Design of a Real-Time Pricing System for E-commerce

Archana Kumari* and Babu Rao. K

***Abstract*—The high volumes and stiff competition call for a smart (both economically and computationally) pricing systems in modern E-commerce. The three market factors, e.g., the Demand for the item, the Competition prices, and the Buyer's characteristics, are instrumental in deciding the sale prices for items of the same quality. These factors are volatile in nature, and therefore, the ability to modify the prices in real-time gives a boost to a seller's profitability. In this paper, we propose a novel design for a real-time pricing system to offer differentiated price suggestions. We build an algorithm that uses the three factors and the target sale quantity for decision making. The pricing is formulated as an optimization problem and is solved by using the technique of Linear Programming (LP). The key parameter in the LP equations is the sale probability, which is derived from historical price requests and 'price request to sale' ratio. The simulated results demonstrate that, by executing multiple subsequent optimization cycles, the pricing solution generates higher revenue than static pricing choices. In order to reduce the time required to serve price requests, we have decoupled the pricing calculation from the request paths, leading to an extremely fast pricing solution.**

***Index Terms*—E-commerce pricing, pricing algorithm, real-time system design, linear programming**

I. Introduction

A. Growth of E-commerce

Over the past couple of decades, the internet-based e-commerce business has been growing at a remarkable rate. Global e-commerce is now the largest in the electronic industry [1]. As per statistics [2], B2C e-commerce sales are expected to grow to approximately 5 trillion by the end of 2022. The internet has simplified the buying and selling of goods and services, and the barrier to entry for new players has come down. With more sellers joining the business of e-commerce and customers having broader price transparency, the business has become very competitive. Under this high competition, it has become harder to generate a profit.

B. Dynamic Pricing

Profitability is directly linked with revenue. As Demirci, and Alptekin [3] discussed, revenue management in e-commerce is the process of decision-making with the objective of maximizing the revenue. Successful revenue management is about matching supply and demand and it involves understanding how customers think and what their perceptions of value are. Pricing (i.e., the price quoted to the customer) is one of the most important aspects of revenue management. It is a process that yields the maximum revenue and, hence, optimal profits. Establishing the appropriate price for the items is the ultimate challenge for revenue maximization. The solution to pricing challenges has been approached in multiple ways. Pricing models include cost-plus pricing, fixed prices, competitive pricing, and dynamic pricing. When it comes to revenues, dynamic pricing has arguably an edge over all the strategies because it has the flexibility and power to cover a wider spectrum of prices, increase sales, and maximize the revenue.

Dynamic pricing, as a pricing strategy, allows online sellers to set multiple prices for the same product based on different combinations of market variables [4]. The price variation is essentially helpful in tapping the various segments of the market and maximizing the profit. The problem of dynamic pricing can be loosely defined as follows: Given several items to sell and a given sale horizon, adaptively adjust prices over time to maximize expected revenues. Uncertain customer demand, steady competition, changing markets, as well as remaining inventory levels must be considered. Therefore, dynamic pricing is difficult to implement and sustain. It is hard to predict the market conditions and other factors quantitatively. While there have been many approaches suggested, it is understandably hard to find a deterministic solution which can work across different product classes.

C. Factors Influencing Dynamic Pricing

Historically, pricing research for selling commodities (offline & online) has focused on various variables to come up with the pricing decision. Some of these are perishability or durability of the item, cost of production & storage, demand for the product, presence & nature of the competition (duopoly, oligopoly), and customer segmentation (demography, individual traits), etc. Demand-based pricing aims to estimate the fluctuation in demand (typically based on seasonality) and it is widely practiced in the transport and hospitality industries (airlines, taxi services, hotels). Competition pricing is to follow the prices set by the competitors. Essentially, when the competition increases or decreases the prices, in order to improve profitability or to stay relevant in the market, the seller also moves prices for her item in the same direction [5]. The approach often depends on competitive settings but with limited demand information. Another important factor is customer characterization. Customers are more than one kind [6, 7], and e-commerce generates massive user behavior data that can provide great value [8] for deriving the prices per segment of customers. Finally, the 'Price elasticity of demand,' i.e., the sensitivity of demand to variations in the price of a commodity, is at the heart of many dynamic pricing approaches, and this allows sellers to come up with multiple price combinations that facilitate revenue optimization.

D. Proposal: 'Dynamic Pricing in Real-Time'

Real-time pricing is the practice of changing prices very quickly based on various sales factors, such as stock

Manuscript received July 16, 2022; revised September 20, 2022; accepted October 17, 2022.

Archana Kumari is with CMR University, India.

Babu Rao. K is with School of Engineering and Technology, CMR University, India.

*Correspondence: archana.17phd@cmr.edu.in

DOI: 10.7763/IJCTE.2023.V15.1329

availability of a high-demand product, prices charged by competitors, the browsing history of customers, customers' previous purchases, and even the weather. In this paper, we are going to address the challenge of practical dynamic pricing using a data-driven approach based on linear programming. Our objective is to demonstrate that the approach can provide good quality results in a reasonable execution time. We make use of historical data to derive estimations for sales probabilities, customer segmentation (myopic and strategic customer separations), and competition statistics. Additionally, we start with a sales target, i.e., sell 'N' items in a specified time period ('T'). Our strategy breaks down the sales target 'N' for a large period into multiple smaller 'period targets' using the historical sale volume profile. The LP is built on sales probabilities, 'period targets,' and customer segments. The LP Solution provides multiple price values which can be shown to the customer on the price request.

The paper is structured as follows: The existing literature on dynamic pricing and its various aspects is reviewed and discussed in Section II. Section III outlines the problem's scope, objectives, and building blocks prior to actually describing how a linear programming solution for dynamic pricing has been devised. The system's implementation and how to create real-time pricing are also covered in the same section. In Section IV, we highlight the results of our research and compare them with cost-plus pricing, a standard pricing strategy. Finally, the conclusion and suggestions for future work are discussed in Section V.

II. Related Literature

Dynamic Pricing is a multidisciplinary field of study that applies economics, operations research (OR), statistics, stochastic optimization, computational research, data analytics, and behavioral science to the Revenue Management challenge [9]. The existing Dynamic Pricing literature points to several approaches, strategies, and techniques that have been used to solve this challenge, many of which are specific to a particular use case. In this section of the paper, we will categorize and summarize the extensive research that has been conducted in this field.

A. Business Specific Use-Cases

Hotels have been adopting dynamic pricing as a lever to influence demand based on a range of variables, such as advance booking, length of stay, group size, seasonality, special events, weekdays and weekends, customer profiles, and distribution channels [10]. The aviation industry uses dynamic pricing to determine the price of a product (i.e., the ticket) on a daily basis. Given that the commodity is perishable and cannot be refilled, the challenge for the airline business is to set a price at each moment that minimizes this revenue loss and takes account of the customers' willingness to pay [11]. Cloud service platforms (for example, AWS, GCP) may use dynamic pricing to discount unsold compute instances at a lower price and maximize revenue [12, 13].

B. Demand, Product Type, Customer Type & Competition

A significant amount of research has been conducted to understand demand as it is the most critical aspect of the Dynamic Pricing strategies. Price elasticity of demand is a dimensionless construct that measures how a product's consumption changes in response to a change in its price [14]. Khandelwal *et al.* [15] extended the price elasticity with a function for demand forecasting. Seyedan and Fereshteh [16] focused their study on demand forecasting in supply chain management, where high-dimensional data is generated from numerous sources. Adenso-Díaz *et al.* [17] made mathematical formulation for a dynamic price for perishable food items and strategies the offering of aged units at a lower price than fresh units. Nurma *et al.* [18] employed personalization and customer segmentation to build price differentiation and boost sales. Kremer *et al.* [19] employed customer segmentation into myopic and strategic customers, respectively. Demand and competition are the primary considerations with a dedicated algorithm to gather competition prices [20].

C. Flavors of Dynamic Pricing Strategies

The optimal dynamic pricing and inventory problem for a dual-channel supply chain with a single manufacturer and retailer is examined by Li and Mizuno [21], they formulate the problem in the context of unpredictable demand and various power configurations. Grigoriev *et al.* [22] had their approach dedicated to a single product sold over a finite time window, with discount scheduling and managing the price sensitivity around demand by suggesting the discount schedule. To take a step further in price differentiation, the proposal of NYOP (Name your own price) showed up in [23], i.e., the onus of suggesting the initial price is on the buyer, and the offer is accepted if it is above some threshold price set by the seller. Harsha *et al.* [24] employed simulated markets of buyers with varying levels of strategic sophistication, and they adopt dynamic stochastic modelling to solve the problem of pricing for retailers which do sell the same stock of products both in-store and online. The goal of dynamic pricing proposal is to achieve the best price while minimizing potential revenue losses caused by unknown demand-parameters [25]. The optimization is carried out under the assumption that the demand function is linear under a fixed and short selling interval.

D. Technological Handlings: Data-Mining, Machine Learning and Artificial Intelligence

Kastius and Schlosser [26] attempted to employ RL (reinforcement learning) as it focuses on solving games by maximizing a reward function and can potentially address the limitations of other approaches in presence of competition. They computed self-adaptive pricing strategies using DQN and SAC algorithms as well as presented the performance comparisons. In order to maximize total expected revenue, Pasechnyuk *et al.* [27] found the equilibrium prices by establishing the balancing of supply and demand. The revenue function assumes that consumers follow the discrete choice demand model and suppliers are aware of the costs of quantity adjustments. They have used stochastic gradient methods to solve the formulated optimization problem. Ghose and Tran [28] discussed an optimization approach using neural networks to come up with demand and customer expectation models dependent on intrinsic product features such as quality and post-sale services. Alzhouri *et al.* [13]

explored dynamic pricing for underutilized resources in order to optimize cloud revenue. They use linear programming to develop a workable stochastic model, and trials prove that this method of dynamic pricing can increase or reduce the price effectively and efficiently. For their, delivery cost dynamic pricing problem, Strauss *et al.* [29] had formulated a feasible linear programming equation and joined demand management decisions with routing costs while accounting for customer choice behavior. Chen and Wang [30] used the technique of data mining on customer behavior data to come up with a dynamic pricing model for e-commerce. Predictive big data analytics (BDA) is employed by [16] for supply-channel demand forecasting. Jiang and Guo [31] proposed a pricing system which reconciles the quality and dynamic pricing by incorporating the user reviews online backed by the simulation approach. Cohen *et al.* [32] attempted to address the multi-item multi-period pricing problem that supermarket retailers face, and they develop a graphical representation for profit maximization and then find the result by solving the maximum weighted path problem on a layered graph. Chen and Chen [25] have developed a model that has polynomial-time solutions when demand for information is limited. The model adopts the minimax regret criterion to make robust online decisions with limited information, which is applied to a variety of problems, such as pricing.

E. Summary of Literature Survey

The existing literature on dynamic pricing is very diverse, ranging from business use-cases, demand modeling, competition pricing, inventory management, multichannel business models, use of Big-data techniques, and reinforcement learning. In the context of e-commerce and its high growth, it is important to have the ability to generate dynamic pricing in real-time, which we perceive is an area where not much work has been conducted. We also think that, the linear programming that has been successfully applied in other use-cases can be explored for the retail e-commerce case.

III. Detailed Description of the Pricing Strategy

A. A Typical User Story

The user in our story is a retail business owner. Please refer to her as Alice. On an e-commerce platform, Alice sells items such as fashion apparel, groceries, and electronics. She has a good market share for the products she sells, so she wants to maximize her company's revenue. She manually adjusts the prices of her products, using a hybrid of discount pricing [33] and cost-plus pricing [34]. She plans to use data from her platform, such as item clicks, sales, and customer behavior, to implement powerful dynamic pricing on non-KVIs (known value items). She usually has a sales target for these products, which is defined as the “number of items to be sold over a certain period of time” (e.g., a million mobile covers over the next 3 months of a popular dimension). She wants to use a strategy that is both powerful and intuitive, and that can respond to the price request in real time while not being computationally expensive.

B. Objective Statement

Design a system for real-time dynamic pricing solutions that provides high-quality, competitive prices while remaining computationally cost-effective.

C. Assumptions and Scope

The pricing strategy discussed here makes some intuitive assumptions. For the purpose of generality, we have simulated the seller prices and customer behavior data to carry out all the experiments. In order to simulate the demand behavior, our first assumption is that demand for online retail items follows seasonality. For example, some item categories have higher sales during festive seasons, or sales over the weekends are statistically expected to be higher than on weekdays. Next, the simulated market data assumes the customer base is segmented into myopic and strategic customers. We shall discuss simulation data in more detail in the coming sections. Handling new product launches is outside the scope of this work because sellers typically decide to manually handle prices during the initial stage of the product using strategies such as a hybrid of discount pricing or cost-plus pricing.

D. Pricing Elements

As previously stated, the following market factors are the key for pricing decisions in a modern e-commerce setup. The first one is the *Ongoing Demand;* it’s widely researched and the practiced one. The *Demand* drives the prices up and down when there is a lack of it. In this work, the rate of incoming client price requests (count per period) and the sale probability represent the demand. Another factor is the *Customer Segment*, we consider myopic customers to be those who are less affected by *time* or *Competitive pricing*. Strategic customers are the inverse of myopic ones. So, it does make sense to analyze and present segments separately. Apart from the three market factors, *Sale Target* represents inventory at the seller’s place, and sellers' quoted prices have taken the same into consideration.

E. Schema for Input data

Our dynamic pricing solution is driven by data, and in order to derive the ‘pricing element’ described earlier, we use the schema defined in Table I to simulate the input data:

TABLE I: Schema for Input Data to Pricing Strategy

Time	Customer Myopia	Competition Price	Price Offered	Sold?
Date-Time	(Float, <1)	Fixed point Number	Fixed point Number	Boolean

Even though we used simulated inputs in this study, columns like Time, Price Offered, and Sold could be easily extracted from request and transaction data sources in a real e-commerce setup (such as tables in (Relational Database Management System) RDBMS). Myopia is a customer characteristic that should be quantified as the percentage of purchases made regardless of competition prices (an individual customer's myopia should range from 0-1, with '0' being completely strategic and '1' being completely myopic).

F. Competition Data Collection

In Table I, we have included a column for competition prices, which is important for *strategic customers*. Finding the competition prices is essentially a task of infrastructure design, and a typical setup would deploy a web-crawler periodically polling the competition portals for the item prices. Fig. 1 describes a typical real-world data pipeline for our use-case. As stated, our work is a ‘proof of concept,’ we used simulation to generate the data corresponding to the schema described in Table I.

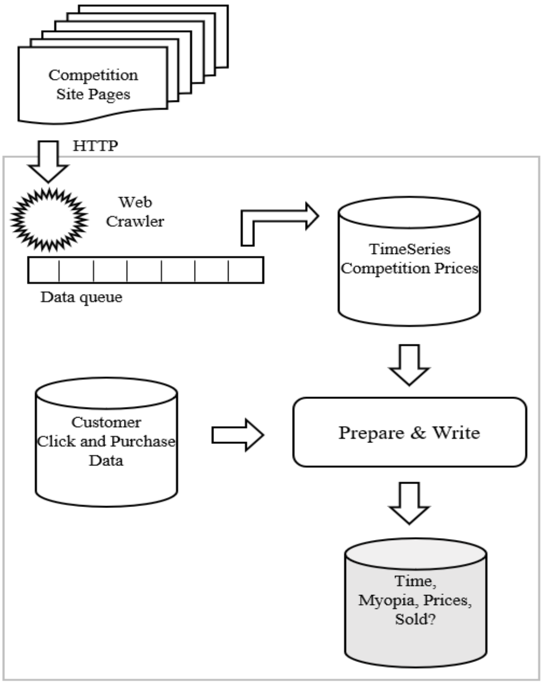

Fig. 1. Real world ‘Data Acquisition and Preparation’.

G. Quantification of Customer Segments

Myopic & Strategic segment ratios: Myopic requests are requests from customers characterized as ‘myopic.’ Fig. 1 defines a customer's myopia as a floating-point number, i.e., a percentage of total purchases she makes at prices higher than the competition. To classify a customer as myopic, we must first define a threshold (say 20%), and if the customer's myopia ratio is greater than this number, she is classified as myopic; or else, she is classified as strategic. We define a ratio for myopic customer segments in our pricing formulation, and we express this as:

$$C_1 = \frac{Number\ of\ Price\ Requests\ from\ Myopic\ Customer}{Total\ number\ of\ Price\ Requests} \quad (1a)$$

Similarly, for strategic customers request ratio, we can define C_2 as:

$$C_2 = \frac{Number\ of\ Price\ Requests\ from\ Strategic\ Customer}{Total\ number\ of\ Price\ Requests} \quad (1b)$$

H. Sale Probabilities Estimation

Precise ‘Demand’ is not feasible to evaluate, so we use the calculated sale probabilities from the raw data. We define ‘sale probabilities’ at Price $\boldsymbol{P_t}$ for the customer segment C_k as below:

$$\forall P_t \in \boldsymbol{P}, C_k \in \boldsymbol{C},\ S_{tk} = \frac{\sum Items\ Sold\ at\ price \geq Pt}{\sum Prices\ offered\ at\ price \geq Pt} \quad (2)$$

where $\boldsymbol{P}$ is the set of possible price points for the algorithm to work on and $\boldsymbol{C}$ is the set of customer segments. The price set $\boldsymbol{P}$ can either be decided by the user (seller) or can be trivially set as a fixed increasing sequence (Eq. 3) by using price-step (***stp***) and min and max prices ($\boldsymbol{P_{min}}$ and $\boldsymbol{P_{max}}$, are defined by the user).

$$\{\ stp * P_t \mid Pt \in [\ P_{min}\ ..\ P_{max}\ ]\ \} \quad (3)$$

Based on Eq. 2, *Sale-probabilities* are calculated from the raw input data and stored on an ongoing basis in the following schema, as shown in Table II.

TABLE II: ONGOING SALE-PROBABILITIES TABLE

Customer	***Price***	***Sale Probability***
C_1	P_1	S_{11}
C_1	P_2	S_{21}
C_2	P_3	S_{32}
C_2	P_4	S_{42}

I. Simulation of Input Data

This section describes the shape of simulated data that we populate as input into Table I as part of our experimental setup. A random (or gauss random function) function around a mean value can be used to trivially simulate the daily request count (Fig. 2(A)). We consider seasonality; Fig. 2(B) shows a comparison of weekend vs. weekday request volume. Fig. 3(A) and Fig. 3(B) show a typical price distribution for strategic and myopic customers, respectively. The same graphs show how many items were sold at each price. Here, C1 (Eq. 1a) is around 0.09, implying that strategic requests outnumber strategic ones by about tenfold. The price elasticity of demand among strategic customers is greater than that of myopic customers, and thus the sale probability for strategic customers changes more in response to changes in offered prices (Fig. 3(A) and Fig. 3(B)).

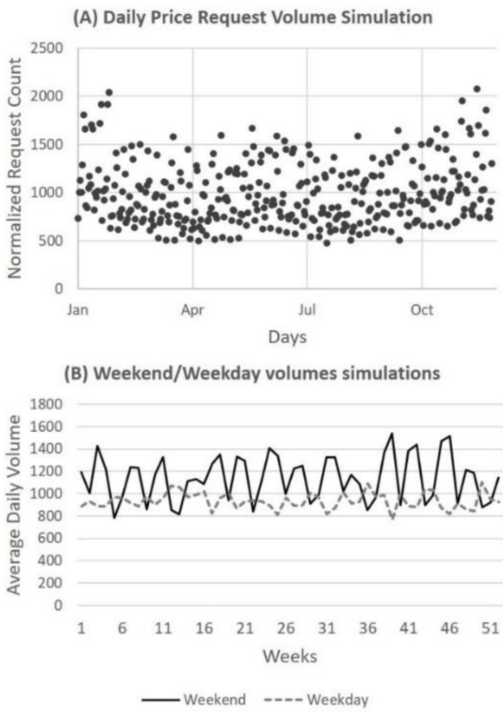

Fig. 2. Simulation of demand.

We use the Python programming language for simulation of input data, with a schema shown in Table I. This data contains rows of customer requests, whether the product was sold or not, and other important pricing factors. We use prepackaged Python utilities like 'math' and 'random' to introduce randomization into the simulation data.

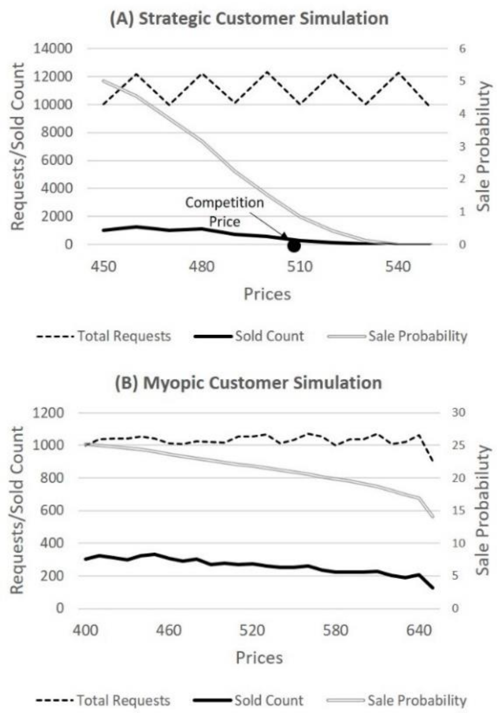

Fig. 3. Customer response simulation.

J. Periodic Optimization Cycle & Period Sale Target

As stated in our user story, sellers have a longer time horizon for selling. We run the optimization cycle on a regular basis in this approach (a much smaller time window than the overall time horizon). This is done to account for changes in market dynamics, such as changes in demand or competition prices.

For each period of the optimization cycle, we create a 'period-sale-target' by splitting the user-specified sale-target into smaller targets. By doing this, we ensure the full available time horizon is utilized and sellers do not lose revenue by trying to sell the significant inventory to early buyers. Also, when dealing with larger inventory, being sale-target aware lets the seller be a bit more aggressive in terms of pricing. In this work, we have experimented with two approaches to sale-target splitting, i.e., *TimeWeighted* (TW) and *VolumeWeighted* (VW). The first approach, Time-weighted, simply suggests that the 'sale-target' be divided equally across all periods, whereas *VolumeWeighted*, as the name implies, sets the target based on historical sale volume.

K. Linear Programming (LP) Formulation

We argue that this problem has now taken the shape of a linear programming challenge. We are going to optimize for *Total Revenue* (Y). Let us define Q_{tk} as the total number of price requests served by the engine with price P_t in customer segment k, then the objective function for maximizing the revenue can be written as below:

Find, Vector $Q \rightarrow \{ Q_{tk} \}$ such that it,

$$Maximizes, Y_{(revenue)} = \sum_{k=1}^{2} RC_k P_t S_{tk}\, Q_{tk} \quad (4)$$

where R is the total expected number of price requests across k customer segments (in our case, $k = 2$, myopic and strategic customers), C_k represents the ratio of expected request count in the customer category (k) and total requests. P_t is the price to offer, S_{tk} is the sale probability at price P_t, customer segment k. The following are the constraints for our optimization problem:

Constraint 1: The total sale should be less than the target sales.

$$T_{(target\ sale)} \geq \sum_{k=1}^{2} RC_k S_{tk} Q_{tk} \quad (5)$$

Constraint 2: The sum of 'customer segment ratios' should be one.

$$1 = \sum_{k=1}^{2} C_k \quad (6)$$

Constraint 3: The price should be bound to min and max values. Quantities should be zero or positive integers.

$$P_t := \{ P_t \in \mathbb{R}^n \mid P_{min} \leq P_t \leq P_{max} \} \quad (7a)$$

$$Q_{tk} := \{ Q_{tk} \in \mathbb{Z}^{mxn} \mid 0 \leq Q_{tk} \} \quad (7b)$$

L. System Implementation

Combination of Customer Segmentation & periodic LP optimization has the potential to provide a great quality result. In order to be real-time for customer requests, we want to shift the expensive computations (sales probability computation, running the LP Cycle) off the 'time sensitive critical path' of request processing. Fig. 4 depicts the system implementation of the approach discussed. There are two distinct modules which are shown 1) Price-Generator, and 2) Price-Request-Handler.

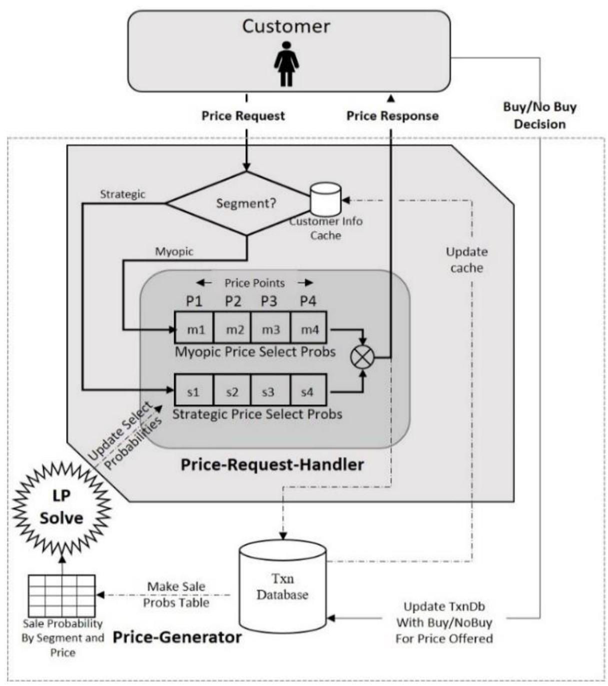

Fig. 4. System implementation of real-time dynamic pricing.

The Price-Generator is where most of the pricing logic has been built. It has the responsibility of periodic target creation, calculation of sales probabilities, solving the LP periodically, and finally yielding the quantity vectors (Q_{tk} in the LP Objective function).

```
algorithm price-generator is
  INPUT:
      TotalSaleTarget, TotalTime, OptimzationCycleTimePeriod
  Output:
      Saves PriceSelectProbabilities for price-generator
 Body:
TotalCycles := TotalTime/ OptimzationCycleTimePeriod
 For Cycle → 1 to Total Cycle:
      PeriodTarget := TimeWeighted/VolumeWeighed  for cycle
      PeriodSaleData   := TxnDbLookup(cycle);
      SaleProbailities :=Calculated from PeriodSaleData
      PriceRequestCounts := SolveLP( … );
      PriceSelectProbabilities ← PriceRequestCounts
      Save( PriceSelectProbabilities );
      Sleep( Rest of the cycle);
```

Fig. 5. Pseudo-code for price-generator component.

As the quantity vector is only the expected request count at various price points, in order to select a price for a given customer segment (C_k) we define a *PriceSelectProbabilities* for different prices (P_t).

$$PSP_{tk} = \frac{Q_{tk}}{\sum Q_{tk}} \tag{8}$$

The pseudo-code for the Price-Generator is given in Fig. 5.

While the price-generator will periodically create the *PriceSelectProbabilities*, the buyer will interface with something called a *'Price-Request-Handler.'* This module is supposed to perform a computationally low-cost operation in order to meet the requirement of low latency. In the event of a price request, the requesting customer is first mapped into the right segment (myopic/strategic), likely via a quick cache read. Then, by using the *PriceSelectProbabilities* map for the segment, a price is selected randomly (based on the associated probability of the price point). The pseudo code for the price-request handler is given in Fig. 6:

```
algorithm price-request-handler is:
  INPUT:
      CustomerId, Product
   OUTPUT:
      Dynamic Price
   CustSegment = CacheLookup( CustometId )
   PriceSelectProbabilities ← From the Price-Generator output
   selector ← random();
   return PriceSelectProbabilities[selector]
```

Fig. 6. Pseudo-code for price-request-handler component.

M. *Solving the LP*

We have used the Python programming language to do the system implementation. Google OR-tools [35] have been used to solve the LP formulation. The OR-tools provide an easy-to-use way to formulate the objective function (Eq. 3) and constraints (Eq. 4, Eq. 5, Eq. 6a & Eq. 6b) and run the optimization cycle to yield a quantity vector ($\{Q_{tk}\}$) and convert the same to *Price Selection Prabilities* (Eq. 8).

N. *Establishing the Sufficiency of Required Data Points*

The solution described in the previous section assumes that sufficient data points exist, and in this section, we propose a potential solution to how to ascertain this. To formulate the LP equations in our experiment, we needed myopic segment ratio and sales probability. We computed these variables with an increasing number of 'sample sizes' and, after a certain number of points, the calculations stated to converge. Fig. 7 shows the plot for sales-probability (at a fixed price point) and myopic segment ratio. We annotate on the graphs (using gray circles) where the saturation point lives.

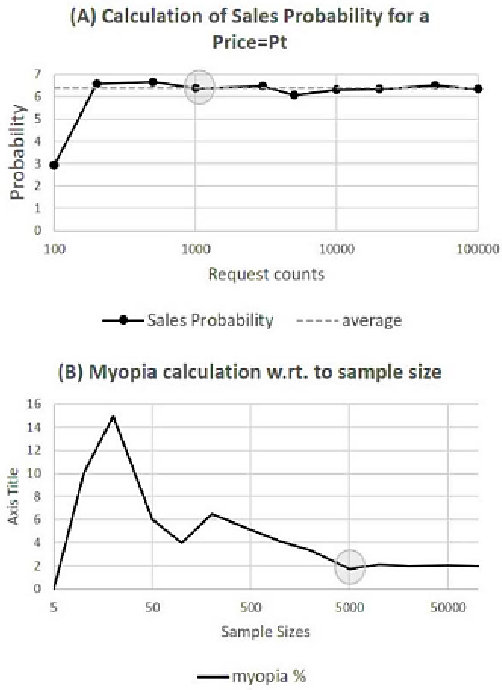

Fig. 7. Determining the right sample size.

IV. RESULTS

We conducted a few experiments with the previously described input to evaluate the quality of our algorithm's output. As stated in the objective statement, the system should produce high-quality competitive prices at a low computational cost, and the results presented below reflect our findings.

A. *Quality of the LP Solution*

The quality of the solution is defined as the revenue generated by selling the target sale quantity; any unsold quantity is considered stale. We compared the revenue generated by our algorithm to various cost-plus pricing schemes (i.e., 15, 25, 40 & 50 percent over the production cost). We performed pricing action (using our LP solution and cost-plus pricings) with multiple target sale quantities, sweeping from minimum to maximum. Our results show that the LP solution (with both *TimeWeighted* and *VolumeWeighted* target splitting) outperforms all cost-plus pricing schemes (as shown in Fig. 8 (A)). The performance difference can be explained intuitively by the following facts:

1. The LP solution distinguishes between myopic and strategic clients (hence it can sell some quantities at high prices, even with high inventories).

2. The solution can select multiple prices for specific target quantities (Fig. 8(C)), resulting in higher revenue.

It is worth noting that the envelope of all cost-plus pricing is not significantly inferior to the LP solution, and it would not be incorrect to assume that the LP approach automates the cost-plus pricing process once a sale target is provided, and it does so simultaneously across customer segments.

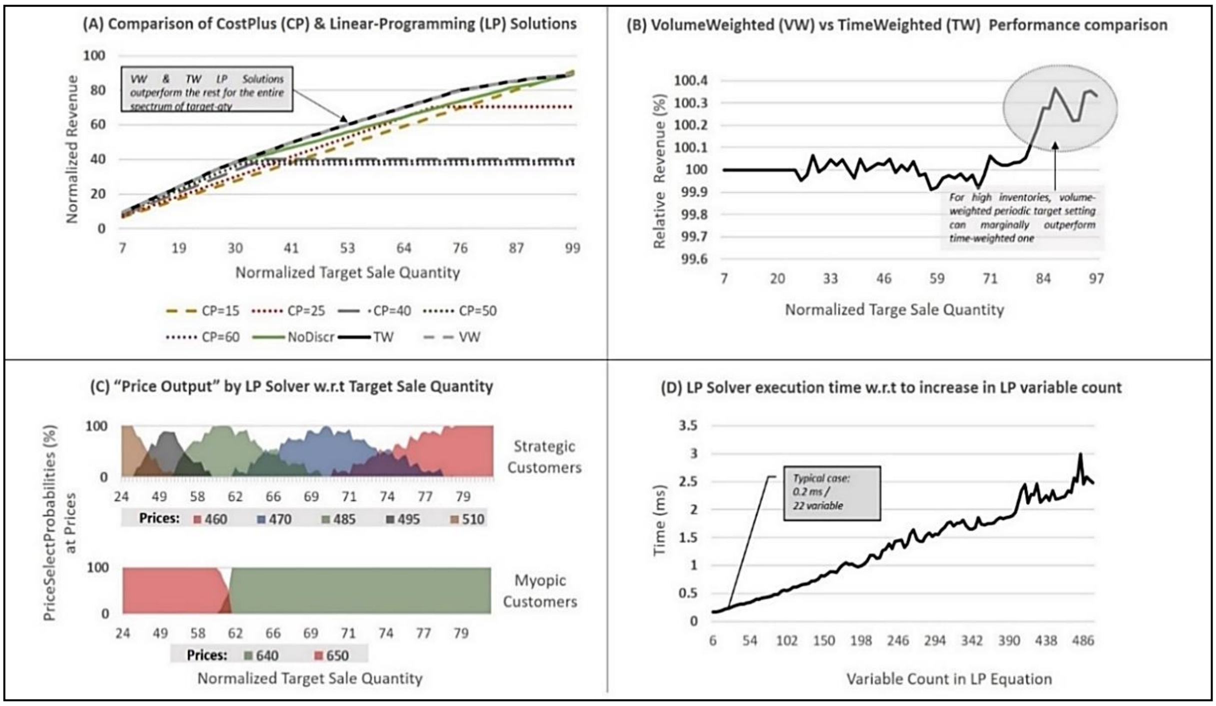

Fig. 8. Results (A) comparison of Cost-plus (CP) & Linear-Programming (LP) solutions (B) VolumeWeighted (VW) vs TimeWeighted (TW) performance comparison (C) "Price Output" by LP Solver w.r.t Target Sale Quantity (D) LP Solver execution time w.r.t to increase in variable count.

B. Comparison of VW and TW Approaches

We use VolumeWeighted and TimeWeighted approaches to divide the targets into smaller ones. The revenues of the VW and TW approaches are compared in Fig. 8(B), and it is shown that there is little difference between the TW and VW approaches for smaller target sales quantities, but the VW approach works better when there is more to sell. The reason for the performance difference is that the TimeWeighted approach does not account for 'High volume time period,' and as a result, when the 'target sale quantity' is high, the TW approach sells fewer items than the VW approach.

C. 'Price Output' by the LP Solution

Fig 8 (C) depicts how LP chooses price points for customer segments with varying target sale quantities. When the 'target sale quantities' are high, LP offers a mix of lower prices and vice versa. The graph also implies that LP solution would be able to automatically raise prices when demand exceeds available inventory.

D. Time Effectiveness of the LP Solution

Finally, the final graph (Fig. 8(D)) shows how long it takes to run an LP cycle. This is a critical metric for determining whether this solution will be used in real-time. In this experiment, we attempted to increase the number of price points, and thus the number of variables in LP equations (see Eq. 4). Python was used to run the experiment on a general-purpose Windows-10 OS on an Intel i5 CPU. LP could complete most practical use cases in a millisecond.

Given that one LP cycle can serve up to 100-1000 customer price requests, the average time spent running LP optimization on general-purpose hardware should be in the single digit microsecond range, making this strategy ideal for real-time pricing requirements.

V. Conclusion

The task of dynamic pricing is difficult, and doing so in real-time is even more challenging. This paper proposes a deterministic, practical, and high-quality solution to the same problem. The solution's simplicity stems from the fact that all the information needed to build it is available in a typical transaction store. The findings show that using periodic optimization with a split target quantity increases revenue. Linear programming is also a deterministic tool, which is a plus. While LP is not traditionally thought to be computationally expensive (due to its linear time complexity), we still move this cost off the request path in this work, which improves computational efficiency and, as a result, makes the proposed system a viable real-time solution for e-commerce. The study also argues that the proposed solution is scalable for both low and high-volume products.

A. Contributions Summary

The following are the value additions provided by the proposed work: 1) This research develops a novel sale probability-based LP formulation of the dynamic pricing problem for e-commerce. 2) Proposes running multiple LP optimization cycles for split target sale quantity across smaller time windows to react to changes in demand and competition pricing. 3) Provides a system architecture for offloading expensive computations such as sales-probability calculation and solving the LP equations from client requests, reducing the time required to serve client requests.

B. Potential Future Work

This work purposefully excludes the 'cold start' or new product launch from its scope; however, a heuristic can be built to address the same and incorporate it into this framework. Another potential future work is managing and validating the data quality needed to estimate the sale-probability. The LP solution is not only a data user, it is also able to generate data that can be used in subsequent optimization cycles, which is an additional topic for research.

Conflict of Interest

The submitted work was carried out with no conflict of interest. Hence, the authors declare no conflict of interest.

Author Contributions

Archana Kumari developed the methodology of work, performed experiments, collected the results, and wrote the paper. Babu Rao. K provided guidance in the preparation of the paper. All the authors had approved the final version.

References

[1] P. Kumar and N. Agarwal, "E-commerce: A catalyst of marketing especially on FMCG product under Covid pandemic," *Asian Journal of Advances in Research*, vol. 8, no. 2, pp. 1-11, May 2021.

[2] Retail e-commerce sales worldwide from 2014 to 2024. [Online]. Available: https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/

[3] B. Demirci and E. S. Alptekin, "Revenue management in e-commerce," in *Proc. the International Multi Conference of Engineers and Computer Scientists*, vol. 2, 2013, pp. 8-10.

[4] A. MacKay and S. Weinstein. (December, 2021). Dynamic pricing algorithms, consumer harm, and regulatory response. *SSRN Electronic Journal.* [Online]. Available: https://ssrn.com/abstract=3979147

[5] Y. Lei, S. Jasin, and A. Sinha, "Joint dynamic pricing and order fulfillment for e-commerce retailers," *Manufacturing & Service Operations Management*, vol. 20, no. 2, pp. 269-284, March 2018.

[6] G. Liu, Z. Guan, and W. Hua, "Dynamic pricing under cost reduction in the presence of myopic and strategic consumer," *Discrete Dynamics in Nature and Society*, pp. 1-15, June 2019.

[7] Y. Levin, J. McGiil, and M. Nediak, "Optimal dynamic pricing of perishable items by a monopolist facing strategic consumers," *Production and Operations Management*, vol. 90, pp. 40-60, 2010.

[8] W. Yang and J Guo, "Consumers' purchase behavior preference in e-commerce platform based on data mining algorithm," *International Journal of Circuits, Systems and Signal Processing*, vol. 16, pp. 603-609, Jan 2022.

[9] W Han and B Bai., "Pricing research in hospitality and tourism and marketing literature: A systematic review and research agenda," *International Journal of Contemporary Hospitality Management*, vol. 34, no. 5, pp. 1717-1738, 2022.

[10] G. Abrate, R Sainaghi, and A. Mauri, "Dynamic pricing in Airbnb: Individual versus professional hosts," *Journal of Business Research*, vol. 141, pp. 191-199, 2022.

[11] D. Leon and R. Tabatabaei, "A stochastic dynamic pricing model for the multiclass problems in the airline industry," *European Journal of Operational Research*, vol. 242, pp. 188-200, 2015.

[12] J. Li, J. H. Wang, and J. Wang, "Dynamic pricing for idle resource in public clouds: Guarantee revenue from strategic users," in *Proc. IEEE Conference on Computer Communications Workshops*, 2022.

[13] F. Alzhouri, A. Agarwal, and Y. Liu, "Maximizing cloud revenue using dynamic pricing of multiple class virtual machines," *IEEE Transactions on Cloud Computing*, vol. 9, no. 2, pp. 682-695, 2021.

[14] P. R. Thimmapuram and J. Kim, "Consumers' price elasticity of demand modeling with economic effects on electricity markets using an agent-based model," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 390-397, 2013.

[15] L. Khandelwal, J. Regan, and D. Little, "Promotheus: An end-to-end machine learning framework for optimizing markdown in online fashion e-commerce," in *Proc. the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3447-3457, Aug 2022.

[16] M. Seyedan and M. Fereshteh, "Predictive big data analytics for supply chain demand forecasting: Methods, applications, and research opportunities," *Journal of Big Data*, vol.7, no. 1, pp. 1-22, 2020.

[17] B. Adenso-Díaz, S. Lozano, and A. Palacio, "Effects of dynamic pricing of perishable products on revenue and waste," *Applied Mathematical Modelling*, vol. 45, pp. 148-164, 2017.

[18] S. Nurma, L. Nugroho, R. Ferdiana, and P. Santosa, "Review on customer segmentation technique on ecommerce," *Advanced Science Letters*, vol. 22, pp. 3018-3022, 2016.

[19] M. Kremer, B. Mantin, and A. Ovchinnikov, "Dynamic pricing in the presence of myopic and strategic consumers: Theory and experiment," *Production and Operations Management*, vol. 26, no. 1, pp. 116-133, 2017.

[20] S. B. Hwang and S. Kim, "Dynamic pricing algorithm for e-commerce," in *Advances in Systems, Computing Sciences and Software Engineering. Springer*, Dordrecht, pp. 149–155, 2006.

[21] M. Li and S. Mizuno, "Dynamic pricing and inventory management of a dual-channel supply chain under different power structures," *European Journal of Operational Research*, vol. 303, no. 1, pp. 273-285, 2022.

[22] A. Grigoriev, B. Hiller, S. Marbán, T. Vredeveld, and R. Zwaan, "Dynamic pricing problems with elastic demand," *Operations Research Letters*, vol. 40, no. 3, pp. 175-179, 2012.

[23] O. Hinz, H. Hann, and M. Spann, "Price discrimination in e-commerce? An examination of dynamic pricing in name-your-own price markets," *MIS Quarterly*, vol. 35, pp. 81-98, March 2011.

[24] P. Harsha, S. Subramanian, and J. Uichanco, "Dynamic pricing of omnichannel inventories," *Manufacturing & Service Operations Management*, vol. 21, 2018.

[25] M. Chen and Z. L. Chen, "Optimal pricing to minimize maximum regret with limited demand information," *Computers & Operations Research*, vol. 124, December 2020.

[26] A. Kastius and R. Schlosser, "Dynamic pricing under competition using reinforcement learning," *Journal of Revenue and Pricing Management*, vol. 21, pp. 50–63, 2022

[27] D. Pasechnyuk, P. Dvurechensky, S. Omelchenko, and A. Gasnikov, "Stochastic optimization for dynamic pricing," in *Proc. OPTIMA 2021. Communications in Computer and Information Science*, vol. 1514, pp. 82-94, 2021.

[28] T. K. Ghose and T. T. Tran, "A dynamic pricing approach in e-commerce based on multiple purchase attributes," *Advances in Artificial Intelligence*, vol. 6085, pp. 111-122, 2010.

[29] A. Strauss, N. Gulpinar, and Y. Zheng, "Dynamic pricing of flexible time slots for attended home delivery," *European Journal of Operational Research*, vol. 294, no. 3, pp. 1022-1041, 2021.

[30] Y. Chen and F. Wang, "A dynamic pricing model for e-commerce based on data mining," in *Proc. International Symposium on Computational Intelligence and Design*, vol. 1, pp. 363-366, 2009.

[31] Y. Jiang and H. Guo, "Design of consumer review systems and product pricing," *Information Systems Research*, vol. 26, no. 4, pp. 714-730, 2015.

[32] M. C. Cohen, S. Gupta, J. Kalas, and G. Perakis, "An efficient algorithm for dynamic pricing using a graphical representation," *Production and Operations Management*, vol. 29, no. 10, pp. 2326-2349, 2020.

[33] S. Lee, "A research on pricing strategy of e-commerce logistics," in *Proc. 2010 International Conference on E-Business and E-Government*, 2010, pp. 5028-5030.

[34] M. Simnica, "Pricing strategies and practices in different industries," in *EMCB 2020 Conference Proceedings*, June 2020, pp. 35-42.

[35] OR-Tools 9.3. Laurent Perron and Vincent Furnon. [Online]. Available: https://developers.google.com/optimization/

Archana Kumari is currently pursuing the Ph.D. (computer science) from CMR University, Bengaluru. She has five years of experience teaching computer science in various engineering colleges. She did B. Tech in computer science and engineering from Uttar Pradesh Technical University and M. Tech in computer science and engineering from Bhagwant University, Rajasthan.

Babu Rao. K teaches artificial intelligence, ANN, computer vision, information, and network security at CMR University's School of Engineering and Technology. He has served and held several academic and administrative positions including registrar I/C., principal, and director, He has also conducted workshops on FDP – ISTE-IITB &IITK STTPs sponsored by the MHRD India. He is also a reviewer for reputed Journals and conferences, and edited one book while supervising research scholars, whilst guiding M. Tech students in their academic dissertation work.