5, $\mu(b)$ = 0.6b and $\lambda(b)$ = 0.8 – 0.7e^{-b}.

Parameter settings used in example 2 are $\sigma(b)$ = 1+ 0.1b it is distributed with Gama distribution with α=6 and β =0.5, $\mu(b)$ =0.4b and $\lambda(b)= \frac{(1+0.4e^{4-4b})}{(1+e^{4-4b})}$ there curves which are being recreted corresponds to level *I* = 2,4 and 6 for both example 1 and example 2.

To replicate the optimal bidding strategies illustrated in Ye et al. (2015), SciPy's optimization solver (minimize) was initially implemented, applying it within a dynamic programming framework to compute the optimal bid and price at each time and inventory state. The resulting plots for Example 1 (single-mode) and Example 2 (multi-mode) were not exactly identical to the reference figure from the paper (Figure 1), but they showed the same structural behavior namely, a smooth, monotonic increase in optimal bids in Example 1, and a discrete, step-like bidding policy in Example 2. While the absolute bid values were slightly lower in some time periods (typically within a difference of ±0.1–0.3), the overall shape, timing, and logic of the bidding curves were consistent.

This raised an important question: Are these deviations a result of an implementation error or simply due to numerical differences? To address this, I re-implemented the simulation using an alternative method: grid search. Instead of relying on a numerical optimizer, this method iterated through a fixed grid of possible bid and price values, ensuring an exhaustive evaluation of the objective function.

The key insight is that both methods SciPy optimization solver and the grid-based search yielded near-identical results, thereby confirming the correctness of the implementation. Furthermore, both methods independently reproduced the defining patterns of Example 1 and Example 2: the gradual slope of the single-mode policy and the stepped behavior of the multi-mode strategy.

The minor differences from the paper's plots likely stem from small variations in numerical methods, grid resolution, function scaling, or missing parameter fine-tuning that were not fully disclosed in the original paper. Nevertheless, the consistency across methods and the qualitative alignment with the published figures offer strong evidence that the model is both correctly implemented and reliable for simulation-based analysis.

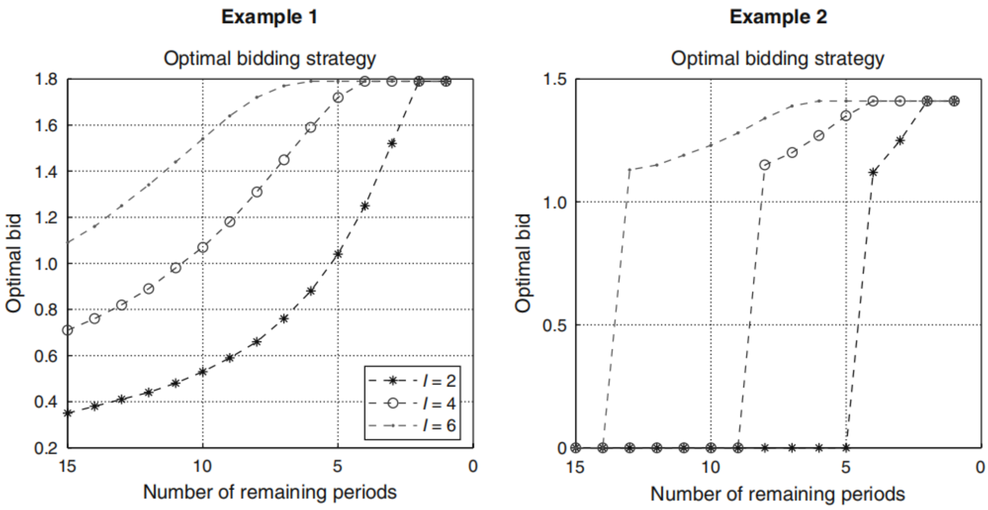

Figure 2 : Graph extracted from (Ye et al., 2015).

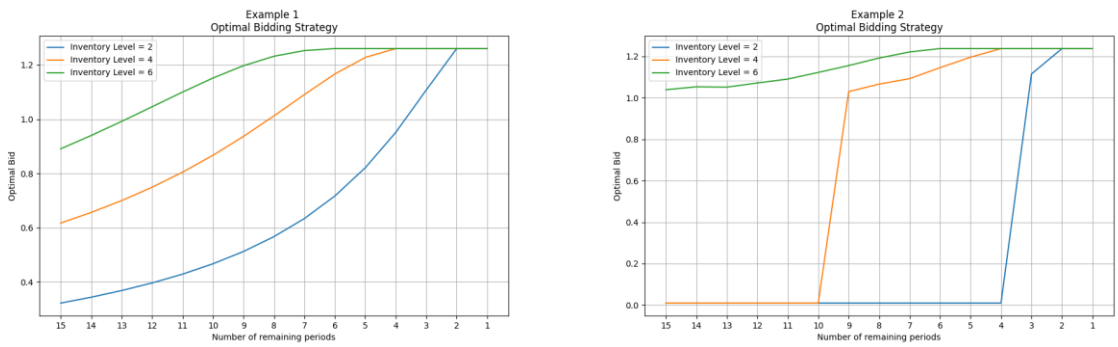

Figure 3 : Model solved using SciPy optimizer.

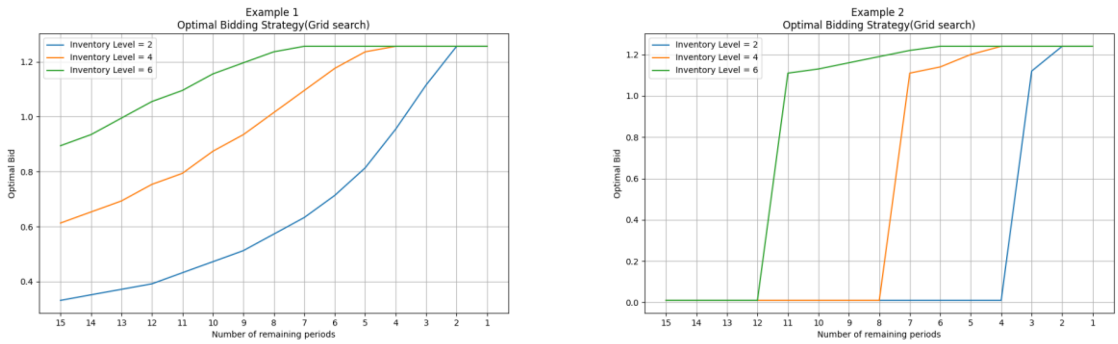

Figure 4 : Model solved using Grid search.

The recreated graph for the single-mode bidding structure demonstrates a clear trend where the optimal bid increases as the number of remaining time periods decreases. This reflects the retailer's increasing urgency to attract consumers and clear inventory as the selling horizon shortens. The smooth progression of bids aligns with the theoretical expectations of single mode bidding. Additionally, for higher inventory levels, the optimal bid remains elevated throughout the time horizon, signaling the retailer's need to adopt a more aggressive approach to sell larger stock. This behavior is consistent with the findings in the original graph, though the recreated graph exhibits slightly higher bid levels for inventory levels of 4 and 6. These deviations could stem from differences in parameter values, such as the scale or variability of the bid function, which influence the numerical outcomes of the simulation.

For the multi-mode bidding structure, the recreated graph effectively captures the step-like pattern indicative of the on-and-off bidding strategy. This strategy alternates between higher bids during critical periods, aimed at attracting consumers, and lower bids or zero bids during less critical intervals to conserve resources. The recreated graph showcases these alternating regions, mirroring the strategy's objective of balancing bidding costs with inventory and time constraints. The use of aggressive bidding during specific intervals aligns well with the behavior described in the original graph. However, the transitions in the recreated graph are smoother compared to the original, where sharper discontinuities are observed between high and low bids. These differences might result from variations in numerical precision or the constraints applied in the simulation.

The successful recreation and validation of the theoretical model through graphical comparison provide a solid foundation for advancing to the simulation phase of this study. By confirming the alignment between the theoretical predictions and the reproduced results, the robustness of the optimization framework has been established. This validation allows us to confidently proceed to the simulation analysis, where we expand upon the theoretical findings by incorporating real-world complexities, such as stochastic consumer behavior and dynamic interactions between pricing, bidding, and inventory constraints.

The simulation phase will provide empirical insights into the practical implementation of the model under varying scenarios, offering actionable recommendations for businesses aiming to optimize their advertising and pricing strategies in competitive marketplaces.

3 Simulation Framework

This chapter presents the simulation framework developed to translate the theoretical model into a practical, computational tool. The simulation aims to capture the dynamic interaction between pricing, bidding strategies, and stochastic consumer behavior across a finite selling horizon. By incorporating probability distributions and randomized events, the simulation mimics real-world market uncertainties and provides empirical insights into optimal decision making. The structure of this chapter first discusses the role of probability distributions in modeling consumer behavior, followed by a detailed explanation of the simulation design, logic, and implementation.

3.1 Incorporating Probability Distributions in Simulations

Consumer behavior in online marketplaces is often unpredictable, influenced by factors such as pricing strategies, advertising visibility, and individual preferences. To capture this uncertainty, probability distributions play a crucial role in modeling key variables like consumer demand, reservation prices, and click-through rates. By integrating probability distributions into optimization models, businesses can better understand and anticipate consumer behavior, enabling more informed decision-making in pricing and advertising strategies. Three probability distributions are taken into study namely Gamma, Beta and Weibull. These 3 distributions are taken because they are discussed in (Ye et al., 2015).

The gamma distribution is one of the continuous distributions in which the distributions are very versatile and give useful presentations of many physical situations. They are perhaps the most applied statistical distribution in the area of reliability (U Eric et al., 2021). The Gamma distribution is defined by two parameters: the shape parameter (α) and the scale parameter *(θ)*.

The corresponding probability density function (PDF) in the shape-rate parameterization is $f(\mathrm{x}) = \frac{1}{\Gamma(\alpha)\theta^{\alpha}} x^{\alpha-1} e^{\frac{-x}{\theta}}$ (U Eric et al., 2021) for Simulation Design and Implementation x, α, Θ>0 where *f(x)* is the gamma function.

The shape parameter (α) controls the overall form of the distribution. Higher values result in a more symmetric appearance. The scale parameter *(θ)*, on the other hand, affects how spread out the distribution is; increasing this parameter broadens the range, while decreasing it produces a tighter, more concentrated curve.

The Weibull distribution is one of the best-known lifetime distributions. It effectively describes observed failures of many different types of components and phenomena. Over the last three decades, numerous articles have been written on this distribution. Several reviews present historical facts, the various forms of the distribution used by practitioners, and the possible confusions and errors arising from its non-uniqueness. Comprehensive studies have been dedicated to systematically exploring this distribution. More recently, monographs have been published that cover nearly every aspect of the Weibull distribution and its extensions. (Lai et al., 2006)

The corresponding probability density function (PDF) is $f(t) = \frac{\beta}{\theta^{\beta}} t^{\beta-1} e^{-(\frac{t}{\theta})^{\beta}}$ where t>0 Luko (1999).

Where the variable t represents the time until failure, which could refer to the number of cycles, actual time, or similar measures. The Weibull distribution is defined by the shape parameter β and scale parameter θ (Luko, 1999).

The beta distribution is a continuous probability distribution that models random variables within a finite interval. It is commonly used to represent the time required to complete a task, the distribution of order statistics, and the prior distribution for binomial proportions in Bayesian analysis. Typically, the standard beta distribution is defined over the interval [0,1], making it especially suitable for modeling probabilities in experiments with two possible outcomes. However, it can also be adapted to other intervals as needed. To accommodate various applications, several generalized forms of the beta distribution have been developed, including models that extend the standard beta distribution with additional parameters (H et al., 2023).

Probability density function of beta distribution is $f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}$ (Sugiyama, 2016) where $B(\alpha,\beta) = \int_0^1 x^{\alpha-1}\,(1-x)^{\beta-1}\,dx$ is a beta function where α and β are positive integers.

When comparing the impact of different consumer reservation price distributions on optimal bidding and pricing strategies, it is critical to isolate the influence of distributional shape (e.g., skewness, tail behavior) from the central tendency. To achieve this, all three distributions Gamma, Beta, and Weibull are parameterized to have the same expected value (mean), approximately 3.

Normalizing the mean makes sure that any changes in the simulation results are due to how reservation prices are spread out, not because people are willing to pay more on average. Without this step, a distribution with a higher average would naturally lead to higher profits, but that would mix up two different effects how different customers are and how much they can spend. This would hide the real insights the study is trying to find. By standardizing the mean across distributions:

- Comparative fairness is achieved in evaluating model behavior.
- The interpretability of policy outcomes improves, as bidding and pricing adaptations can be directly traced to distribution shape (e.g., long tail vs. left skew).
- It reflects a realistic benchmarking practice where different consumer segments may have the same average value but exhibit varying behavior patterns around that average.

This normalization is especially important in a setting where the firm must learn or predict how to allocate limited advertising budget over a finite time horizon. If one distribution implied a higher baseline profitability, it would distort the true effect of consumer behavior variability on optimal strategy.

Importantly, the Gamma distribution with parameters α=6 and β=0.5 is adopted directly from Ye et al. (2015) The mean of a Gamma distribution is calculated as:

Mean(μ) =$\alpha \cdot \beta$ (Fader & Hardie, 2005)   if we multiply we get 3.

To ensure parity, the Beta and Weibull distributions were parameterized to match this expected value:

- Beta Distribution (scaled to [0, 10])

The mean of a Beta distribution scaled to the interval [0,10]. Mean (μ) = a ($\frac{\alpha}{\beta+\alpha}$) (Fader & Hardie, 2005) solving from this equation we get α and β value as 3 and 7.

- Weibull Distribution

The mean of a Weibull distribution is calculated using the Gamma function:

Mean (μ) = $\alpha \cdot \Gamma(1+\frac{1}{\beta})$ (Pinelis, 2018) solving this we get values α and β value as 2 and 3.35.

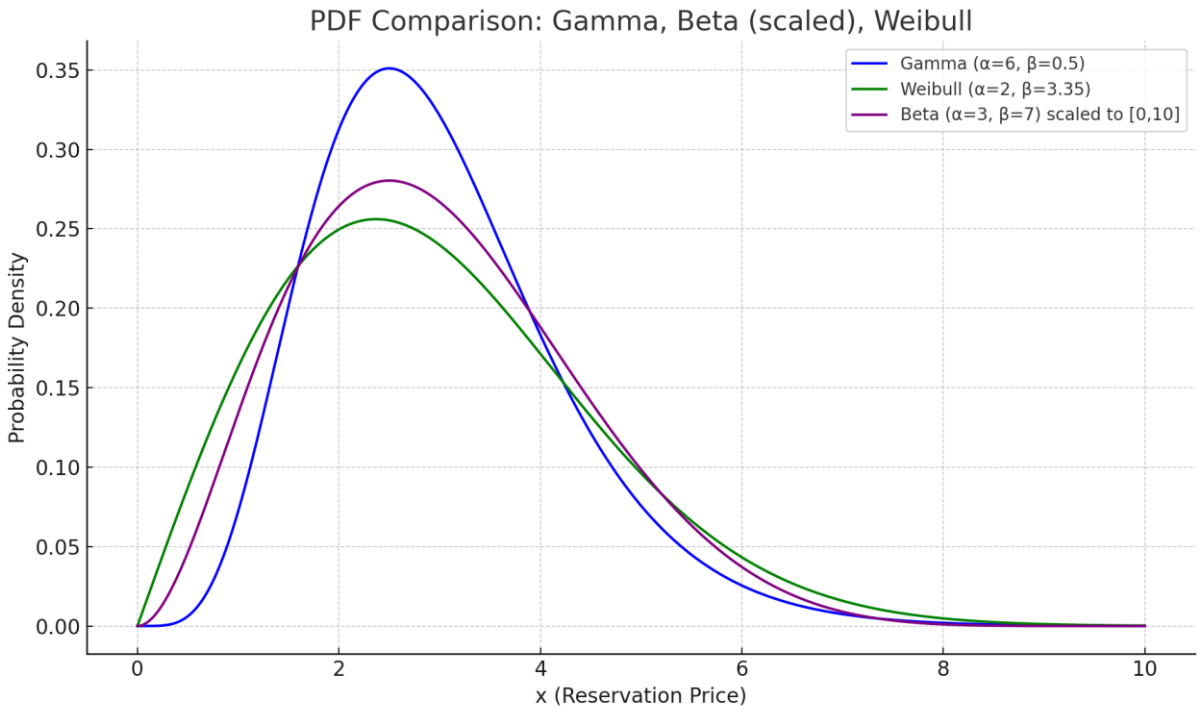

Figure 5 : Comparison of PDF (Gamma, Scaled Beta and Weibull).

Figure 5 illustrates the probability density functions (PDFs) of the Gamma, Weibull, and Beta (scaled) distributions used in the study, all normalized to share the same expected value of approximately 3.0. Despite this shared mean, the figure reveals distinct differences in the shape and tail behavior of each distribution. The Gamma distribution exhibits a moderately symmetric form with a pronounced peak and gradual decline, representing a balanced consumer base centered around the mean. The Weibull distribution displays a longer right tail, indicating a small but significant probability of encountering high reservation price consumers. In contrast, the Beta distribution is concentrated near lower values and tapers off quickly, reflecting a price sensitive consumer segment with very few high WTP individuals. This visual comparison underscores the rationale behind normalizing the means ensuring that observed differences in strategy performance stem from structural differences in consumer behavior, rather than variations in average willingness to pay.

3.2 Simulation Design and Implementation

This section outlines the implementation of the simulation framework used to evaluate the optimal bidding and pricing strategies under stochastic consumer behavior. Building on the theoretical model and dynamic programming approach described earlier, the simulation introduces randomness to reflect real-world uncertainties in consumer actions. Specifically, it aims to capture the interaction between inventory levels, bid values, price decisions, and probabilistic consumer responses across a finite time horizon.

The simulation runs for T=15 time periods, with a maximum inventory level of $Ymax$=6. For each inventory level and time period, the optimal price and bid decisions are obtained using numerical optimization. These values are then used to simulate consumer behavior over 1000 iterations, generating a range of possible outcomes that provide empirical insights into system performance (follow appendix B).

The core logic of the simulation is based on generating two independent random numbers in each time period. The first random number determines whether a consumer clicks on the advertisement. If a click occurs, the second random number determines whether the consumer proceeds to make a purchase. This structure reflects real-world user behavior in online marketplaces, where not every advertising impression leads to a click, and not every click results in a sale.

More precisely, for each simulation run and time period, the model generates two random numbers. The first random number is compared with the click probability $\lambda(b)$, which depends on the bid amount b. If first random number is $<\lambda(b)$, a click is registered. If this condition is not satisfied, no click occurs, and no further action is taken for that period the profit remains zero, and inventory is unchanged.

If a click occurs, the second random number is compared with the purchase probability $1-F(p|b)$, where $F(p|b)$ is the cumulative distribution function representing the probability that the consumer's reservation price is less than the posted price p. If second random number $<1-F(p|b)$, the consumer purchases the product. In this case, the profit is calculated as the difference between price and bid, and inventory is reduced by one unit. If the second condition fails, a click occurs but no purchase is made, resulting in a loss equal to the bid amount.

The profit matrix (Π) is updated accordingly in each of the three possible cases:

- If no click occurs: profit matrix for that period updates to 0
- If a click occurs but no purchase then $-b$.
- If both click and purchase occur then $p-b$.

The inventory matrix is updated only when a purchase occurs, decreasing the available inventory by one unit. In all cases, the bid and price used in that period are stored. This structure ensures that each simulation run captures a complete sales trajectory under uncertainty. After completing all 1000 simulation runs, the model computes the average values across all iterations for profit, bid, price, and inventory. These average trajectories are used to generate visualizations of system behavior over time.

The resulting plots illustrate the dynamic nature of the decision environment. For example, bid values typically decrease as the number of remaining periods reduces or as inventory diminishes, indicating less aggressive advertising is needed. Similarly, price strategies reflect a balance between maximizing revenue and ensuring sufficient conversion. Profit trends generally show higher values in earlier periods, when inventory is abundant and aggressive bidding can be justified.

In this study(follow appendix B), the bid, price, and profit values are first determined through a dynamic programming approach before conducting the simulation. The dynamic programming framework optimizes the retailer's decision at each inventory level and time period by solving a maximization problem that accounts for possible consumer behaviors: no click, click without purchase, and click with purchase. Specifically, for each combination of inventory level y and time period T, the optimal bid b and price p are chosen to maximize the expected future profit. This results in three primary matrices: *bid_b*, *price_p*, and *pi*, representing the optimized bid values, price values, and expected cumulative profits, respectively, across all states.

Once these optimal values are computed, the simulation phase is conducted by introducing randomness into consumer responses. In the simulation, the pre computed optimal bids and prices serve as decision rules, and for each time period, random numbers are drawn to simulate whether a consumer clicks on an advertisement and, conditional on a click, whether the consumer proceeds to purchase. Based on these random events, realized profits, bids, prices, and inventory adjustments are recorded in the matrices *bid_b_ans*, *price_p_ans*, *pi_ans*, and *Y_max_ans*. Thus, while dynamic programming provides the theoretically optimal strategy under uncertainty, the simulation incorporates stochastic consumer behavior to observe realized outcomes and variability around the optimal decisions.

A sample simulation with a reduced inventory level and a smaller number of simulation iterations has been conducted to clearly illustrate the detailed workings of the simulation process. Simplifying these parameters allows for easier understanding and transparent explanation of how the inventory, bid, price, and profit evolve under stochastic consumer behavior.

A sample simulation was conducted using 5 iterations over a reduced time horizon T=4 and a maximum inventory level of 3. (Follow the tables below). At Time Period 4, the retailer begins with an inventory level of 3 units. According to the simulation, the optimal bid value selected is 1.19 and the corresponding price is 2.95. The probability of receiving a click, $\lambda(b)$, is 0.58, while the probability of a successful purchase after the click (i.e., $1 - F(p|b)$) is 0.78. A random number is drawn for the click event which is 0.37 and is less than 0.58, indicating that a click occurred. A second random number is also drawn for the purchase event which is 0.65 and falls below 0.78, leading to a purchase. Therefore, a sale occurs, and the retailer earns a profit of 2.95 − 1.19 = 1.76. Inventory decreases by one unit, resulting in 2 units remaining.

At Time Period 3, the inventory remains at 3 units. The optimized bid is 1.11, and the price is 2.95. The click probability is 0.56, and the purchase probability is 0.78. A random number drawn for the click event which is 0.83 and exceeds 0.56, so no click occurs. As a result, no sale happens, no cost is incurred, and the inventory remains unchanged at 2 units. Profit for this time step is 0

At Time Period 2, the system is operating with 2 units of inventory. The bid and price selected are 1.26 and 2.95, respectively. The probability of a click is 0.6, and the probability of purchase is 0.8. The random number for click generation is 0.42 and below 0.6, and the second random number for purchase is 0.23 and also below 0.8 both events succeed. Hence, the retailer realizes a profit of 2.95 − 1.26 = 1.69, and inventory reduces by 1.

At Time Period 1, the inventory is 1 units. The optimal bid and price remain 1.26 and 2.95, respectively. The probability of a click is 0.60, and purchase probability is 0.80. This time, neither a click nor a purchase occurs due to higher random numbers which were 0.79 and 0.98. Thus, the retailer records zero profit, and the inventory remains the same 1 unit by the simulation's end.

This detailed example highlights how the simulation integrates optimized decision-making with stochastic consumer behavior to generate realistic sales paths. It demonstrates how random outcomes in clicks and purchases directly affect realized profits, inventory depletion, and the sequence of price and bid decisions across time. Furthermore, it clarifies how the transition between theoretical optimization (captured in matrices like bid, price) and empirical simulation outcomes (captured in matrices like, Bid after simulation, Price after simulation Profit after simulation, and Inventory after simulation) occurs through a two-stage randomization process first for clicks and then for purchases.

To further support this explanation, the following tables present both the optimized values and the actual outcomes observed during the simulation. They illustrate the dynamic interplay between inventory, consumer behavior, pricing strategy, and profit generation across different time periods and simulation runs

Bid					
Inventory	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
0	0	0	0	0	0
1	0	1.26	0.89	0.68	0.54
2	0	1.26	1.26	1.11	0.96
3	0	1.26	1.26	1.26	1.19

Price					
Inventory	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
0	0	0	0	0	0
1	0	2.95	2.95	3	3.06
2	0	2.95	2.95	2.95	2.95
3	0	2.95	2.95	2.95	2.95

Probability of click					
Inventory	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
0	0	0	0	0	0
1	0	0.6	0.51	0.44	0.39
2	0	0.6	0.6	0.56	0.53
3	0	0.6	0.6	0.6	0.58

Probability of purchase					
Inventory	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
0	0	0	0	0	0
1	0	0.8	0.71	0.64	0.58
2	0	0.8	0.8	0.76	0.73
3	0	0.8	0.8	0.8	0.78

Bid after simulation					
Simulation	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
1	0	1.26	1.26	1.11	1.19
2	0	1.26	1.26	1.26	1.19
3	0	1.26	1.26	1.26	1.19
4	0	1.26	1.26	1.11	1.19
5	0	1.26	1.26	1.26	1.19
6	0	1.26	0.89	1.11	1.19

Profit after simulation					
Simulation	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
1	0	0	1.69	0	1.76
2	0	1.69	1.69	1.69	-1.19
3	0	0	-1.26	1.69	-1.19
4	0	1.69	1.69	0	1.76
5	0	0	1.69	-1.26	0
6	0	1.69	-0.89	1.84	1.76

Price after simulation					
Simulation	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
1	0	2.95	2.95	2.95	2.95
2	0	2.95	2.95	2.95	2.95
3	0	2.95	2.95	2.95	2.95
4	0	2.95	2.95	2.95	2.95
5	0	2.95	2.95	2.95	2.95
6	0	2.95	2.95	2.95	2.95

Inventory after simulation					
Simulation	**Time 0**	**Time 1**	**Time 2**	**Time 3**	**Time 4**
1	1	1	2	2	3
2	0	1	2	3	3
3	2	2	2	3	3
4	0	1	2	2	3
5	2	2	3	3	3
6	0	1	1	2	3

Simulation results									
Time	**Inventory (Before)**	**Probability of bid**	**Probability of purchase**	**Click?**	**Purchase?**	**Profit**	**Inventory (After)**	**Bid after**	**Price after**
4	3	0.58	0.78	Yes	Yes	1.76	2	1.19	2.95
3	2	0.6	0.8	No	No	0	2	1.11	2.95
2	2	0.56	0.76	Yes	Yes	1.69	1	1.26	2.95
1	1	0.6	0.8	No	No	0	1	1.26	2.95

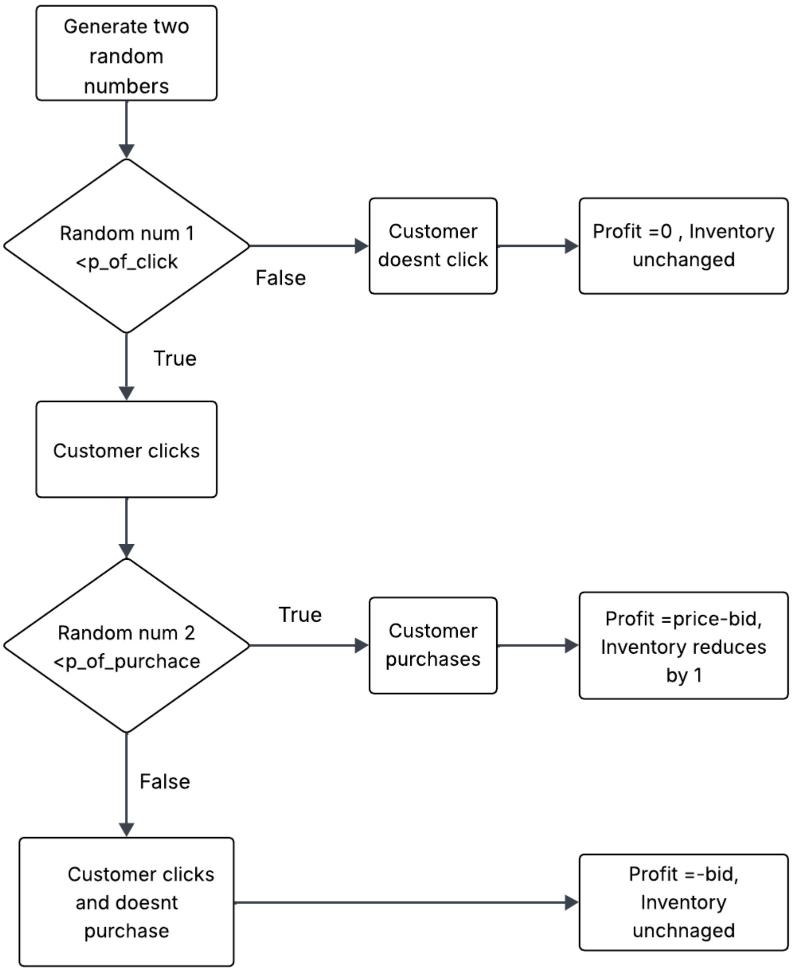

Figure 6 : Simulation Process Flow.

This flowchart visually represents the logic behind the customer decision process in the simulation, but it is not the actual simulation model itself. Through this simulation structure, the model successfully captures the key features of dynamic pricing and advertising strategies in uncertain online marketplaces. It provides a robust foundation for analyzing the impact of strategic decisions on profitability and for exploring alternative policies in the subsequent results and discussion chapter.

4 Results

This chapter presents the results of the simulation study, analyzing how single-mode and multi-mode bidding strategies perform under different consumer reservation price distributions (Gamma, Weibull, Beta). It examines the effects of market uncertainty and inventory constraints on key outcomes such as bid levels, pricing, and profits. The chapter is structured to highlight the evolution of strategies over time, compare results across distributions, and offer practical insights for retail decision-making.

4.1 Gamma Distribution

The Gamma distribution used in this study has a shape parameter of α = 6 and a scale parameter of β = 0.5, producing a distribution that is right-skewed but relatively symmetric. Gamma example 1 as illustrated in figure 7 , is a stable, low-variance strategy designed for consistent performance. In the bid evolution chart, the mean bid follows a gentle downward slope over the 15 time periods, with the 25th and 75th percentiles tightly clustered around it. This narrow spread reflects consistent decision-making across simulations, suggesting the strategy performs reliably even as customer behavior varies.

The price chart echoes this consistency. Prices stay within a narrow range (roughly 2.9 to 3.3), aligning well with the expected reservation prices of the Gamma distribution. This implies that pricing is calibrated to perceived customer value high enough to preserve margin, but balanced to support steady conversions.

The profit curve provides key insights into the variability of outcomes. The mean profit remains consistently positive and declines gradually as inventory is sold off. However, the 25th percentile stays flat at zero throughout, indicating that at least a quarter of simulations failed to generate any profit likely due to no engagement or conversion. The substantial gap between the mean and 75th percentile further highlights that only a portion of runs achieved strong returns, while others underperformed entirely. This suggests a strategy with a skewed performance distribution offering solid upside in some cases, but failing to engage in others

Inventory levels show a smooth and nearly linear decline, with minimal variation between simulations. This steady depletion suggests that bid and price strategies are well-matched to demand, resulting in consistent conversions and efficient inventory turnover.

Gamma example 1 reflects a moderately conservative strategy with selective success. While it generates positive average profits and maintains stable pricing and bidding behavior, the consistent zero-profit outcomes in the lower quartile reveal a vulnerability to non-engagement. This suggests that the strategy performs well under the right conditions but may fail to capture value in a meaningful portion of cases. For retailers, this approach may be appropriate in stable markets where engagement rates are generally high, but it also highlights the need to address scenarios where customer response is weak or delayed.

Gamma example 2 exhibits a distinctly exploratory bidding strategy, characterized by a wide spread between the 25th and 75th percentiles. In several time periods, the 25th percentile drops to zero, indicating that some simulations chose not to bid at all — likely in response to poor expected returns or prior underperformance. In contrast, the 75th percentile climbs above 1.2 in multiple periods, reflecting aggressive bidding efforts to capture demand. The mean bid fluctuates throughout the horizon, underscoring the strategy's reactive nature. From a retailer's perspective, this approach reflects a willingness to experiment and capitalize on emerging opportunities, but it also introduces volatility and execution risk. The model effectively balances between maximizing visibility when conditions are favorable and conserving resources when engagement appears unlikely.

In contrast, the pricing strategy remains relatively stable. The 25th and 75th percentiles stay close to the mean throughout, indicating consistent price setting with only minor adjustments in response to market dynamics.

Profit outcomes, however, show greater dispersion. The mean remains positive,occasionally outperforming Example 1 yet the 25th percentile stays at zero across all periods. This suggests that at least a quarter of simulations saw no engagement, resulting in no revenue or loss. These flat outcomes are more likely a result of conservative bids failing to attract attention, rather than poor conversions.

Inventory levels further underscore the variability in strategic outcomes. The wider spread between the 25th and 75th percentiles , with higher inventory retention in the upper quartile suggests that in some simulations, products remained unsold due to limited engagement or weak conversion. In contrast, the lower quartile shows faster inventory depletion, reflecting more effective customer interactions and higher conversion rates.

Overall, Gamma example 2 illustrates a higher-variance outcome profile. While the average performance remains stable, a significant portion of simulation runs yield no return — reflecting missed opportunities rather than operational inefficiency. In at least 25% of simulations, the model generated no revenue, likely due to a lack of customer engagement (e.g., no advertising clicks). In these cases, there were no advertising costs, but also no sales — resulting in zero profit rather than an actual loss. This outcome highlights a cautious bidding response in low-engagement scenarios, where the strategy conserves budget but sacrifices potential reach. Such an approach may be more appropriate for dynamic, high-risk environments, where the possibility of high returns can justify increased variability in performance.

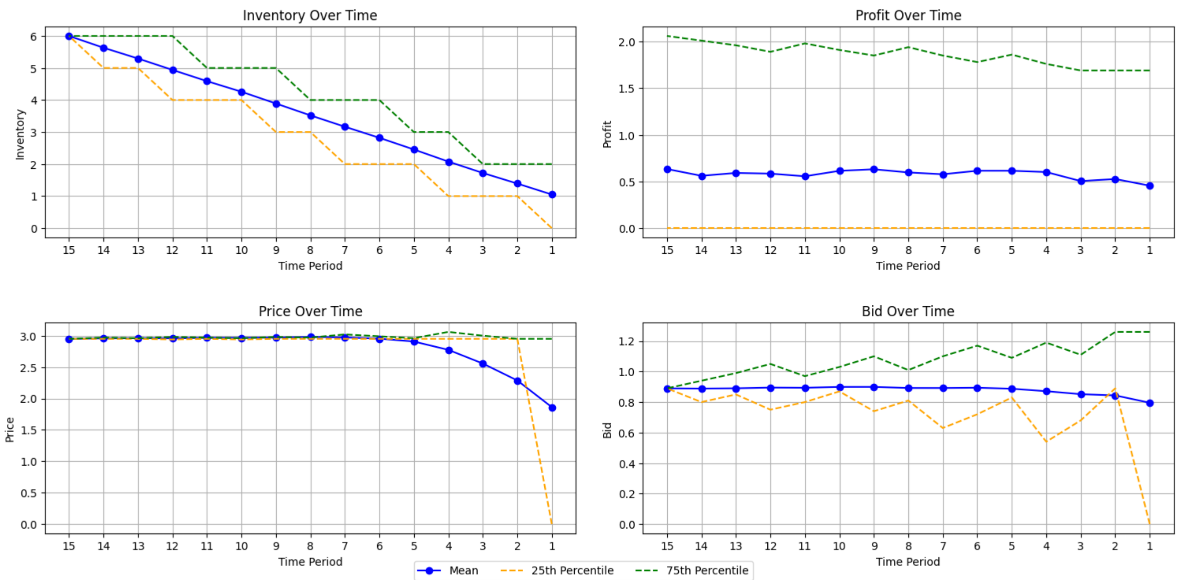

Figure 7 : Gamma Distribution — Example 1 Strategy.

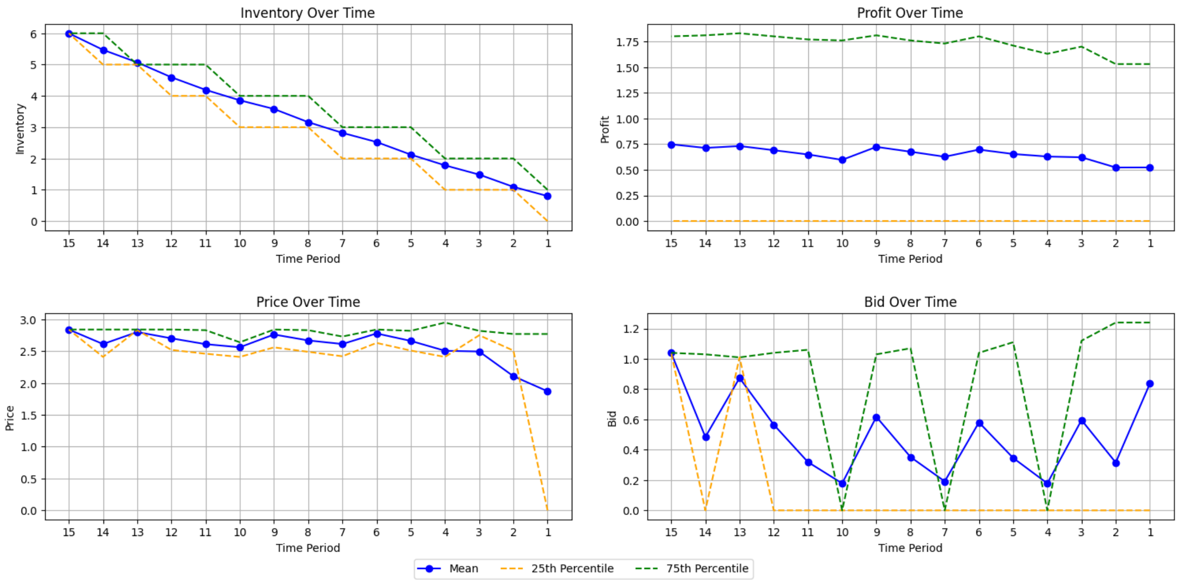

Figure 8 : Gamma Distribution — Example 2 Strategy.

4.2 Weibull Distribution

The Weibull distribution (α = 2, β = 3.35) introduces positive skewness. Example 1 in Weibull distrubution demonstrates a steady, conservative bidding strategy. The mean bid remains stable around 0.85, with only modest variation between the 25th and 75th percentiles. This tight band suggests that the retailer followed a controlled bidding strategy across simulations, consistently applying bids that balanced cost-efficiency with market visibility. For customers, this means advertising visibility is predictable, and for the retailer, the bidding strategy prioritizes cost control and steady engagement.

Pricing follows a similarly disciplined pattern. Prices remain tightly grouped in the 3.1 to 2.9 range across all periods and percentiles. This consistency implies that the model quickly settled on a price that matches the central mass of the Weibull reservation price curve. From a retailer's standpoint, the consistent pricing suggests a disciplined, value-based strategy designed to align closely with customer willingness to pay while preserving pricing power. For customers, this stability ensures a fair and expected price point, which can support long-term trust and purchase intent.

Profit performance is more mixed. While the mean profit remains positive, the 25th percentile is flat at zero across all time periods. This indicates that in at least a quarter of simulations, no engagement, no clicks, no revenue, and no cost occurred. These represent missed opportunities rather than losses. On the other hand, the 75th percentile reaches profit levels above 2.0 midway through, showing that when bidding and pricing align well with high-WTP customers, significant gains are achievable.

Inventory levels support this interpretation of steady but cautious execution. Mean inventory depletes gradually, with both percentile bands staying close to the mean. Slightly slower depletion in the upper quartile suggests that in some runs, customers weren't reached or converted efficiently. But overall, the tight clustering implies uniform customer interaction and conversion behavior.

In summary, Weibull example 1 reflects a risk-averse retail strategy tailored to a market with a few high-value customers and many average ones. By maintaining consistent bids and prices, the retailer avoids overspending and underpricing, resulting in stable performance across most scenarios. While this strategy may sacrifice some upside, it's well-suited to predictable environments where long-term reliability is more valuable than short-term wins.

Weibull example 2 illustrates a higher-variance and more exploratory bidding strategy compared to Example 1. The bid evolution graph shows a pronounced spread: the 25th percentile drops to zero in several time periods, while the 75th percentile consistently exceeds 1.0. This indicates that in a significant number of simulations, the model opted not to bid , likely in response to poor expected conversion .Whereas in other runs, it pursued aggressive bidding to capture demand. The mean bid also fluctuates more across the 15 time periods, suggesting that the model dynamically adjusted its bidding strategy in response to performance feedback or shifting demand signals.

Price dynamics in this example are similarly variable. The 25th percentile dips below 2.6 in several mid-to-late periods, suggesting that in some simulations, the model aggressively reduced prices, likely to stimulate conversions or address slow inventory turnover. In contrast, the 75th percentile holds steady around 2.9–3.0, indicating that other simulations maintained premium pricing. This wide spread suggests a dual strategy space: some runs opted for discount-driven tactics, while others pursued margin preservation. From the customer perspective, this could translate to inconsistent pricing experiences, depending on the simulation scenario.

The profit curve clearly reflects this performance volatility. While the mean profit remains positive throughout, the 25th percentile falls below zero in multiple periods, indicating that a quarter of the simulations incurred actual losses. These losses are likely the result of advertising spending (e.g., clicks) without resulting purchases which is a costly outcome for the retailer. This highlights the risk of mismatch between bid aggressiveness and true customer willingness to pay. However, the 75th percentile exceeds 1.5 throughout and quite stable, confirming that under favorable conditions, the model can yield strong profits.

Inventory behavior reinforces this interpretation. The 25th percentile shows faster depletion, likely driven by aggressive bidding or price reductions that resulted in higher conversions. In contrast, the 75th percentile retains more inventory, suggesting that in some simulations, lower bidding activity or misaligned pricing led to underperformance and unsold stock. This divergence in inventory usage reflects inconsistent market engagement across runs.

In summary, Weibull example 2 represents a more opportunistic, higher-risk strategy. It explores a broad space of bidding and pricing decisions, which can deliver strong results when conditions align but also carries the risk of financial loss. This strategy may be appropriate in dynamic or uncertain market environments where the potential upside justifies variability and execution risk. However, for retailers prioritizing predictable outcomes and consistent engagement, the more disciplined approach seen in Weibull Example 1 would be a better fit.

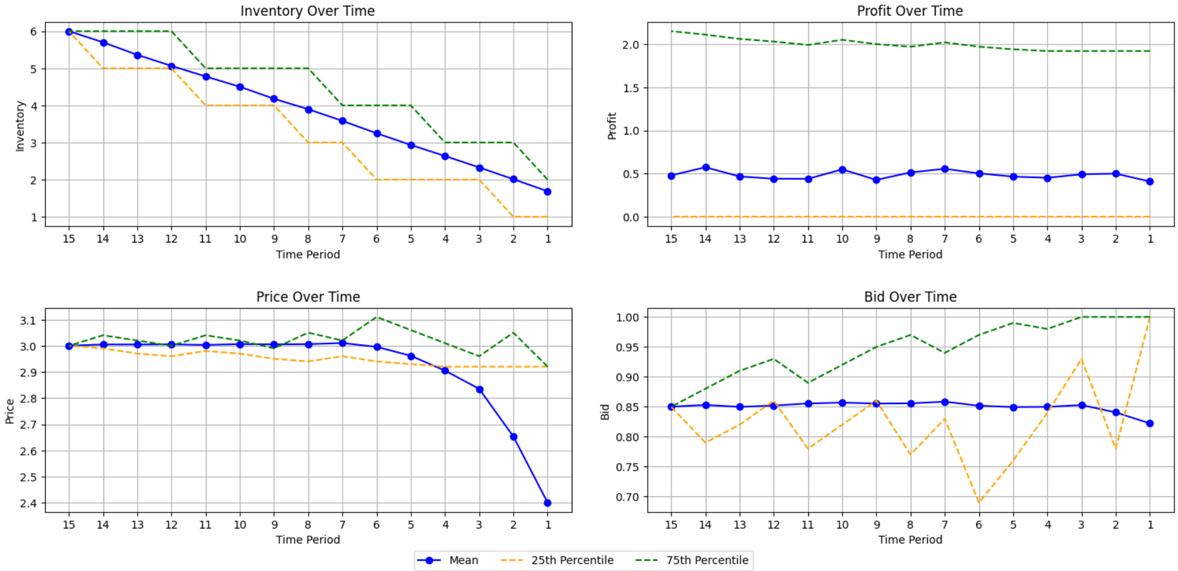

Figure 9 : Weibull Distribution – Example 1 Strategy

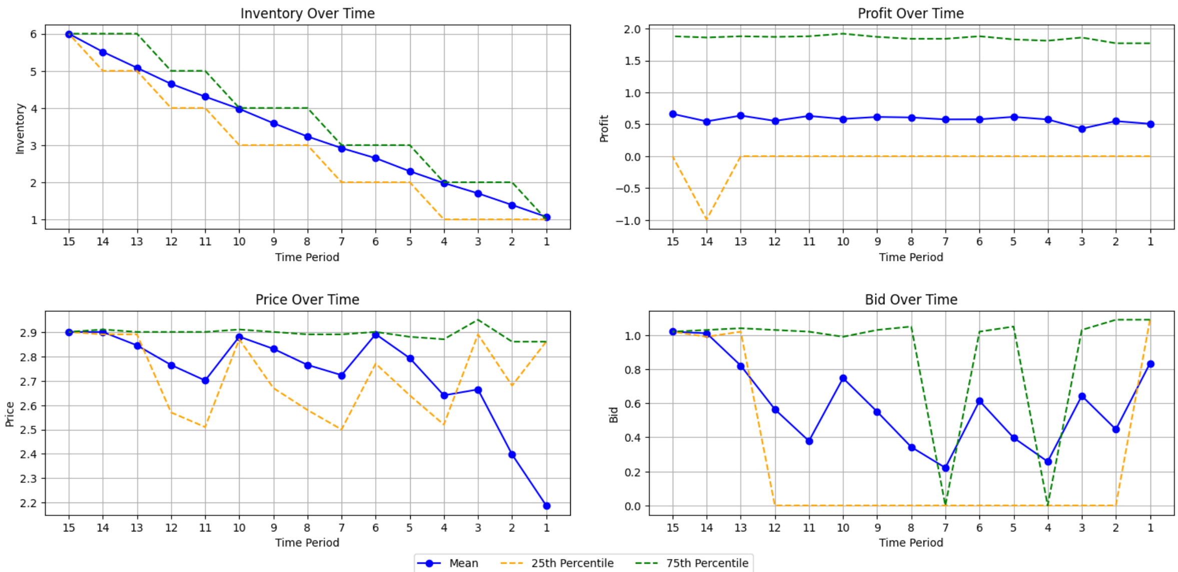

Figure 10 : Weibull Distribution – Example 2 Strategy

4.3 Beta Distribution.

The Beta distribution ($\alpha = 3$, $\beta = 7$), scaled to [0, 10], is technically right-skewed but concentrated in the lower half of the price range. This reflects a market where most consumers have relatively modest willingness to pay. For retailers, this demands a pricing and bidding strategy that captures low-to-moderate value buyers efficiently, without overextending costs or overpricing inventory.

The bid evolution graph in Beta example 1 demonstrates a moderately consistent strategy with some adaptive variation. The mean bid remains steady around 0.85 across all time periods, reflecting a baseline preference for balanced engagement. While the 25th and 75th percentiles generally track within a narrow range, the spread begins to widen slightly in the later stages, suggesting that some simulations adapted bids upward or downward in response to market conditions. This behavior indicates a model that maintains cost control while allowing limited flexibility to optimize visibility. For the retailer, this approach balances efficiency with reliability, ensuring stable exposure without excessive risk or spending..

Pricing behavior is similarly restrained. Prices hover around 3.0 with very limited fluctuation across time or simulations. The tight percentile bands indicate that the strategy consistently settled on a value-based pricing policy, likely aligned with the peak of the Beta-distributed customer reservation price curve. Consumers encounter consistent pricing, which may improve trust and purchase likelihood, while retailers benefit from margin stability.

Profit outcomes confirm the effectiveness of this balanced approach. The mean profit remains positive throughout the horizon, and the 25th percentile never drops to zero — indicating that every simulation yielded at least some return. The narrow spread across percentiles reflects low volatility, signaling that the model reliably identifies and serves its target segment without taking risky bets.

The inventory graph further supports this stability. The rate of inventory depletion is smooth, with minimal divergence between percentile bands. This uniformity suggests that bid and price decisions generalized well across simulated consumer behaviors, ensuring consistent sell-through regardless of individual variation.

In summary, Beta example 1 showcases a stable and reliable bidding-pricing strategy, ideal for environments where customer preferences are tightly clustered and relatively predictable. The retailer's approach prioritizes consistency and cost-efficiency over aggressive experimentation, resulting in dependable profits and smooth operations and qualities that are highly desirable in risk-averse retail settings.

Beta example 2 reveals a broader and more variable strategic profile compared to Example 1. The bid evolution graph shows a pronounced spread between the 25th and 75th percentiles, particularly in the mid-to-late periods. While some simulations adopted aggressive bidding (with 75th percentile values near or above 1.0), others bid minimally or not at all, as indicated by the 25th percentile reaching zero. This reflects the model's adaptive, exploratory behavior testing both conservative and high-engagement strategies based on prior performance signals. For the retailer, this represents a willingness to take calculated risks in search of optimal engagement.

Pricing dynamics mirror this strategic divergence. The mean price fluctuates more than in Beta Example 1, and the 25th percentile dips significantly in several periods, indicating that some simulations relied on price cuts to stimulate demand. Meanwhile, the 75th percentile remained consistently higher, suggesting that other simulations held firm on pricing to preserve margins. This split reflects a dual strategy: some runs pursued volume through discounting, while others sought value capture from customers with higher willingness to pay.

The profit trajectory reflects this variability. While the mean profit remains positive throughout, the 25th percentile hovers at or near zero across multiple periods. This implies that in at least one-quarter of the simulations, no profit was achieved likely due to a lack of advertising engagement, where no clicks resulted in neither cost nor revenue. These scenarios represent missed opportunities rather than financial losses. In contrast, the upper quartile continues to show strong profit performance, highlighting the model’s potential when bid-price strategies align effectively with market conditions.

he inventory behavior in Beta Example 2 is relatively consistent across simulations. The 25th and 75th percentiles remain fairly close to the mean for most of the time horizon, indicating that inventory depletion followed a similar pattern in the majority of runs. In contrast to Example 1, where the percentile bands were more widely separated, this outcome suggests that the model maintained tighter control over inventory consumption. For the retailer, this reflects a more uniform execution of the strategy, which reduces uncertainty but may limit responsiveness to extreme market signals..

In summary, Beta example 2 reflects a more flexible, exploratory approach that adjusts aggressively to market feedback. It offers upside potential in favorable conditions but carries the risk of underperformance in less responsive scenarios. While this strategy may be well-suited to dynamic markets where experimentation is encouraged, it may be less appropriate in stable environments where predictable performance and cost control are priorities.

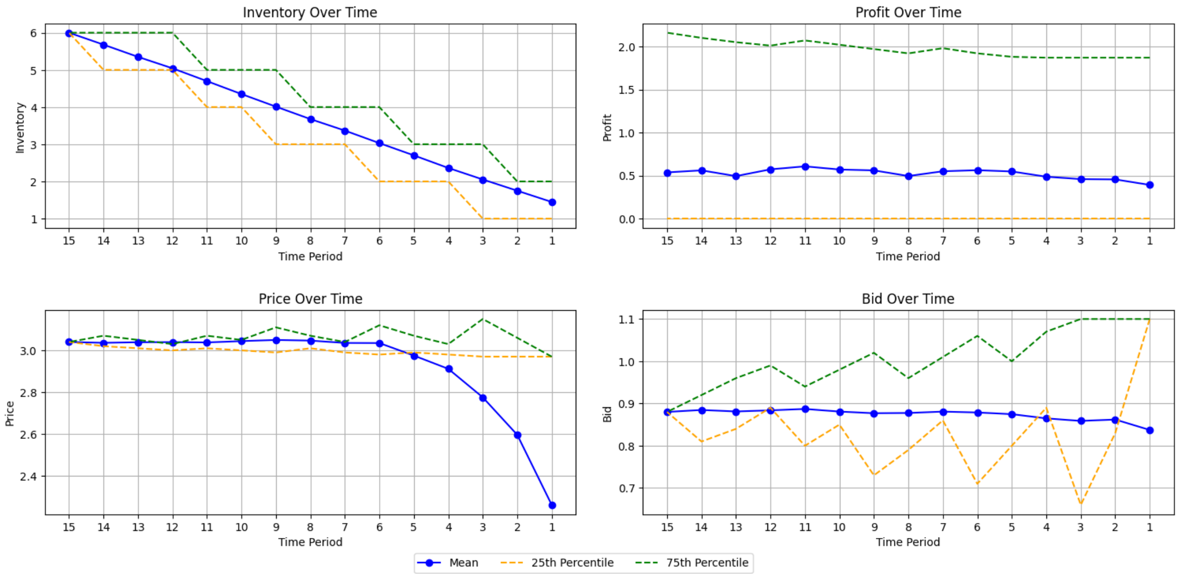

Figure 11 : Beta Distribution — Example 1 Strategy

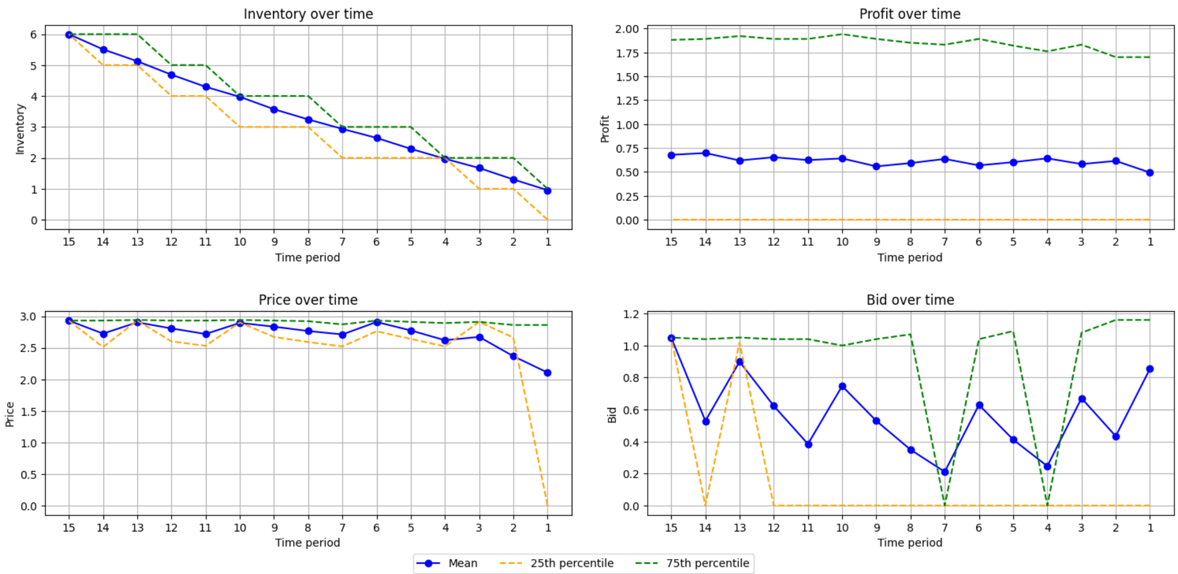

Figure 12 : Beta Distribution — Example 2 Strategy

Overall, the simulation results demonstrate that bidding and pricing strategies must be carefully tailored to the characteristics of consumer demand, particularly as represented by different reservation price distributions. Across the Gamma, Weibull, and Beta scenarios, a consistent trade-off emerges between strategic consistency and exploratory adaptation.

Strategies such as Gamma Example 1, Weibull Example 1, and Beta Example 1 prioritized stability, exhibiting tightly clustered bid, price, and profit outcomes. These approaches delivered reliable returns, steady inventory depletion, and minimal downside risk making them well-suited for markets where customer preferences are predictable and performance consistency is valued.

In contrast, the more exploratory strategies seen in Example 2 of Gamma, Weibull, and Beta displayed significantly greater variability. These models engaged in dynamic adjustment, with a wider spread in bid and price decisions across simulations. While this broadened the opportunity for high profits in favorable conditions, it also introduced execution risk including non-engagement, profit stagnation, and uneven inventory consumption.

From a retailer’s perspective, these findings reinforce that no single strategy is universally optimal. The ideal approach depends on the volatility of the target market, the shape of the consumer reservation price distribution, and the organization’s tolerance for risk. Stable strategies are effective in well-understood environments, while more adaptive, higher-variance approaches may be advantageous in competitive or uncertain contexts, where potential upside can justify increased variability.

5 Conclusion and Outlook

This thesis examined how dynamic bidding and pricing strategies perform under uncertainty in sponsored search environments. Using a simulation-based approach inspired by the model Ye et al. (2015) , the study compared single-mode and multi-mode strategies across Gamma, Weibull, and Beta consumer valuation distributions. Each distribution was calibrated to have the same mean, allowing for controlled comparisons of strategic behavior under different demand shapes. The analysis focused on the interaction between demand heterogeneity, inventory constraints, and the evolution of pricing and bidding decisions over time.

The results demonstrate that strategy performance is highly sensitive to the distributional characteristics of the market. In relatively predictable environments such as those modeled by the Gamma distribution, the single-mode strategy produced steady profits and smooth inventory turnover. This reflects a risk-conscious approach that avoids excessive variation and is well suited to retailers who prioritize stability and budget control. In contrast, the multi-mode strategy adapted more aggressively, especially under skewed demand settings like the Weibull and Beta distributions. These scenarios showed that while exploratory bidding and pricing can uncover high-value opportunities, they also introduce greater outcome variability and a higher incidence of non-engagement, leading to missed revenue.

From a retailer's perspective, an important insight is that two markets with the same average customer willingness to pay may behave very differently. It is not enough to consider the average value alone. What matters is how customer value is distributed. In some markets, most consumers may cluster around a low price point, while in others there may be a wider spread that includes a few high-value buyers. Strategies that ignore this structure risk misallocating advertising spend or setting prices that fail to convert. Aligning strategy with the shape of the demand distribution allows retailers to better match offers to customers, reduce inefficiencies, and improve overall return on investment.

The simulation framework presented in this thesis offers practical value as a decision-support tool. By visualizing the range of outcomes for each strategy under varying market conditions, retailers can gain a clearer understanding of the trade-offs between consistency and flexibility. This enables more informed decision-making and supports the development of strategies that are responsive to both customer behavior and operational constraints.

Looking ahead, this research can be extended in several valuable ways. First, using real-world data such as transaction records or browsing behavior could improve the model's accuracy and allow for better validation. Second, the current model assumes only one seller. Adding competitors would make the simulation more realistic and help explore how businesses compete for advertisements and customer attention. Third, the strategies used in this thesis remain fixed over time. Future work could explore smarter strategies that learn and adapt using methods like machine learning. Finally, adding budget constraints would help simulate the financial limits retailers face in real digital advertising campaigns.

In summary, this thesis contributes a flexible and insightful simulation framework for analyzing how consumer valuation distributions affect dynamic pricing and bidding strategies in digital advertising. It provides both theoretical insights and practical tools for understanding strategy performance under uncertainty. As online retail continues to grow in complexity and competition, the ability to model and test strategic responses in a controlled environment will be essential for effective campaign design. This work lays a foundation for more adaptive and data-informed advertising strategies in the future.

6 References

Agrawal, S., Feng, Y., & Tang, W. (2023). Dynamic Pricing and Advertising with Demand Learning.

BuzzBoard. (2024). *Figuring Out Bidding Strategies for Search Engine Marketing Campaigns*. https://www.buzzboard.ai/figuring-out-bidding-strategies-for-search-engine-marketing-campaigns/#:~:text=To%20achieve%20SEM%20bidding%20success,of%20driving%20conversions%20and%20revenue.

Cuemath. (n.d). *Probability Density Function*. https://www.cuemath.com/data/probability-density-function/

Dadge, M., Ashish, K., Ashu, Y., Ayush, Kumar, Pal, & Ashwin, S. (2014). Optimization of Inventory Management Using Mordern Techniques.

DeCross, M., & Liou, L. (n.d). *Continuous Random Variables - Cumulative Distribution Function*. https://brilliant.org/wiki/continuous-random-variables-cumulative/

Everitt, & Skrondal. (2006). *The Cambridge Dictionary of Statistics* (4th edition).

Fader, P. S., & Hardie, B. G. S. (2005). Applied Probability Models in Marketing Research: Introduction.

Fain, D. C., & Pedersen, J. O. (2006). Sponsored search: A brief history. *Bulletin of the American Society for Information Science and Technology*, *32*(2), 12–13. https://doi.org/10.1002/bult.1720320206

Ferreira, K. J., Lee, B. H. A., & Simchi-Levi, D. (2016). Analytics for an Online Retailer: Demand Forecasting and Price Optimization. *Manufacturing & Service Operations Management*, *18*(1), 69–88. https://doi.org/10.1287/msom.2015.0561

Gallego, G., & van Ryzin, G. (1994). Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons.

H, A., CN, A., & J, F. (2023). Investigating the parameters of the beta distribution. *World Journal of Advanced Research and Reviews*, *19*(1), 815–830. https://doi.org/10.30574/wjarr.2023.19.1.1409

Irvine, M. (2024). *Google Ads Benchmarks for YOUR Industry*. https://www.wordstream.com/blog/ws/2016/02/29/google-adwords-industry-benchmarks

Jansen, B. J., Liu, Z., & Simon, Z. (2013). The effect of ad rank on the performance of keyword advertising campaigns. *Journal of the American Society for Information Science and Technology*, *64*(10), 2115–2132. https://doi.org/10.1002/asi.22910

Klipfolio. (2025). *SEO Click-Through Rate (CTR)*. https://www.klipfolio.com/resources/kpi-examples/seo/click-through-rate#:~:text=SEO%20Click%2DThrough%20Rate%20(CTR)%20Key%20Terms,the%20number%20of%20goals%20completed.

Kopalle, P. K., Pauwels, K., Akella, L. Y., & Gangwar, M. (2023). Dynamic pricing: Definition, implications for managers, and future research directions. *Journal of Retailing*, *99*(4), 580–593. https://doi.org/10.1016/j.jretai.2023.11.003

Lai, C.-D., Murthy, D. N., & Xie, M. (2006). Weibull Distributions and Their Applications, 63–78. https://doi.org/10.1007/978-1-84628-288-1_3

LeadsView. (2024). *Mastering Bid Management in SEM: Tactics for Budget Optimization*. https://www.linkedin.com/pulse/mastering-bid-management-sem-tactics-budget-optimization-leadsview-6vovc/

Luko, S. N. (1999). A Review of the Weibull Distribution and Selected Engineering Applications. Advance online publication. https://doi.org/10.4271/1999-01-2859

Pinelis, Y. K. (2018). Tutorial: Parametric Reliability Models.

Salehi, S., & Mirmohammadi, S. H. (2023). A solution approach for sponsored search advertising and dynamic pricing for a perishable product and an online retailer with budget constraint. *Computers & Industrial Engineering*, *177*, 109086. https://doi.org/10.1016/j.cie.2023.109086

Schwabe, D. (2013). Proceedings of the 22nd international conference on World Wide Web. Advance online publication. https://doi.org/10.1145/2488388

Smith, J., Sanchez, M., & Rossi, G. (2024). Management Strategies and Engineering Sciences.

Sugiyama, M. (2016). Examples of Continuous Probability Distributions, 37–50. https://doi.org/10.1016/B978-0-12-802121-7.00015-7

Tunuguntla, V., Basu, P., Rakshit, K., & Ghosh, D. (2019). Sponsored search advertising and dynamic pricing for perishable products under inventory-linked customer willingness to pay. *European Journal of Operational Research*, *276*(1), 119–132. https://doi.org/10.1016/j.ejor.2018.12.026

U Eric, Olusola Oti Michael O., & C, F. (2021). A Study of Properties and Applications of Gamma Distribution.

Wang, T., Venkatesh, R., & Chatterjee, R. (2007). Reservation Price as a Range: An Incentive-Compatible Measurement Approach. *Journal of Marketing Research*, *44*(2), 200–213. https://doi.org/10.1509/jmkr.44.2.200

Ye, S., Aydin, G., & Hu, S. (2015). Sponsored Search Marketing: Dynamic Pricing and Advertising for an Online Retailer. *Management Science*, *61*(6), 1255–1274. https://doi.org/10.1287/mnsc.2014.1915

7 Appendix

This appendix contains the python code used to implement the simulation model and bidding strategies discussed in the thesis. It includes the full logic for dynamic programming, distribution setup, and result visualization.

7.1 Appendix A : Replication of Theoretical Model from (Ye et al., 2015).

Optimization with SciPy's optimize Module Example1

```
import numpy as np
import scipy.stats as stats
from scipy.optimize import minimize
import matplotlib.pyplot as plt

# ------------------ Parameters ------------------
alpha_X = 6
beta_X = 0.5
T = 15
Y_max = 6

# ------------------ Model Functions (Example 1) ------------------
def mu_b(b):
    return 0.6 * b

def lambda_b(b):
    return 0.8 - 0.7 * np.exp(-b)

def F_p_given_b(p, b):
    z = (p - mu_b(b)) / (1 + 0.1 * b)
    return stats.gamma.cdf(z, a=6, scale=0.5)



# ------------------ Initialization ------------------
pi = np.zeros((Y_max + 1, T + 1))       # π(y, t): expected profit
b_star = np.zeros((Y_max + 1, T + 1))   # Store optimal bids

# Enforce terminal condition explicitly: π(y, 0) = 0
pi[:, 0] = 0

# ------------------ Dynamic Programming ------------------
for t in range(1, T + 1):
    for y in range(1, Y_max + 1):
        def objective(x):
            b, p = x
            f = F_p_given_b(p, b)
            term1 = (1 - lambda_b(b)) * pi[y, t - 1]
            term2 = lambda_b(b) * f * (pi[y, t - 1] - b)
            term3 = lambda_b(b) * (1 - f) * (p + pi[y - 1, t - 1] - b)
            return -(term1 + term2 + term3)

        result = minimize(
    objective,
    x0=[1, 1],  # tuned initial guess
    bounds=[(0.01, None), (0.01, None)],
    method='L-BFGS-B',
    options={'maxiter': 100, 'ftol': 1e-9}
```

```
)

        pi[y, t] = -result.fun
        b_star[y, t] = result.x[0]

# ------------------ Plotting ------------------
plt.figure(figsize=(10, 6))
for y in [2, 4, 6]:
    plt.plot(range(T, 0, -1), b_star[y, T:0:-1], label=f'Inventory Level =
{y}')

plt.title('Example 1 \nOptimal Bidding Strategy')
plt.xlabel('Number of remaining periods')
plt.ylabel('Optimal Bid')
plt.legend()
plt.grid(True)
plt.xticks(range(15, 0, -1))
plt.gca().invert_xaxis()
plt.show()
```

Optimization with SciPy's optimize Module Example 2.

```
import numpy as np
import scipy.stats as stats
from scipy.optimize import minimize
import matplotlib.pyplot as plt

# ------------------ Parameters (from paper) ------------------
alpha_X = 6        # Gamma shape parameter
beta_X = 0.5       # Gamma scale parameter
T = 15             # Number of time periods
Y_max = 6          # Maximum inventory level

# ------------------ Functions for Example 2 ------------------
def mu_b(b):
    return 0.4 * b  # Paper: Example 2

def lambda_b(b):
    return (1 + 0.4 * np.exp(4 - 4 * b)) / (1 + np.exp(4 - 4 * b))  #
Paper: Example 2

def sigma_b(b):
    return 1 + 0.1 * b  # Shared across both examples

def F_p_given_b(p, b):
    z = (p - mu_b(b)) / sigma_b(b)
    z = np.maximum(z, 0)  # avoid invalid values
    return stats.gamma.cdf(z, a=alpha_X, scale=beta_X)



# ------------------ Initialization ------------------
pi = np.zeros((Y_max + 1, T + 1))      # π(y, t): expected profit
b_star = np.zeros((Y_max + 1, T + 1))  # Store optimal bids

# ------------------ Terminal Condition (from paper) ------------------
pi[:, 0] = 0  # Inventory has zero value at time 0

# ------------------ Dynamic Programming ------------------
for t in range(1, T + 1):
```

```
    for y in range(1, Y_max + 1):
        def objective(x):
            b, p = x[0], x[1]
            f = F_p_given_b(p, b)
            term1 = (1 - lambda_b(b)) * pi[y, t - 1]
            term2 = lambda_b(b) * f * (pi[y, t - 1] - b)
            term3 = lambda_b(b) * (1 - f) * (p + pi[y - 1, t - 1] - b)
            return -(term1 + term2 + term3)

        result = minimize(
            objective,
            x0=[1, 2],  # Stable initial guess
            bounds=[(0.01, None), (0.01, None)],
            method='L-BFGS-B'
        )

        pi[y, t] = -result.fun
        b_star[y, t] = result.x[0]  # Store optimal bid

# ------------------ Plotting ------------------
plt.figure(figsize=(10, 6))
for y in [2, 4, 6]:
    plt.plot(range(T, 0, -1), b_star[y, T:0:-1], label=f'Inventory Level =
{y}')


plt.title('Example 2 \nOptimal Bidding Strategy')
plt.xlabel('Number of remaining periods')
plt.ylabel('Optimal Bid')
plt.legend()
plt.grid(True)
plt.xticks(range(15, 0, -1))
plt.gca().invert_xaxis()
plt.show()
```

Optimization with Grid Search Example1

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import gamma

# ------------------ Parameters ------------------
alpha_X = 6
beta_X = 0.5
T = 15
Y_max = 6

# ------------------ Model Functions (Example 1) ------------------
def mu_b(b):
    return 0.6 * b

def sigma_b(b):
    return 1 + 0.1 * b

def lambda_b(b):
    return 0.8 - 0.7 * np.exp(-b)

def F_p_given_b(p, b):
    z = (p - mu_b(b)) / sigma_b(b)
    return gamma.cdf(z, a=alpha_X, scale=beta_X)

# ------------------ Grid Search Implementation ------------------
```

```
# Define search grid for bid (b) and price (p)
bid_grid = np.linspace(0.01, 2.0, 100)
price_grid = np.linspace(0.01, 4.0, 100)

# Initialize profit and bid matrices
pi = np.zeros((Y_max + 1, T + 1))
b_star = np.zeros((Y_max + 1, T + 1))

# Terminal condition
pi[:, 0] = 0

# Dynamic programming with grid search
for t in range(1, T + 1):
    for y in range(1, Y_max + 1):
        best_profit = -np.inf
        best_b = 0
        for b in bid_grid:
            for p in price_grid:
                f = F_p_given_b(p, b)
                term1 = (1 - lambda_b(b)) * pi[y, t - 1]
                term2 = lambda_b(b)  * f * (pi[y, t - 1] - b)
                term3 = lambda_b(b)  * (1 - f)  * (p + pi[y - 1, t - 1] - b)
                profit = term1 + term2 + term3
                if profit > best_profit:
                    best_profit = profit
                    best_b = b
        pi[y, t] = best_profit
        b_star[y, t] = best_b

# ------------------ Plotting ------------------
plt.figure(figsize=(10, 6))
for y in [2, 4, 6]:
    plt.plot(range(T, 0, -1), b_star[y, T:0:-1], label=f'Inventory Level =
{y}')

plt.title('Example 1 \nOptimal Bidding Strategy(Grid search)')
plt.xlabel('Number of remaining periods')
plt.ylabel('Optimal Bid')
plt.legend()
plt.grid(True)
plt.xticks(range(15, 0, -1))
plt.gca().invert_xaxis()
plt.show()
```

Optimization with Grid Search Example 2

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import gamma

# ------------------- Parameters (Example 2) -------------------
alpha_X = 6
beta_X = 0.5
T = 15
Y_max = 6

# ------------------ Model Functions (Example 2) ------------------
def mu_b(b):
    return 0.4 * b

def sigma_b(b):
    return 1 + 0.1 * b
```

```
def lambda_b(b):
    return (1 + 0.4 * np.exp(4 - 4 * b)) / (1 + np.exp(4 - 4 * b))

def F_p_given_b(p, b):
    z = (p - mu_b(b)) / sigma_b(b)
    z = np.maximum(z, 0)  # Ensure input is non-negative
    return gamma.cdf(z, a=alpha_X, scale=beta_X)

# ------------------ Grid Search Implementation ------------------
bid_grid = np.linspace(0.01, 1.5, 150)     # more fine-grained bids
price_grid = np.linspace(0.5, 3.5, 200)    # tighter, realistic price band


pi = np.zeros((Y_max + 1, T + 1))
b_star = np.zeros((Y_max + 1, T + 1))

pi[:, 0] = 0  # Terminal condition

for t in range(1, T + 1):
    for y in range(1, Y_max + 1):
        best_profit = -np.inf
        best_b = 2
        for b in bid_grid:
            for p in price_grid:
                f = F_p_given_b(p, b)
                term1 = (1 - lambda_b(b)) * pi[y, t - 1]
                term2 = lambda_b(b) * f * (pi[y, t - 1] - b)
                term3 = lambda_b(b) * (1 - f) * (p + pi[y - 1, t - 1] - b)
                profit = term1 + term2 + term3
                if profit > best_profit:
                    best_profit = profit
                    best_b = b
        pi[y, t] = best_profit
        b_star[y, t] = best_b

# ------------------ Plotting ------------------
plt.figure(figsize=(10, 6))
for y in [2, 4, 6]:
    plt.plot(range(T, 0, -1), b_star[y, T:0:-1], label=f'Inventory Level = 
{y}')

plt.title('Example 2 \nOptimal Bidding Strategy(Grid search)')
plt.xlabel('Number of remaining periods')
plt.ylabel('Optimal Bid')
plt.legend()
plt.grid(True)
plt.xticks(range(15, 0, -1))
plt.gca().invert_xaxis()
plt.show()
```

7.2 Appendix B : Python Code for Simulation Implementation.

Simulation Using the Gamma Distribution Example 1

```
import numpy as np
import scipy.stats as stats
from scipy.optimize import minimize
import matplotlib.pyplot as plt
```

```
# Parameters
alpha_X = 6
beta_X = 0.5
T = 15
Y_max = 6
n_sim = 1000

# Simulation result arrays
pi_ans = np.zeros((n_sim + 1, T + 1))
price_p_ans = np.zeros((n_sim + 1, T + 1))
bid_b_ans = np.zeros((n_sim + 1, T + 1))
Y_max_ans = np.zeros((n_sim + 1, T + 1))

# Functions
def mu_b(b):
    return 0.6 * b

def lambda_b(b):
    return 0.8 - 0.7 * np.exp(-b)

def sigma_b(b):
    return 1 + 0.1 * b

def F_p_given_b(p, b):
    z = (p - mu_b(b)) / sigma_b(b)
    z = np.maximum(z, 0)
    return stats.gamma.cdf(z, a=alpha_X, scale=beta_X)

# Matrices for dynamic programming
pi = np.zeros((Y_max + 1, T + 1))
bid_b = np.zeros((Y_max + 1, T + 1))
price_p = np.zeros((Y_max + 1, T + 1))
p_of_purchase = np.zeros((Y_max + 1, T + 1))
p_of_click = np.zeros((Y_max + 1, T + 1))

# Backward induction
for t in range(1, T + 1):
    for y in range(1, Y_max + 1):
        def objective(x):
            b, p = x[0], x[1]
            term1 = (1 - lambda_b(b)) * pi[y, t - 1]
            term2 = lambda_b(b) * F_p_given_b(p, b) * (pi[y, t - 1] - b)
            term3 = lambda_b(b) * (1 - F_p_given_b(p, b)) * (p + pi[y - 1,
t - 1] - b)
            return -(term1 + term2 + term3)

        result = minimize(objective, x0=[1, 1], bounds=[(0, None), (0,
None)])
        pi[y, t] = round(-result.fun, 2)
        bid_b[y, t] = round(result.x[0], 2)
        price_p[y, t] = round(result.x[1], 2)

# Probabilities
for row in range(1, Y_max + 1):
    for col in range(1, T + 1):
        p_of_purchase[row, col] = 1 - F_p_given_b(price_p[row, col],
bid_b[row, col])
        p_of_click[row, col] = lambda_b(bid_b[row, col])

# Simulation
for iter in range(n_sim + 1):
    rows, cols = p_of_purchase.shape
```

```
    price_p_ans[iter, -1] = price_p[rows - 1, cols - 1]
    bid_b_ans[iter, -1] = bid_b[rows - 1, cols - 1]
    Y_max_ans[iter, -1] = Y_max
    loop_exited = False

    for i in range(rows - 1, -1, -1):
        for j in range(cols - 1, -1, -1):
            r_click = np.random.rand()
            r_purchase = np.random.rand()

            if r_click < p_of_click[i, j]:
                if r_purchase < p_of_purchase[i, j]:
                    pi_ans[iter, j] = price_p[i, j] - bid_b[i, j]
                    j -= 1
                    i -= 1
                    if j >= 0 and i >= 0:
                        price_p_ans[iter, j] = price_p[i, j]
                        bid_b_ans[iter, j] = bid_b[i, j]
                        Y_max_ans[iter, j] = Y_max_ans[iter, j + 1] - 1
                    continue
                else:
                    if j < 0:
                        loop_exited = True
                        break
                    pi_ans[iter, j] = -bid_b[i, j]
                    if j <= 0:
                        loop_exited = True
                        break
                    j -= 1
                    if j >= 0:
                        price_p_ans[iter, j] = price_p[i, j]
                        bid_b_ans[iter, j] = bid_b[i, j]
                        Y_max_ans[iter, j] = Y_max_ans[iter, j + 1]
            else:
                if j < 0:
                    loop_exited = True
                    break
                pi_ans[iter, j] = 0
                if j <= 0:
                    loop_exited = True
                    break
                j -= 1
                if j >= 0:
                    price_p_ans[iter, j] = price_p[i, j]
                    bid_b_ans[iter, j] = bid_b[i, j]
                    Y_max_ans[iter, j] = Y_max_ans[iter, j + 1]
        if i < 0 or loop_exited:
            break

# Plotting with percentiles and more spacing
def plot_with_percentiles(data_arrays, titles, y_labels):
    fig, axs = plt.subplots(2, 2, figsize=(12, 10))
    lines, labels = [], []

    for data_array, title, y_label, ax in zip(data_arrays, titles,
y_labels, axs.flat):
        data_array = np.array(data_array)
        data_array = np.delete(data_array, 0, axis=1)
        p25 = np.percentile(data_array, 25, axis=0)
        p75 = np.percentile(data_array, 75, axis=0)
        mean_vals = np.mean(data_array, axis=0)
        x_vals = list(range(1, data_array.shape[1] + 1))
```

```
        # Color-coded lines
        l1, = ax.plot(x_vals, mean_vals, marker='o', color='blue',
label="Mean")
        l2, = ax.plot(x_vals, p25, linestyle='--', color='orange',
label="25th Percentile")
        l3, = ax.plot(x_vals, p75, linestyle='--', color='green',
label="75th Percentile")

        ax.set_title(title)
        ax.set_xlabel("Time Period")
        ax.set_ylabel(y_label)
        ax.set_xticks(x_vals)
        ax.grid()
        ax.invert_xaxis()

        if not lines:
            lines = [l1, l2, l3]
            labels = [line.get_label() for line in lines]

    # Shared legend below
    fig.legend(lines, labels, loc='lower center', ncol=3,
bbox_to_anchor=(0.5, 0.02))
    plt.tight_layout(rect=[0, 0.05, 1, 1])
    plt.subplots_adjust(hspace=0.4)  # More space between rows
    plt.show()

# Run the plot
plot_with_percentiles(
    [Y_max_ans, pi_ans, price_p_ans, bid_b_ans],
    ["Inventory Over Time", "Profit Over Time", "Price Over Time", "Bid
Over Time"],
    ["Inventory", "Profit", "Price", "Bid"]
)
```

Simulation Using the Gamma Distribution Example 2

```
import numpy as np
import scipy.stats as stats
from scipy.optimize import minimize
import matplotlib.pyplot as plt

# Parameters
alpha_X = 6
beta_X = 0.5
T = 15  # Number of periods
Y_max = 6  # Max inventory level to consider
n_sim = 1000

# Initialize storage
pi_ans = np.zeros((n_sim + 1, T + 1))
price_p_ans = np.zeros((n_sim + 1, T + 1))
bid_b_ans = np.zeros((n_sim + 1, T + 1))
Y_max_ans = np.zeros((n_sim + 1, T + 1))

# Functions (from paper, Example 2)
def mu_b(b):
    return 0.4 * b

def lambda_b(b):
    return (1 + 0.4 * np.exp(4 - 4 * b)) / (1 + np.exp(4 - 4 * b))
```

```
def sigma_b(b):
    return 1 + 0.1 * b

def F_p_given_b(p, b):
    z = (p - mu_b(b)) / sigma_b(b)
    z = np.maximum(z, 0)
    return stats.gamma.cdf(z, a=alpha_X, scale=beta_X)

# Dynamic programming matrices
pi = np.zeros((Y_max + 1, T + 1))
bid_b = np.zeros((Y_max + 1, T + 1))
price_p = np.zeros((Y_max + 1, T + 1))
p_of_purchase = np.zeros((Y_max + 1, T + 1))
p_of_click = np.zeros((Y_max + 1, T + 1))

# Optimization loop
for t in range(1, T + 1):
    for y in range(1, Y_max + 1):
        def objective(x):
            b, p = x[0], x[1]
            term1 = (1 - lambda_b(b)) * pi[y, t - 1]
            term2 = lambda_b(b) * F_p_given_b(p, b) * (pi[y, t - 1] - b)
            term3 = lambda_b(b) * (1 - F_p_given_b(p, b)) * (p + pi[y - 1,
t - 1] - b)
            return -(term1 + term2 + term3)

        result = minimize(objective, x0=[1.0, 2.0], bounds=[(0, None), (0,
None)], method='L-BFGS-B')
        pi[y, t] = round(-result.fun, 2)
        bid_b[y, t] = round(result.x[0], 2)
        price_p[y, t] = round(result.x[1], 2)

# Probabilities
for y in range(1, Y_max + 1):
    for t in range(1, T + 1):
        p_of_purchase[y, t] = 1 - F_p_given_b(price_p[y, t], bid_b[y, t])
        p_of_click[y, t] = lambda_b(bid_b[y, t])

# Simulation
for iter in range(n_sim + 1):
    rows, cols = p_of_purchase.shape
    price_p_ans[iter, -1] = price_p[rows - 1, cols - 1]
    bid_b_ans[iter, -1] = bid_b[rows - 1, cols - 1]
    Y_max_ans[iter, -1] = Y_max
    loop_exited = False

    for i in range(rows - 1, -1, -1):
        for j in range(cols - 1, -1, -1):
            r_click = np.random.rand()
            r_purchase = np.random.rand()

            if r_click < p_of_click[i, j]:
                if r_purchase < p_of_purchase[i, j]:
                    pi_ans[iter, j] = price_p[i, j] - bid_b[i, j]
                    j -= 1
                    i -= 1
                    if i >= 0 and j >= 0:
                        price_p_ans[iter, j] = price_p[i, j]
                        bid_b_ans[iter, j] = bid_b[i, j]
                        Y_max_ans[iter, j] = Y_max_ans[iter, j + 1] - 1
                    continue
                else:
```

```
                    if j < 0:
                        loop_exited = True
                        break
                    pi_ans[iter, j] = -bid_b[i, j]
                    if j <= 0:
                        loop_exited = True
                        break
                    j -= 1
                    if j >= 0:
                        price_p_ans[iter, j] = price_p[i, j]
                        bid_b_ans[iter, j] = bid_b[i, j]
                        Y_max_ans[iter, j] = Y_max_ans[iter, j + 1]
            else:
                if j < 0:
                    loop_exited = True
                    break
                pi_ans[iter, j] = 0
                if j <= 0:
                    loop_exited = True
                    break
                j -= 1
                if j >= 0:
                    price_p_ans[iter, j] = price_p[i, j]
                    bid_b_ans[iter, j] = bid_b[i, j]
                    Y_max_ans[iter, j] = Y_max_ans[iter, j + 1]
        if i < 0 or loop_exited:
            break

# Plotting function with spacing and colored percentiles
def plot_with_percentiles(data_arrays, titles, y_labels):
    fig, axs = plt.subplots(2, 2, figsize=(12, 10))
    lines, labels = [], []

    for data_array, title, y_label, ax in zip(data_arrays, titles,
y_labels, axs.flat):
        data_array = np.array(data_array)
        data_array = np.delete(data_array, 0, axis=1)
        p25 = np.percentile(data_array, 25, axis=0)
        p75 = np.percentile(data_array, 75, axis=0)
        mean_vals = np.mean(data_array, axis=0)
        x_vals = list(range(1, data_array.shape[1] + 1))

        # Plot with distinct colors
        l1, = ax.plot(x_vals, mean_vals, marker='o', color='blue',
label="Mean")
        l2, = ax.plot(x_vals, p25, linestyle='--', color='orange',
label="25th Percentile")
        l3, = ax.plot(x_vals, p75, linestyle='--', color='green',
label="75th Percentile")

        ax.set_title(title)
        ax.set_xlabel("Time Period")
        ax.set_ylabel(y_label)
        ax.set_xticks(x_vals)
        ax.grid()
        ax.invert_xaxis()

        if not lines:
            lines = [l1, l2, l3]
            labels = [line.get_label() for line in lines]

    # Shared legend with padding
    fig.legend(lines, labels, loc='lower center', ncol=3,
```

```
bbox_to_anchor=(0.5, 0.02))
    plt.tight_layout(rect=[0, 0.07, 1, 1])
    plt.subplots_adjust(hspace=0.4)
    plt.show()

# Plot results
plot_with_percentiles(
    [Y_max_ans, pi_ans, price_p_ans, bid_b_ans],
    ["Inventory Over Time", "Profit Over Time", "Price Over Time", "Bid
Over Time"],
    ["Inventory", "Profit", "Price", "Bid"]
)
```

Simmulation codes for the Beta and Weibull distributions are structurally similar to the Gamma distribution example and are therefore not included in the appendix.