Ann Oper Res (2006) 143: 59–75
DOI 10.1007/s10479-006-7372-3

Learning dynamic prices in electronic retail markets with customer segmentation

C. V. L. Raju · Y. Narahari · K. Ravikumar

Abstract In this paper, we use reinforcement learning (RL) techniques to determine dynamic prices in an electronic monopolistic retail market. The market that we consider consists of two natural segments of customers, *captives* and *shoppers*. Captives are mature, loyal buyers whereas the shoppers are more price sensitive and are attracted by sales promotions and volume discounts. The seller is the learning agent in the system and uses RL to learn from the environment. Under (reasonable) assumptions about the arrival process of customers, inventory replenishment policy, and replenishment lead time distribution, the system becomes a Markov decision process thus enabling the use of a wide spectrum of learning algorithms. In this paper, we use the Q-learning algorithm for RL to arrive at optimal dynamic prices that optimize the seller's performance metric (either long term discounted profit or long run average profit per unit time). Our model and methodology can also be used to compute optimal reorder quantity and optimal reorder point for the inventory policy followed by the seller and to compute the optimal volume discounts to be offered to the shoppers.

Keywords Electronic retail market · Dynamic pricing · Customer segmentation · Captives · Shoppers · Volume discounts · Inventory replenishment · Markov decision process · Reinforcement learning · Q-learning

Notation:

(q, r)	Inventory policy, q = reorder quantity and r = reorder point
λ	Rate of Poisson arrival process of customers
f	Fraction of customers who are captives
p	Posted price per unit item
N	Maximum number of items requested by backlogged customers

C. V. L. Raju (✉) · Y. Narahari
Electronic Enterprises Laboratory, Computer Science and Automation, Indian Institute of Science
e-mail: {raju, hari}@csa.iisc.ernet.in

K. Ravikumar
General Motors India Science Labs, Bangalore
e-mail: ravikumar.karumanchi@gm.com

I_{max}	Maximum inventory level at the retail store
$\frac{1}{\mu}$	Mean replenishment lead time
$\frac{1}{\mu_s}$	Mean time after which a shopper revisits the retail store
w	Lead time quote provided by the retailer to the captives
$U_c(p, w)$	Captive's utility for a given p and w
p_c, w_c	Private price and lead time respectively of an arriving captive
$\beta, (1-\beta)$	Weights given to lead time and price respectively of an arriving captive in the utility function
$U_s(p)$	Shopper's utility for a given price p
p_s	Private price of an arriving shopper
H_I	Holding cost per unit item per unit time
H_q	Back-logged cost for each back-logged demand per unit time
$X(t)$	State of the system at time t
A	Set of pricing actions (ie., set of possible dynamic prices)
$P_{ij}(p)$	Transition probabilities of the underlying Markov process with going price p
$F_{ij}(.\|p)$	Distribution function of time until next transition
$S_p(.)$	Single stage revenue of selling items to customers
$C(.)$	Backorder cost per unit per unit time
π, π^*	Stationary deterministic policy and optimal stationary deterministic policy respectively
$V_\pi(i)$	Long-term discounted expected profit from state i for the policy π
t_n	n^{th} Transition epoch
α	Discount factor in discounted optimality
$Q(i, p)$	Long-term expected profit starting from state i when the first action to be followed in state i is p
γ_n	Learning parameter
J_π	Expected long run averaged reward starting from state i for the policy π
T_{ij}	Mean sampled transition time from state i to j

1. Introduction

Sellers have always faced the problem of setting the right prices for goods and services that would generate the maximum revenue for them. Determining the right prices to charge a customer for a product or a service is a complex task. It requires that a company knows not only its own operating costs and availability of supply but also how much the customer values the product and what the future demand would be Elmaghraby and Keskinocak (2002). A company therefore needs a wealth of information about its customers and also be able to adjust its prices at minimal cost. Advances in Internet technologies and e-commerce have dramatically increased the quantum of information the sellers can gather about customers and have provided universal connectivity to customers making it easy to change the prices. This has led to increased adoption of dynamic pricing and to increased interest in dynamic pricing research.

1.1. From fixed pricing to dynamic pricing

There is a revolution brewing in pricing that promises to profoundly alter the way goods are marketed and sold. In the future, sellers will offer special deals, tailored just for every customer, just for the moment on everything (right price to the right customer at the right time). Behind this sweeping change is the wiring of the economy. The Internet, corporate networks, and wireless setups are linking people, machines, and companies around the globe and connecting sellers and buyers as never before. This is enabling buyers to quickly and easily compare products and prices, putting them in a better bargaining position. At the same time, the technology allows sellers to collect detailed data about customers' buying habits, preferences, even spending limits, so they can tailor their products and prices. This raises hopes of a more efficient marketplace.

1.1.1. Examples of dynamic pricing

Quantity or volume discounts is a simple example of dynamic pricing that is followed by almost all retailers and merchants. Consumer segmentation is another simple example: senior citizens may be allowed discounts, students and academic institutions are allowed discounts on software packages, etc. Sales promotions provide another common example. We present below some specific examples of e-business companies.

The airline industry is an outstanding example of successful and profitable deployment of dynamic pricing strategies. The kind of pricing strategy followed here is popularly known as yield management or revenue management (Mcgill and van Ryzin, 1999; Smith et al., 2001). Essentially, the method here is to dynamically modulate prices over time by adjusting the number of seats available in each pre-defined fare class. Advantage is taken of a natural segmentation in the consumers: business travelers for whom the flight dates and timings are primary and fares are secondary; casual travelers for whom prices are important and the dates/timings are flexible; and hybrids for whom both factors are at an equal level of importance. Yield management systems essentially forecast demand, closely monitor bookings, and dynamically adjust seats available in each segment, so as to maximize profits. This method is currently being practiced in hotel rooms, cruises, rental cars, etc.

Priceline.com allows travelers to name their price for an airline ticket booked at the last minute and get a ticket from an origin to a destination at a fractional cost of the full fare. Priceline.com uses complex software that enables major airlines to fill unsold seats at marginal revenues. The business model of Priceline.com is attractive for airlines since it can generate additional revenues on seats that would have otherwise gone unsold. Transactions through Priceline.com do not influence buyers with high willingness to pay since a number of serious restrictions apply to the cheaper tickets.

Buy.com (Smith et al., 2000; DiMicco et al., 2002) uses software agents to search web sites of competitors for competitive prices and in response, Buy.com lowers its price to match these prices. The pricing strategy here is based on the assumption that their customers are extremely price sensitive and will choose to purchase from the seller offering the lowest price. This has resulted in Buy.com register high volumes of trade, however due to the low prices, the profits are low, often times even negative. This example illustrates that overly simplistic or incorrect model of buyer behavior can produce undesirable results.

Amazon.com is another example of a e-business company which has experimented with dynamic prices on their products, for example, popular DVDs. Depending on the supply and demand, the prices on a particular DVD varied over a wide range. Customers found out about this and were not comfortable at what they saw as random prices on a commodity which is plenty in supply. Thus price fluctuations can often lead to reduced loyalty from customers if fairness is not perceived.

1.2. Motivation, contributions, and outline

The examples of Buy.com and Amazon.com illustrate that overly simplistic or incorrect model of buyer behavior can produce undesirable results. On the other hand, the example of yield management in the airlines industry suggests that sound modeling and analysis can lead to a dynamic pricing strategy that can generate substantially higher revenue. Today's economy is ready for dynamic pricing, however the prices will have to be adjusted in fairly sophisticated ways to reap the benefits of dynamic pricing. Motivated by this, in this paper, we look into an approach for dynamic pricing in typical electronic retail markets. In particular, we consider a single seller, monopolistic market with two natural segments of customers as already described. Machine learning in general and reinforcement learning in particular have been shown in the literature to be effective modeling tools for solving the dynamic pricing problem. In this paper, we show the natural way in which RL can be applied to the problem at hand and develop dynamic pricing strategies for the retail merchant based on this model. The following are the specific contributions of this paper.

- We consider a single seller retail store (such as amazon.com) which sells a designated product and offers volume discounts for customers buying multiple items. This leads to two segments of customers described above. By making reasonable assumptions on the seller's inventory policy, replenishment lead times, and the arrival process of the customers, we set up a Markov decision process model for the dynamics of this system. In this model, the actions correspond to the prices offered by the seller to the two segments of customers.
- We show that the seller can use reinforcement learning strategies to modulate his prices dynamically so as to maximize a chosen performance metric. We consider two representative performance metrics: long term discounted profit and long run time averaged profit. The seller uses Q-learning to learn the optimal dynamic pricing policy.
- We show that the model and our methodology can be used to provide support for tactical decision making such as determining the optimal reorder quantity, optimal reorder point, and optimal level of volume discount to be offered by the seller.

The paper is organized as follows. In Section 2, we review relevant literature. In Section 3, we provide a detailed description of the retail market that we model in this paper and present all the assumptions about the dynamics. In Section 4, we show that the system dynamics is captured by a Markov decision process and set up an RL framework for the dynamic pricing problem for the retail market. We present the Q-learning framework for two performance metrics: (1) long term discounted profit (2) long run time averaged profit. Section 5 describes a simulation experiment with long term discounted cost as the performance metric and brings out the insights in the dynamic pricing strategy obtained. Section 6 describes a simulation experiment with long run profit per unit time as the performance metric and shows how we can use the analysis to determine optimal reorder quantity, optimal reorder point, and optimal volume discount level. We conclude the paper in Section 7 with a summary and several directions for future work.

2. A review of relevant work

Dynamic pricing in retail markets has been researched quite extensively, since decades. Early works include that of (Varian, 1980) and (Salop and Stiglitz, 1982). (Elmaghraby and Keskinocak, 2002) and (Swann, 1999) provide a comprehensive review of models of traditional retail markets where inventories are a major consideration. (Elmaghraby and

Keskinocak, 2002) discuss two types of markets: markets with no inventory replenishment and markets with inventory replenishment.

Our interest in this paper is specifically on the use of machine learning based models, which have recently emerged as a popular modeling tool for dynamic pricing. In a typical market, the environment constantly changes with demands and supplies fluctuating all the way. In such a scenario, it is impossible to foresee all possible evolutions of the system. The amount of information available is also limited (for example, a seller does not have complete information about the pricing of the competing sellers). With machine learning-based models, one can put all available data into perspective and change the pricing strategy to adapt best to the environment. We provide a review of relevant work in this area. For a detailed review, the reader is referred to the survey paper by Narahari et al. (2003).

Gupta et al. (2002) consider a web-based multi-unit Dutch auction where the auctioneer progressively decrements per unit price of the items and model the problem of finding a decrementing sequence of prices so as to maximize total expected revenue, in the presence of uncertainty with regard to arrival pattern of bidders and their individual price-demand curves. The above decision problem is modeled as a single agent reinforcement learning in an uncertain non-stationary auction environment. Under the assumption of independent bidder valuations, the authors develop a finite horizon Markov Decision Process model with undiscounted returns and solve it using a Q-learning algorithm.

Carvalho and Puttman (2003) consider the problem of optimizing sales revenues based on a parametric model in which the parameters are unknown. For example, after t days of sale, a seller knows the prices he has set on the preceding $t - 1$ days and can observe the demands on the preceding $t - 1$ days. The seller can learn about the parameters of the demand function and use it to set prices so as to maximize the revenues over a given time horizon. Several pricing rules are studied and compared. It is shown that a one-step look-ahead rule performs fairly robustly for a single seller environment studied.

In the paper by Brooks et al. (1999), the performance of two different pricing strategies (both based on machine learning) is compared in the context of single seller markets for electronic goods. The first strategy uses a one parameter pricing model and the second one uses a two parameter pricing model. It is shown that a dynamic pricing strategy based on two parameter learning outperforms the one based on one parameter learning. The paper derives analytical methods to determining optimal prices for a model with complete information. It is shown that the profits increase as one moves from a one parameter model to a two parameter model. Simulations are used to explore a dynamic model in which the seller is uncertain about customer valuations and learns the optimal prices gradually.

Hu and Zhang (2002) studies three different types of pricing algorithms (or pricing agents) in a simulated market. The first agent uses reinforcement learning to determine the prices, by learning an optimal action for one period based on the rewards it receives for that action. The second agent uses a traditional Q-learning method, by learning about Q-values which represent long-term optimal values for the agent's own actions. The third agent uses a sophisticated Nash Q-learning algorithm, by learning about Q-values which represent long-term Nash equilibrium values for agent's joint actions. The third agent performs better than the second and the second outperforms the first agent in a simulated market where the agents compete with one another. This shows that learning methods that take future rewards into account perform better than myopic methods. Also, the learning method that takes into account the presence of other agents performs better than the method that ignores other agents.

The paper by DiMicco et al. (2002) describes a simulator using which different types of markets can be simulated with different types of dynamic pricing algorithms, including machine learning based algorithms.

In our paper here, we look at a single seller monopolistic market where volume discounts segment the customers in a natural way into two categories, namely captives and shoppers. The market considered also models inventory replenishment and stochastic arrivals of customers. A study of such a market model in the context of dynamic pricing has not been done before. The use of reinforcement learning in such a setting also is unique.

3. A Model of a retail store with customer segmentation

Online retail stores have attempted to segment their consumer markets in the hope of charging different prices and reaping the benefits of dynamic pricing. Internet technology permits online retail market companies to customize their marketing and pricing to fit particular market segments (for example, nearly all online purchases of material goods require a conventional mailing address—an address that can tell a merchant a great deal about the background of a particular customer). By using data from the actual behavior of individuals, e-commerce companies have the potential to micro-manage their marketing and pricing strategies, so as to customize nearly every sales offer.

We consider a retail store where customer segmentation is done based on simple volume discounts. Amazon.com is one example. Any retail store for goods like apparels, DVDs, food, etc. would provide other immediate examples. Volume discounts could be of the form: buy two, take three (for example buy two shirts and get a third one free).

To make our developments simple, we confine to the case with two types of price-quantity packages offered by the retailer; price for unit quantity and *buy two and get one free*. Such offers can be realistically implemented as follows: Consumers who access the web-page of any retailer would initially see unit price offer against the item requested for and also will see an alert depicting “discounts available at higher volumes”. Consumers who select the former option or package can be safely assumed to be not so price-sensitive and hence would be willing to pay high price for the item if any additional service offer also comes as part of the package. We assume that retailer offers a lead time commitment to such consumers in the event of no stock being available at the time of request. Customers who choose this option will *learn* over time such additional service benefits from the retailer and adhere to the same retailer for future purchases. We call such customers “captives” to the retailer. We assume further that each captive places an order (or purchases, if stock is readily available) only if he derives strictly positive utility from the price quote and lead time quote of the retailer, else leaves the market. On the other hand, the customer who wishes to choose the second offer will have low willingness to pay per unit item and select the retailer with lower value of the first option (provided this price is affordable to him), with ties broken arbitrarily. These customers would be willing to bear with the inconveniences imposed by the retailer. Dynamic pricing policies of airlines reflect the same phenomenon. In our study, we consider *waiting-based* inconvenience by each seller as detailed below. Captives get preference over the second class of customers with regard to supply of items in the absence of stock. The retailer can implement this priority in the following way. The consumer who clicks on the alert will be next led to a transaction page that exhibits the volume offer in the event of availability of stock after serving the pending “captive” orders. Either when no stock is available or when replenishment quantity to arrive is just sufficient enough to meet the pending orders, then the customer will be requested to place his orders in his shopping cart. The customer will revisit his shopping cart after a random time interval to check for the availability status and will cancel his order if he again observes stock-out, else will purchase the quantity at the offer made provided the price quote is rewarding enough for him and balks from the system otherwise. The retailer serves these customers in the order according to the time-stamps recorded at the shopping-carts. We call such customers as *shoppers* in the sequel. This type of behavior models, for example, the consumers who purchase objects for a future resale.

The retailer follows the standard (q, r) policy for replenishment with values for q and r so chosen as to ensure all captives identical expected lead time quotes and dynamically chooses his unit price in the above setting so as to maximize his profits in the presence of uncertainty with regard to arrival pattern of consumers, their purchase behavior and also, under uncertain replenishment lead times. Further, the seller is assumed to incur unit purchase price and holding cost per unit per unit time for keeping items in his inventory and also, cost per unit time per back-logged request. We assume zero ordering costs. Since each customer, captive or otherwise, is served according to the time-stamps recorded, backlogged requests can be modeled using *virtual* queues. Backlogged captive orders form a priority queue at each retailer whereas the shoppers' requests in their respective shopping carts form an *orbit* queue with single retrial. It is important to note that only the seller but not the customers will be able to observe these queues.

We analyze dynamic pricing in the monopolized retail market where the queues described above model the dynamics at the retailers. Figure 1 provides a schematic for these dynamics. Below we provide the mathematical formalism for the above model.

- Customers arrive at the market according to a Poisson process with rate λ. A fraction f of these customers are captives at the seller and the remaining fraction constitute the shoppers.
- The seller posts per unit price p as part of his menu to the arriving customers.
- The seller turns away any arriving request if the total demand backlogged exceeds N at that point in time.
- The seller has finite inventory capacity $I_{\max}$ and follows a fixed reorder policy for replenishing; when the inventory position, current inventory level plus the the quantity ordered, falls below a level r, the seller would order for replenishment of size $(I_{\max} - r)$. This is the classical (q, r) policy of inventory planning.
- The replenishment lead times at the seller is exponentially distributed with mean $\frac{1}{\mu}$.
- The captive measures his utility of a price quote p and lead time quote w (equal to the expected replenishment lead time, $\frac{1}{\mu}$) by:

$$U_c(p, w) = [(1 - \beta)(p_c - p) + \beta(w_c - w)]\Theta(p_c - p)\Theta(w_c - w) \tag{1}$$

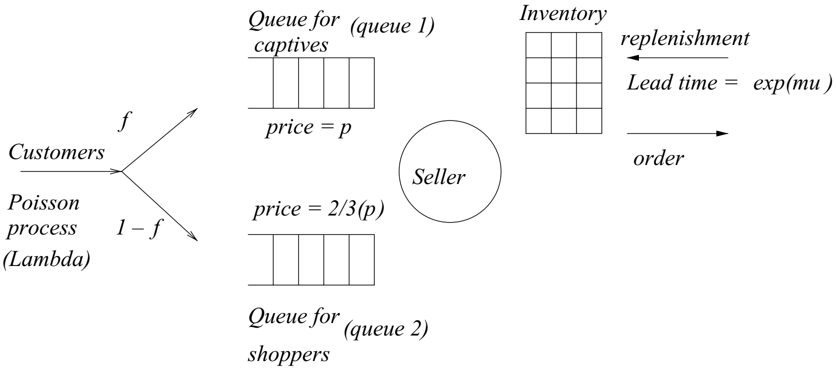

Fig. 1 A model of a retail store with two customer segments

where $\Theta(x) = 1$ if $x \geq 0$ and is zero otherwise, and $0 \leq \beta \leq 1$. $p_c \sim U(0, p_{\max}]$ and $w_c \sim U(0, w_{\max}]$ with $U(.)$ denoting the uniform distribution over the specified interval for given $p_{\max}$ and $w_{\max}$.

- The shopper measures his utility of the unit price option $\frac{2p}{3}$ on the menu of the seller by:

$$U_s(p) = (p_s - p)\Theta(p_s - p) \tag{2}$$

where $p_s \sim U(0, p_{\max})$ for a given $p_{\max}$ and p_s is the shopper's maximum willingness to pay per unit item.

- Every shopper upon placing an order in his shopping cart at the retailer will revisit the retailer again after an interval of time distributed exponentially with rate μ_s.
- The seller sets his unit price p from a finite set A.
- The seller incurs a holding cost rate of H_I per unit item per unit time and a cost of H_q per each back-logged request. The purchasing price per unit item is P_c. We assume zero reorder costs.

The seller dynamically resets the prices at random intervals so as to maximize his expected profits. Under the above Markovian distributional assumptions, the dynamic pricing problem can be reduced to a Markov Decision Process. However, in reality, retailers do not in general have knowledge about the distributions underlying the model and also about buyers' behavior. In such cases, these retailers learn about their most beneficial prices over time using Reinforcement Learning (RL), an apt paradigm for learning in Markov decision processes. In this paper, we consider two performance metrics: (1) long term total discounted profit the seller will accumulate (over an infinite time horizon) and (2) long run profit per unit time the seller can make.

4. Reinforcement learning framework for dynamic pricing in the retail store

Because of the assumptions about the arrival process, replenishment process, and the customer waiting process, the Markovian nature of the dynamics is immediate. Reinforcement learning procedures have been established as powerful and practical methods for solving decision problems in Markov decision processes (Sutton and Barto, 1998; Singh, 1994). RL expects a *reinforcement signal* from the environment indicating whether or not the latest move is in the right direction. The Markov decision process described above is tailor made for the use of reinforcement learning. The seller is a natural learning agent here. The customers are segmented into shoppers and captives in a natural way and the seller can quickly identify them based on whether or not they are seeking volume discounts. The seller can observe the queue sizes. He also knows the rewards incumbent upon the entry or exit of customers. These rewards serve as the reinforcement signal for his learning process. Thus the application of RL is natural here.

The queues queue 1 and queue 2 in Figure 1 at the retailer are virtual queues for captives and shoppers at that retailer. Note that queue 2 is an orbit queue with each shopper who has registered his orders in the shopping cart will make a retrial after an interval of time that is assumed to be exponentially distributed with mean $\frac{1}{\mu_s}$.

Let $\mathbf{X}(t) := (X_1(t), X_2(t), I(t))$ be the state of the system at the retailer with $X_i(.)$'s representing the number of backlogged requests in queue i at the retailer and $I(.)$, the inventory level at the retailer at time t. The retailer posts unit price quote and the volume discount alert on his web-page and will reset the prices only at transition epochs, that is, whenever a purchase happens (and hence the inventory drops) or when a request is backlogged in either of the queues. Recall that the captive (shopper) will purchase or make a backlog request only when U_c in (1) (U_s in (2)) is positive. It is easy to see that price dynamics can be modeled as a continuous time Markov Decision Process model. Below we give the state dynamics:

At time 0, the process $\mathbf{X}(t)$ is observed and classified into one of the states in the possible set of states (denoted by S). After identification of the state, the retailer chooses a pricing action from A. If the process is in state i and the retailer chooses $p \in A$, then

(i) the process transitions into state $j \in S$ with probability $P_{ij}(p)$
(ii) and further, conditional on the event that the next state is j, the time until next transition is a random variable with probability distribution $F_{ij}(.|p)$.

After the transition occurs, pricing action is chosen again by the retailer and (i) and (ii) are repeated. Further, in state i, for the action chosen p, the resulting reward, $S_p(.)$, the inventory cost, $H(i)$ and the backorder cost $C(i, j)$ costs are as follows: Let $i = [x_1, x_2, i_1]$ and $j = [x_1', x_2', i_1']$.

$$\begin{aligned} S_p(i, p, j) &= p \text{ if } x_1' = x_1 + 1 \\ &= p \text{ if } i_1' = i_1 - 1 \\ &= 2p \text{ if } i_1' = i_1 - 3 \\ &= 0, \text{ otherwise} \\ C(i, j) &= [i_1 - i_1']^+ P_c \\ H(i) &= x_1 H_q + i_1 H_I \end{aligned}$$

The following remarks are in order.

– *Remark 1*
The *complementarity condition* given below holds.

$$I(t)X(t) = 0 \ \forall t.$$

– *Remark 2*
Let p below represent the seller's price in the observed states. Then the following transitions occur:

- $[0, x_2, i_1] \to [0, x_2, i_1 - 1]$ with rate $f\lambda P(U_c(p, \frac{1}{\mu}) > 0) \ \forall x_2, i_1$
- $[0, x_2, i_1] \to [0, x_2 - 1, i_1 - 3]$ with rate $(1 - f)\mu_s P(U_b(p) > 0) \ \forall x_2, i_1$
- $[0, x_2, i_1] \to [0, x_2, i_1 - 3]$ with rate $(1 - f)\lambda P(U_b(p) > 0) \ \forall i_1$.
- $[x_1, x_2, 0] \to [x_1 + 1, x_2, 0]$ with rate $f\lambda P(U_c(p, \frac{1}{\mu}) > 0) \ \forall x_2$
- $[x_1, x_2, 0] \to [x_1, x_2 + 1, 0]$ with rate $(1 - f)\lambda P(U_b(p) > 0) \ \forall x_1$
- $[x_1, x_2, 0] \to [(x_1 - r)^+, x_2, (r - x_1)^+]$ with rate $\mu \ \forall x_2$

Discounted Optimality Let $\pi : S \to A$ denote a stationary deterministic pricing policy, followed by the retailer, that selects an action only based on the state information. Let $t_0 = 0$ and let $\{t_n\}_{n\geq 1}$ be the sequence of successive transition epochs under policy π and $X(t_n-)$ denote the state of the system just before t_n. For any $0 < \alpha < 1$, let the long-term discounted expected profit for the policy π be

$$V_\pi(i) = E_\pi\Big[\sum_{n=1}^{\infty} e^{-\alpha t_{n-1}}(S_p(X(t_n-), \pi(X(t_n-)), X(t_n))) - C(X(t_n-), X(t_n)) - \int_{t_{n-1}}^{t_n} H(X(t_n-))e^{-\alpha t}dt \mid X_1 = i\Big] \tag{3}$$

α above discounts the rewards to the initial time epoch.

Let $V_\alpha^*(i) = \max_\pi V_\pi(i)$. The retailer's problem is to find a $\pi^* : S \to A$ such that $V_{\pi^*}(i) = V_\alpha(i)$

The domain of optimization above can be in general the set of all non-anticipative policies (randomized, non-randomized or history dependent) and from the Markovian assumptions it follows that there exists always a stationary deterministic optimal policy and hence, we have restricted the domain to stationary policies only (and they are finite in our finite state model).

From the Bellman's optimality condition, it follows that:

$$V_\alpha^*(i) = \max_p \left\{\overline{R}_\alpha(i, p) + \sum_{j\in S} P_{ij}(p)\int_0^\infty e^{-\alpha t}V_\alpha^*(j)dF_{ij}(t|p)\right\} \tag{4}$$

where

$$\overline{R}_\alpha(i, p) = \sum_{j\in S} P_{ij}(p)\left[R(i, p, j) + \int_0^\infty \int_0^t e^{-\alpha s}H(i)ds dF_{ij}(t|p)\right] \tag{5}$$

(4) above can be solved algorithmically using any fixed point iteration scheme, such as the *value iteration* for the typical Markov decision processes. Such schemes are assured of convergence because of the contraction property coming from the discount factor, α. Also, one can construct the optimal policy π^* by assigning $\pi^*(i)$ to equal the maximizer on the RHS of (4). However, as we assumed earlier, the retailer does not have any information about underlying distributions involved at various stages and hence the conditional averaging that appears in (4) cannot be performed to derive the optimal value, and hence the optimal pricing policy. This motivates the retailer to use online learning to converge to optimal policy π^* eventually. One can think of devising a *learning* scheme based on any fixed point iteration methods on V^*. Even if we assume that this can be done, we will still need to know about P_{ij}'s above to construct π^*. To obviate this difficulty, the *Q-learning* has been proposed by Watkins and Dayan (1992) for Markov Decision Processes. Below we proceed to modify it to suit to our continuous time case. To motivate the algorithm, consider the following *Q-value* associated with an action p in state i:

$$Q(i, p) = \overline{R}_\alpha(i, p) + \sum_{j\in S} P_{ij}(p)\int_0^\infty e^{-\alpha t}V_\alpha^*(j)dF_{ij}(t|a) \tag{6}$$

In other words, $Q(i, p)$ represents the long-term expected profit starting from state i when the first action to be followed in state i is p, which is possibly different from the optimal action. It is easy to see that $V_\alpha^* = \max_p Q(i, p)$, and the maximizer is the optimal action to perform in state i. Thus, if one somehow gets the Q-values, it is easy to construct an optimal policy. In online learning, these Q-values are obtained from learning through actual execution of various actions in state i and measuring their relative merits. For details, please refer to Watkins and Dayan (1992).

Below we give the actual learning update rule involved in Q-learning for our continuous time MDP.

Let $t_0 = 0$ and start with an initial arbitrary guess, $Q_0(.)$ of $Q(i, p)$ above for all i and p.

- **Step 1**: At any n-th transition epoch at time t_n, observe the state i and select the price action $p_0 \in argmax_p Q(i, p)$ with probability $1 - \varepsilon$ and any other price in A with probability ε for some $\varepsilon > 0$.
- **Step 2**: If $X(t_n) = i$ and the price action chosen is p, then update its Q-value as follows:

$$Q_{n+1}(i, p) = Q_n(i, p) + \gamma_n \left[S_p(i, p, .) - H(i) \left(\frac{1 - e^{-\alpha T_{ij}}}{\alpha} \right) \right.$$
$$\left. + e^{-\alpha T_{ij}} max_b Q_n(j, b) - Q_n(i, p) \right] \qquad (7)$$

where j above is the state resulting from the action p in i and $S_p(.)$ is the reward collected from such action. T_{ij} is the average of sampled transition times between states i and j. γ_n above is called *learning parameter* and should be such that $\sum_n \gamma_n = \infty$ and $\sum_n \gamma_n^2 < \infty$.

Repeat steps (1)–(2) infinitely. Convergence is slow as is typical of any RL-algorithm. The speed of convergence can be drastically improved using function approximations to Q-values based on some observed features. We do not take up that part in this paper.

For a later reference, we will make the following remark:

Remark 3: From the learning procedure above it follows that Q-learning will eventually find *only* a stationary deterministic optimal policy.

Expected long run average reward

Let $\pi : S \to A$ denote a stationary deterministic pricing policy, followed by the retailer, that selects an action only based on the state information. Let $t_0 = 0$ and let $\{t_n\}_{n\geq 1}$ be the sequence of successive transition epochs under policy π and $X(t_n-)$ denote the state of the system just before t_n.

In this case, the performance metric, expected long run averaged reward starting from state i for the policy π be

$$J_\pi(i) = \limsup_{M\to\infty} \frac{1}{M} E_\pi \left[\sum_{n=0}^{M-1} [S_p(X(t_n-), \pi(X(t_n-)), X(t_n)) - \right.$$
$$\left. C(X(t_n-), X(t_n)) - H(X(t_n-))] | X_0 = i \right]. \qquad (8)$$

Retailer's problem is to find $\pi^* : S \to A$ such that

$$J^*(i) = \max_\pi J_\pi(i) \qquad (9)$$

Let us assume that s is a special state, which is recurrent in the Markov chain corresponding to each stationary policy. If we consider a sequence of generated states, and divide it into cycles that each of these cycles can be viewed as a state trajectory of a corresponding stochastic maximized profit path problem with state s is the termination state. For any scalar λ, let us consider the stochastic maximized profit path problem with expected stage profit $\sum_j P_{ij}(p)[S_p(i, p, j) - C(i, j) - H(i)] - \lambda$ for all i. Now we can argue that if we fix the expected stage profit obtained at state i to be $\sum_j P_{ij}(p)[S_p(i, p, j) - C(i, j) - H(i)] - \lambda^*$, where λ^* is the optimal average profit per stage from state s, then the associated stochastic maximized profit path problem becomes equivalent to the initial average profit per stage problem.

Now, the Bellman's equation takes the form:

$$\lambda^* + h^*(i) = \max_p \sum_j P_{ij}[S_p(i, p, j) - C(i, j) - H(i) + h^*(j)] \tag{10}$$

where λ^* is the optimal average profit per stage, and $h^*(i)$ has the interpretation of a relative or differential profit for each state i with respect to the special state s. An appropriate form of the Q-learning algorithm can be written as explained in Abounadi et al. (1996); Bertsekas and Tsitsiklis (1996), where the Q-value is defined as $h^*(i) = \max_p Q(i, p)$.

$$\begin{aligned} Q_{n+1}(i, p) = {} & Q_n(i, p) + \gamma_n[S_p(i, p, j) - C(i, j) - H_I(i)T_{ij} - H_q(i)T_{ij} \\ & + \max_b Q_n(j, b) - \max_c Q_n(t, c) - Q(i, p)] \end{aligned} \tag{11}$$

where j, $S_p(i, p, j)$ and $C(i, j)$ are generated from the pair (i, p) by simulation, and T_{ij} is the average sample time taken by the system for moving from state i to state j while collecting samples through simulation. We have to choose a sequence of step sizes γ_n such that $\sum \gamma_n = \infty$ and $\sum \gamma_n^2 < \infty$.

5. A simulation experiment with long term discounted profit as performance metric

5.1. Description of the system

We simulate and study the retail store model shown in Figure 1, by considering an action set (that is, set of possible prices) $A = \{8.0, 8.5, 9.0, 9.5, 10.0, 10.5, 11.0, 12.0, 13.0, 14.0\}$. The maximum queue capacities are assumed to be 10 each for queue 1 and queue 2 (this means we do not allow more than 10 waiting captives or more than 10 waiting shoppers in the retail store). The maximum inventory level $I_{\max}$ is assumed to be 20 with a reorder point at $r = 10$. We assume that $f = 0.4$, that is, 40 percent of the incoming customers are captives. We consider customers as arriving in Poisson fashion with mean inter-arrival time 15 minutes. The upper and lower limits for the uniform distribution that describes the acceptable price range for captives are assumed to be 8 and 14, respectively. These limits are assumed to be 5 and 11 for shoppers. The upper and lower limits for the uniform distribution that describes the acceptable lead time range for captives are assumed to be 0 hours and 12 hours, respectively. We consider exponential replenishment lead time for reorders with a mean of 3 hours. An impatient shopper drops out of the system after an exponential waiting time having a mean of 1.5 hours. The inventory holding cost (H_I) is chosen as 0.5 per unit per day and

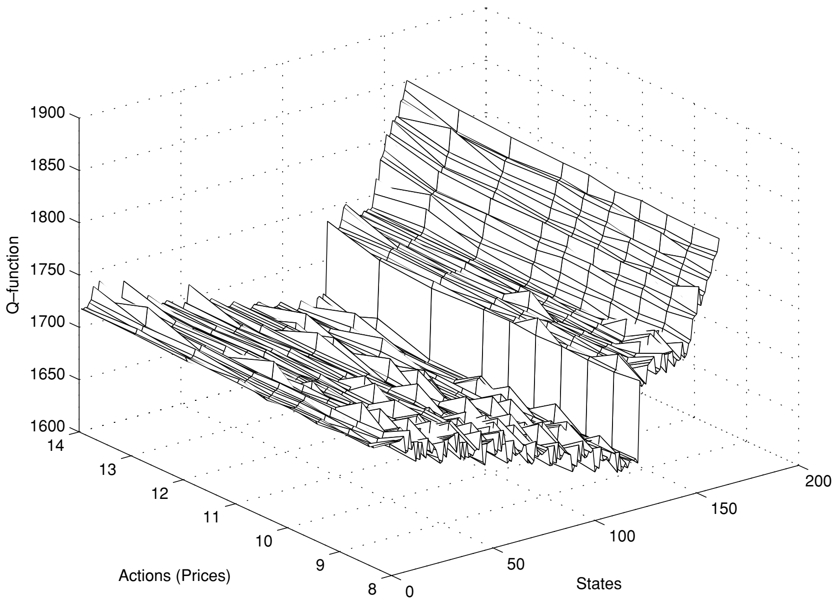

Fig. 2 Q-values for different states and actions

the backorder cost (H_q) is chosen as 0.5 per back order per day. We assume that the seller purchases the items at the rate of 4 per unit.

As already stated, we use the Q-learning algorithm Watkins and Dayan (1992) for learning the best strategy at every state of the system (Equation (4)). We use ε-greedy policy Sutton and Barto (1998) while using the Q-learning algorithm, with discount factor β set at 0.0001. Q-function values are plotted in Figure 2, where the Q values are plotted against (state, action) pairs. This figure by itself only shows some clustering but does not reveal much more. So, we tabulate the Q-function values for obtaining some insights. Table 1 shows the best action (that is, optimal dynamic price) for different individual states. By knowing the Q-function, the seller can compute the best possible price for a given state. The following observations can be made from Table 1.

- We observe that the optimal price is higher whenever the inventory level at the retail store is higher. This is fairly intuitive because of higher inventory holding costs at higher inventory levels.
- In the group of states ($x_1 > 0,\ x_2 = 0,\ i_1 = 0$), we observe high prices if there are low or high number of waiting captives and low prices if there are modest number of waiting captives. This can be explained as follows: when there is no inventory at the retail store and there are few captives in queue 1, the cost of inventory holding and back orders is low, so the seller can afford to wait for a customer who can buy at a high price. When there are many captives in queue 1, due to heavy demand, the seller would increase the price. When the number of waiting captives is neither low nor high, the seller would need to compensate for inventory holding and backorder costs, so would move towards lower price to attract more number of customers.
- In the group of states ($x_1 = 0,\ x_2 > 0,\ i_1 = 0$), we observe lower prices when queue 2 has less number of shoppers and higher prices when there are more number of shoppers.

Table 1 Dynamic prices and the corresponding states where they are optimal

Best price	States of the system
8.0	(0,0,0), (0,1,0), (0,2,0), (0,3,0), (0,4,0), (0,5,0), (1,1,0), (1,3,0), (1,6,0), (1,7,0), (2,3.0), (2,5,0), (2,6,0), (2,7,0), (2,8,0), (2,10,0), (3,1,0), (3,3,0), (3,10,0), (4,2,0), (4,3,0), (4,4,0), (4,5,0), (4,8,0), (4,10,0), (5,0,0), (5,4,0), (6,4,0), (6,9,0), (7,2,0), (7,4,0), (8,3,0), (8,5,0), (9,3,0), (10,1,0), (0,5,1), (0,6,1), (0,7,1), (0,0,2), (0,1,2), (0,3,2), (0,4,2), (0,5,2), (0,10,2)
8.5	(1,4,0), (1,5,0), (1,9,0), (2,4,0), (2,9,0), (3,2,0), (3,7,0), (3,8,0), (4,7,0), (5,5,0), (5,7,0), (5,9,0), (6,0,0), (6,1,0), (6,2,0), (6,3,0), (6,6,0), (7,1,0), (7,7,0), (7,9,0), (8,6,0), (8,7,0), (10,4,0), (0,2,1), (0,3,1), (0,2,2), (10,6,2), (0,7,2), (0,9,2), (0,0,3)
9.0	(1,2,0), (1,8,0), (3,4,0), (3,5,0), (5,2,0), (5,3,0), (5,6,0), (6,5,0), (7,0,0), (7,6,0), (8,2,0), (9,4,0), (0,8,2)
9.5	(2,0,0), (2,1,0), (3,6,0), (4,1,0), (5,1,0), (6,8,0), (7,10,0), (8,1,0), (8,8,0), (9,1,0), (10,8,0)
10.0	(3,9,0), (4,0,0), (5,8,0), (9,2,0), (9,5,0), (0,1,1)
10.5	(2,2,0), (4,6,0), (6,10,0), (7,5,0), (9,6,0), (9,7,0), (9,8,0), (10,2,0), (0,10,1)
11.0	(1,7,0), (6,9,0), (8,4,0), (8,0,0)
12.0	(0,8,0), (5,10,0), (6,7,0), (8,9,0), (10,0,0)
13.0	(0,7,0), (7,3,0), (7,8,0), (9,10,0), (10,5,0), (0,0,6), (0,0,16)
14.0	(0,6,0), (0,9,0), (0,10,0), (1,0,0), (3,0,0), (4,9,0), (9,0,0), (8,10,0), (9,9,0), (10,6,0), (10,7,0), (10,9,0), (0,0,1), (0,8,1), (0,9,2), (0,0,4), (0,0,5), (0,0,7), (0,0,8), (0,0,9), (0,0,10), (0,0,11), (0,0,12), (0,0,13), (0,0,14), (0,0,15), (0,0,17), (0,0,18), (0,0,19), (0,0,20)

This can be explained as follows: since waiting shoppers would pay only at the time of purchase, if the seller announces a high price there is a chance of losing some of the shoppers from the queue 2. But the seller can demand higher price when there are more number of shoppers with the expectation that some of the existing shoppers would buy at high price.

We wish to caution that the intuitive explanations provided above explain the trends observed in this specific instance of the system studied. For a different instance of the same system (that is, with different input parameters), exactly opposite trends might be observed. What is important to note is the complex tradeoffs that are possible and the key role played by the model (Markov decision process) and the tool (RL) in providing decision support in non-trivial settings like this. The interplay among the various system phenomena and processes is quite complex to understand and this highlights the power and use of an appropriate model and an apt tool.

Table 2 Long run average profit per unit time for various (q, r) policies

(q, r) Policies	Long run average profit per unit time
(1,19)	0.754
(2,18)	1.224
(3,17)	1.788
(4,16)	2.131
(5,15)	2.910
(6,14)	3.815
(7,13)	4.147
(8,12)	4.220
(9,11)	5.772
(10,10)	5.748
(11,9)	6.037
(12,8)	7.039
(13,7)	6.923
(14,6)	6.991
(15,5)	7.669
(16,4)	7.473
(17,3)	7.658
(18,2)	**8.410**
(19,1)	7.278

6. A simulation experiment with long run average profit as performance metric

Another performance metric that is extremely useful is the steady state or long run profit per unit time. We retain all system parameters as discussed above, except that we assume the acceptable price range for shoppers as (8,14] instead of (5,11]. We consider special state s as (0, 0, 0) (note that this state is recurrent in the Markov chain corresponding to each stationary policy).

6.1. Optimal values of reorder quantity and reorder point

Assuming the maximum inventory capacity at the retail store to be 20, we simulated different (q, r) policies with the objective of finding the the optimal (q, r) policy that would generate maximum average profit per unit time. See Table 2. Note that q is the reorder quantity while r is the reorder point. From the table, it is clear that a reorder point of 2 and a reorder quantity of 18 are optimal. This means we do not reorder until the inventory position (inventory level at the retail store plus the quantity already ordered) goes lower than 2 and when that happens, we place a replenishment order for a quantity of 18. This is a fairly counter-intuitive result, which shows the complex nature of interactions that govern the dynamics of the system.

6.2. Optimal (q, r) policy with dynamic volume discounts

Instead of a fixed volume discount (2.0/3.0) for the shoppers, we can choose the current volume discount to be offered dynamically, from a set, say, (A_v). For example, we can choose volume discount $d_v \in A_v$ from the set $A_v = \{2.0/3.0, 2.1/3.0, 2.2/3.0,\ 2.3/3.0, 2.4/3.0\}$. Choosing an action now means, selecting an ordered pair from the grid $A \times A_v$, where action $\mathbf{a} = (p, d_v) \in A \times A_v$ and shopper's price per unit item $= d_v p$. We implemented different (q, r) policies with volume discounts selected dynamically, in order to find the optimal (q, r)

Table 3 Long run average profit per unit time for various (q, r) policies optimized over different volume discount levels

(q, r) Policies	Best Long Run Average Profit Per Unit Time
(1,19)	1.117
(2,18)	1.900
(3,17)	3.265
(4,16)	4.195
(5,15)	5.704
(6,14)	7.082
(7,13)	7.441
(8,12)	8.377
(9,11)	10.260
(10,10)	9.638
(11,9)	10.564
(12,8)	11.020
(13,7)	10.309
(14,6)	10.198
(15,5)	**11.515**
(16,4)	10.646
(17,3)	10.404
(18,2)	11.327
(19,1)	9.808

policy that would generate the maximum long run average profit per unit time. See Table 3. Now (15, 5) turns out to be the best choice.

7. Conclusions and future work

In this paper, we have shown how a seller can effectively use reinforcement learning in setting prices dynamically so as to maximize his performance metrics. We believe this is a promising approach to solving the dynamic pricing problem in retail market environments with limited available information.

There are several directions for future work. We have considered a standard (q, r) inventory policy here. Other known inventory policies can be investigated as well. We have considered a single monopolistic seller with no competition. The next immediate model to consider would be a two seller model, which we are already investigating. For multi-agent situations, convergence of the learning algorithms is an important area of investigation which has engaged researchers in machine learning for quite sometime now.

References

Abounadi, J., D. Bertsekas, and V. Borkar. (1996) "Learning algorithms for Markov decision processes with average cost." Technical report, Lab. for Info. and Decision Systems, M.I.T., USA.

Bertsekas, D. P. and J. Tsitsiklis. (1996). *Neuro-dynamic Programming*. Boston, MA, USA: Athena Scientific.

Brooks, C., R. Fay, R. Das, J. K. MacKie-Mason, J. Kephart, and E. Durfee. (1999). "Automated strategy searches in an electronic goods market: Learning and complex price schedules." In: *Proceedings of the First ACM Conference on Electronic Commerce (EC-99)*, 31–40.

Carvalho, A. and M. Puttman. (2003). "Dynamic Pricing and Reinforcement Learning, URL: gg.nwu.edu/academic/deptprog/ meds-dep/OR-Seminars/Puterman.pdf."

DiMicco, J. M., A. Greenwald, and P. Maes. (2002). “Learning Curve: A Simulation-based Approach to Dynamic Pricing.”

Elmaghraby, W. and P. Keskinocak. (2002). “Dynamic Pricing: Research Overview, Current Practices and Future Directions, URL: http://www.isye.gatech.edu/ pinar/ dynamic-pricing.pdf.”

Gupta, M., K. Ravikumar, and M. Kumar. (2002). “Adaptive strategies for price markdown in a multi-unit descending price auction: A comparative study.” In: *Proceedings of the IEEE Conference on Systems, Man, and Cybernetics* 373–378.

Hu, J. and Y. Zhang. (2002). “Online Reinformcenet Learning in Multiagent Systems, URL: cimon.rochester.edu/public-html/papers/priceagent1.pdf.”

Mcgill, J. and G. van Ryzin. (1999). “Revenue management: Research overview and prospects.” *Transportation Science* 33 (2), 233–256.

Narahari, Y., C. Raju, and S. Shah. (2003). “Dynamic Pricing Models for Electronic Business.” Technical report, Electronic Enterprises Laboratory, Department of Computer Science an d Automation, Indian Institute of Science.

Salop, S. and J. Stiglitz. (1982) “The theory of sales: A simple model of equilibrium price dispersion with identical agents.” *The American Economic Review* 72 (5), 1121–1130.

Singh, S. (1994). “Learning to solve Markovian Decision Processes.” Ph.d dissertation, University of Michigan, Ann Arbor.

Smith, B., D. Gunther, B. Rao, and R. Ratliff. (2001). “E-commerce and operations research in airline planning, marketing, and distribution.” *Interfaces* 31 (2).

Smith, M., J. Bailey, and E. Brynjolfsson. (2000). *Understanding Digital Markets: Review and Assessment*. Cambridge, MA: MIT Press.

Sutton, R. S. and A. G. Barto. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.

Swann, J. (1999). “Flexible Pricing Policies: Introduction and a Survey of Implementation in Various Industries.” Technical Report Contract Report # CR-99/04/ESL, General Motors Corporation.

Varian, H. R. (1980). “A Model of Sales.” *The American Economic Review* pp. 651–659.

Watkins, C. J. C. H. and P. Dayan. (1992). “Q-learning.” *Machine Learning* 8, 279–292.