Reinforcement Learning for Dynamic Pricing with resource constraints in a competitive context

Laurie Guenin[1,2], Dominique Barth[1], and Christian Cadéré[2]

[1] DAVID Lab., Université de Versailles SQ/Université Paris Saclay, 45 avenue des Etats-Unis,78000, Versailles, France
[2] WeYield, 10 rue Nieuport, Vélizy-Villacoublay, France

Abstract. Dynamic pricing has emerged as a critical strategy in industries with limited resources and fluctuating demand, such as airlines, hotels, and car rentals. Traditional approaches rely on static price grids or manual interventions, which lack adaptability to real-time market changes. This study investigates the application of Reinforcement Learning (RL) to dynamic pricing in the car rental industry, incorporating resource constraints and competitive dynamics. A repeated game model simulates the competitive interactions of car rental companies seeking to optimize profits while managing limited fleet capacities. Based on real-world data, an experimental evaluation compares RL with a traditional resource-based pricing method and a mixed approach. Results indicate RL's superior performance in adapting to market variability, though it risks underpricing in capacity-constrained scenarios. A proposed mixed method balances competition and resource considerations, outperforming both RL and resource-based strategies by dynamically adjusting to market pressures and resource availability. The findings highlight the potential of RL and hybrid approaches in enhancing revenue management in competitive, resource-limited contexts. Future research will explore automated parameter tuning for dynamic scenario adjustments.

Keywords: Dynamic pricing · Reinforcement Learning · Car rental · Revenue Management.

1 Introduction

Dynamic pricing has become essential for optimizing revenue and improving resource utilization in various industries, particularly in contexts with limited capacities and fluctuating demands. This is especially the case of the car rental industry, given its operational complexity and sensitivity to seasonal, competitive, and customer-driven factors [18]. Traditionally, pricing strategies in this industry have relied on static price grids, predetermined based on criteria such as rental duration, vehicle category, and advance booking periods [18]. While these methods provide a baseline for pricing, they often lack the flexibility required to adapt to real-time market and concurrence changes, resulting in inefficiencies and suboptimal revenue outcomes. In recent years, some car rental companies have adopted more advanced systems, incorporating business rules derived

from revenue management practices in the airline industry [2]. Revenue Management aims to sell the right product to the right customer at the right price [5]. Methods based on Revenue Management remain underutilized and are often constrained by high manual intervention and limited scalability. The advent of automated pricing mechanisms powered by advanced computational models marks a significant paradigm shift in addressing these limitations. Among these, Reinforcement Learning (RL) has emerged as a powerful framework for developing adaptive and robust pricing strategies [4, 32]. Reinforcement Learning is particularly well-suited to dynamic pricing scenarios due to its ability to model complex, multi-agent environments with interdependent decisions. By iteratively learning from the interaction between agents and their environment, RL algorithms can optimize pricing strategies in contexts characterized by competition and resource constraints [18]. RL requires relatively little historical data and can adapt to evolving market conditions in real time. Furthermore, it offers the flexibility to incorporate various constraints, such as capacity limits, fairness in pricing, and heterogeneous customer behaviors. This study investigates the potential of Reinforcement Learning in the context of car rental pricing, comparing it with resource-based pricing. The research adopts a repeated game model to design the competitive dynamics of multiple car rental companies vying for customer demand. Each company's objective is to maximize its long-term profit by selecting optimal prices over a sequence of rental demands while considering resource constraints and the pricing strategies of competitors. The experiments leverage real-world data from a car rental company operating in a competitive market, providing a robust basis for evaluating algorithmic performance. Key metrics, such as total profit and resource utilization, are analyzed across different scenarios. Preliminary results highlight the strengths and weaknesses of the various approaches, with Reinforcement Learning demonstrating superior adaptability and performance in competitive settings. Hybrid methods, combining elements of resource management and competitive dynamics, also show promise in achieving balanced outcomes.

Our contribution. This paper starts with the modelization of our problem of choosing car rental prices using repeated game theory in Section 2. Then, three methods for selecting prices are defined in Section 3. The first one is a resource-based method inspired by the airline industry. The second one uses Reinforcement Learning with a competition-based approach, this method uses a new utility function that corresponds to our problem's needs. The last method combines both the previous methods to consider resource and competition constraints. Finally, the experimental results of the competition between these methods will be presented and discussed.

2 Related Work

The car rental industry, historically reliant on manual pricing and heuristics, is increasingly adopting advanced revenue management techniques driven by

automation and AI. Current manual pricing methods are often based on price grids. The EMSR heuristic [2], derived from airline revenue management, offers an efficient approximation method for demand-based pricing decisions, but it remains a method based on price grids. Recent developments in the car rental industry include the integration of demand forecasting, competitor price monitoring, and fleet utilization metrics into automated pricing engines [21]. Machine learning models are now being deployed to dynamically adjust rental rates based on real-time market signals, such as booking pace, pickup location, and vehicle availability [1]. In various other areas where revenue management [27] has historically been considered, such as airlines, electricity [11] and e-commerce [14], Reinforcement Learning methods are considered to require less data than revenue management methods [14, 4], experimentally shown to be generally more efficient than classical approaches [4, 32]. Early studies have successfully applied tabular Q-Learning in simplified settings [7], while more recent work has leveraged Deep Q-Networks (DQN) [22, 32, 14], Deep Deterministic Policy Gradient (DDPG) [12], and Soft Actor-Critic (SAC) [30] to handle continuous action spaces and high-dimensional state representations. These algorithms have shown strong empirical performance in both simulated and real-world retail markets [33, 7], enabling adaptive pricing that accounts for factors such as inventory levels, seasonality, and competitor behavior. Furthermore, dual-agent architectures have been proposed to jointly optimize pricing and inventory decisions in supply chain contexts [19]. In traditional revenue management sectors like airlines and hospitality, reinforcement learning is emerging as a viable alternative to classical optimization methods. Airlines have used RL to adjust fare class pricing under stochastic demand and cancellations [24, 32], aiming to maximize revenue over the booking horizon. In hospitality, RL enables dynamic room pricing based on demand signals, competition, and booking pace [9]. While recent studies show deep RL can outperform rule-based systems in accuracy and responsiveness, real-world adoption is limited by the complexity of state spaces, customer segmentation, and the need for interpretability [29]. With the use of various methods of Reinforcement Learning, the pricing dynamic problem can be modeled with the repeated game theory [13, 10] because the pricing dynamic problem is a repetition of sets where each set corresponds to the assignment of a price for each customer. Reinforcement Learning is also starting to be used in the car rental industry, where it is better adapted to the different constraints of the market [18]. In particular, it can consider particularities such as unfair pricing policies [16, 15], lack of data, or the fact that the customer base is not homogeneous [18, 15].

3 Model

Consider a car rental company that wants to dynamically choose the optimal price for any received car rental demand in the context of several car rental companies competing for that demand.

In a discrete-time model, where each step corresponds to a period (typically a day or half-day), there is a set $\mathcal{A}$ of car rental companies, each having a fixed quantity of cars to rent for each period. A customer's demand consists of a vehicle during a chosen sequence of periods. In this model, demands are issued sequentially (the set of demands is totally ordered). Each car rental company will propose a price for each demand (if it can satisfy it, considering its current free vehicles for the required period). The customer then chooses one of the proposals received according to specific criteria.

Thus, for each customer's demand, all rental companies having a vehicle available for the demanded sequence of periods are players in a price war modeled as a simultaneous game [8]. Thus, if the sequence of demands is taken into account, what must be analyzed is a repeated price war game under resource constraints, in which each step is modeled as a simultaneous game, and the objective of each player is to maximize their profit over time.

A rental **agent** $A_\alpha \in \mathcal{A}$ with $1 \leq \alpha \leq N$ is characterized by:

- a set of possible prices per day $Pr = \{pr_1 = MinPr, \ldots, pr_M = MaxPr\}$, taken in increasing order; note that in our context, all agents have the same set of prices
- the total number of cars at each period p denoted by $capmax_\alpha^p$,
- the number of cars already rented at this period use_α^p and the number of car available at this period is $cap_\alpha(p)$ with $capmax_\alpha(p) = use_\alpha(p) + cap_\alpha(p)$.

A rental **demand** d is characterized by the rental start period $start_d$ and the end period end_d of the rental with $end_d \geq start_d$.

A sequence of demands $\mathcal{D} = d_1, \ldots, d_N$ is considered, with the demands being received by the agents one after another in this order. The order is specifically determined based on the period res_d during which each demand is received, under the constraint that $res_d \leq start_d$.

3.1 Repeated game model

First, the competition for any demand d between the rental agents is defined as a simultaneous game [28] in which the player set is the agent set $\mathcal{A}$. The set of strategies for each such player $A_\alpha \in \mathcal{A}$ is the set $S_\alpha = Pr$. Note that a player A_α for whom there exists a period p, with $start_d \leq p \leq end_d$, such that $cap_\alpha(p) = 0$ does not participate in the game.

Thus, a strategy profile $\pi = s_1, s_2, ..., s_n$, where $s_\alpha \in S_\alpha$, is the price chosen for d by player A_α with $1 \leq \alpha \leq N$. Let's denote $S_{min} = \{\alpha \in [1, N], s_\alpha = min(\pi)\}$. Given such a profile, the expectation of gain for each player is 0 if the player A_α did not choose the minimum strategy of π otherwise it is the duration of d multiplied by the minimum strategy of π divided by the number of agents who did the minimum strategy of π.

$$\mathbb{E}g_\alpha(\pi, d) = \frac{\mathbb{1}_{\alpha \in S_{min}} \times (end_d - start_d) \times min(\pi)}{|S_{min}|} \tag{1}$$

This unit game is played for each received demand d. The capacity of the winner agent is then updated considering $start_d$ and end_d. In such a game, a pure Nash equilibrium consists of each player A_α such that $MinPr$ is the minimum overall price of players playing this price. In this dynamic pricing context, where demand changes at each new round of play, such a strategy will not prove optimal for profit maximization over all periods.

Since demands of an input set $\mathcal{D}$ are received in a given order, the competition consists of a sequence of simultaneous games. Each game impacts the player capacities of the next games in this sequence. Thus, this sequence of unit games is considered as a repeated game in which the consecutive unit games are linked by the capacities impact and in which the objective of each player is to maximize the sum of expectation of gains over time.

In the following, the notations for the unit game played for demand $d^t \in \mathcal{D}$ will be those given above with t as an exponent. Moreover, at each step t of the repeated game, there is a winner who must therefore update his available capacity cap_α^t. In terms of game knowledge, at each step $t \in \{1, ..., T\}$, each player A_α knows the set of received demands $\mathcal{D}^t = \{d^1, \ldots, d^t\}$ and $demC^t(A_\alpha) \subseteq \mathcal{D}^t$ the subset of demands for which he was chosen since step 1. For each period p, player A_α also knows his capacity $cap_\alpha^t(p)$ updated from $demC^t(A_\alpha)$.

A repeated game profile Π is a sequence of $|D|$ profiles $\pi^t = s_1^t, ..., s_N^t$, each being the profile of the unit game dedicated to demand d^t. The gain for each player A_α in the repeated game is $G_\alpha(\Pi, D) = \sum_{t=1}^{T} \mathbb{E} g_\alpha(\pi^t, d^t)$.

4 Pricing approaches to play repeated game

Attention is now directed toward the algorithmic approaches that will be employed by each player in the repeated game defined above to select a price for each new demand.

Thus, the goal is to maximize the sum of its profits during the T steps and on the ordered set of demands D. In the literature and practice in different fields of application of dynamic pricing, two main types of approaches are often used [6]. On the one hand, resource-based approaches are where prices are set according to the quantity of resources remaining for the requested periods [2, 17]. On the other hand, competition-based approaches are where prices are set according to the prices of the competition made in the past [3, 31]. In the following, three methods are defined: an operational research resource-based approach, a Reinforcement Learning competition-based approach, and finally, a new mixed approach.

4.1 A resource-based approach: EMSR

The resource-based dynamic pricing algorithm considered here is an adaptation in our repeated game model of the Expected Marginal Seat Revenue heuristic (EMSR, see [2]), a popular pricing method in different economic domains.

Each agent A_α owns a set of demands that have been the subject of a contract with this agent during a sequence of past periods P' (for example, a previous year's history). For each of these periods $p' \in P'$, the subset of demands d such that $start_d \leq p' \leq end_d$ is denoted by $Dem_\alpha(p')$, and the subset of those with a price equal to $pr_i \in Pr$ is denoted by $Dem_\alpha(p', pr_i) \subseteq Dem_\alpha(p')$.

Each price $pr_i \in Pr$ has a threshold defined for A_α by

$$thd_{\alpha,i} = \frac{1}{P'} \sum_{p' \in P'} \frac{|Dem_\alpha(p', pr_i)|}{|Dem_\alpha(p')|} \tag{2}$$

The price offered to the demand d by A_α using EMSR is the price defined as follows: for each period p such that $start_d \leq p \leq end_d$, consider the maximum price over all prices pr_{i_p} such that: $\frac{cap_\alpha^{res_d}(p)}{capmax_\alpha(p)} \leq thd_{\alpha,i_p}$.

4.2 A competition-based approach

A method based on competition is now defined, using an increasingly popular machine learning technique called Reinforcement Learning (RL). RL algorithms are based on an agent interacting with a potentially evolving and partially observable environment to try to maximize rewards at each time. Through these interactions, the agent attempts to maximize its reward by realizing a trade-off between exploring new actions and exploiting those that seem optimal [26].

In our context, each RL algorithm is executed by a rental agent (named RL agent). Each learning step corresponds to a step of the repeated game, i.e., to a new demand and, therefore, to new prices proposed by the competitors of the RL agent. The environment, consisting thus of the demand and the competitor prices, is therefore strongly non-stationary. This is why it seems difficult to use efficiently a state-based Reinforcement Learning method like the Q-Learning [26].

The principle of the RL algorithm that is used here is based on the Linear Reward Inaction method (LRI, [20]). The agent's strategies are its set of prices Pr, which is associated with a stochastic vector V. At each step t, the proposed price is chosen randomly according to the state of the vector at this step V^t, then V^{t+1} updates it according to a utility U^t defined as follows:

- $V^{t+1}[s] \leftarrow V^t[s] + b \times U^t \times (1 - V^t[s])$ If $s = s^t$
- $V^{t+1}[s] \leftarrow V^t[s] - b \times U^t \times V^t[s]$ Else

With $b \leq 1$ a chosen learning rate parameter. The utility function U^t is here defined by

$$U^t = \begin{cases} \exp(\frac{-(Pr_{min}-s)^2}{100}) \text{ if the agent is chosen} \\ \exp(\frac{Pr_{min}-s}{10}) \qquad \text{else} \end{cases} \tag{3}$$

With Pr_{min} the minimum price is done by the competitors.

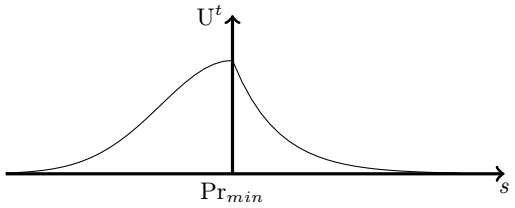

Fig. 1. Utility calculation

This utility function first favors the demand gain, then the utility function looks at the proximity of the price proposed by our agent to the minimum of those proposed by competitors.

Finally, to allow the stochastic vector to evolve at any moment during the repeated game, the probability of each price within the stochastic vector V^t is always greater than or equal to a certain fixed probability v_{min}.

4.3 Mixed approach

Here, a pricing method is defined, attempting to combine the principles of a resource-based approach and a competition-based approach, as described in the previous sections. For this, each demand receives d at step $t = res_d$ and each agent A_α three metrics that are a function of the sequence of current capacity states of the periods between $start_d$ and end_d, i.e. by considering the state of the resources in a two-dimensional way. These metrics need a large enough interval to give an idea of the system's state, so we'll look at them throughout at least the average duration of the previous year's demands. This period will be centered on the demand. These three metrics are defined as follows:

- The pressure $\rho_\alpha(d) = \frac{FreeArea}{TotalArea} = \frac{\sum_{p=start_d}^{end_d} cap_\alpha^{res_d}(p)}{\sum_{p=start_d}^{end_d} capmax_\alpha(p)}$.

- The nuisance $\eta_\alpha(d) = \frac{PeakArea}{FreeArea} = \frac{\sum_{p=start_d}^{end_d} |capmax_\alpha(p) - Peak_\alpha(d)|^+}{\sum_{p=start_d}^{end_d} cap_\alpha^{res_d}(p)}$

  with $Peak_\alpha(d) = \max_{start_d \leq p \leq end_d} use_\alpha^{res_d}(p)$

These two parameters measure the evolution of different areas defined by the evolution of $cap_\alpha^{res_d}(p) = capmax_\alpha(p)$ - $use_\alpha^{res_d}(p)$ and $capmax_\alpha(p)$ between time $start_d$ and end_d, as illustrated by Figure 2, 3 and 4. Pressure represents the average capacity available over the period sequence affected by the demand, and

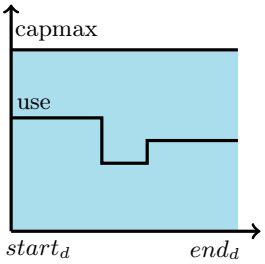

Fig. 2. Total Area

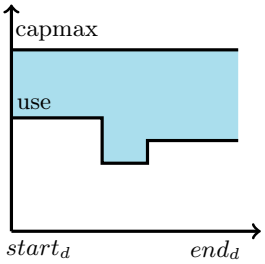

Fig. 3. Free Area

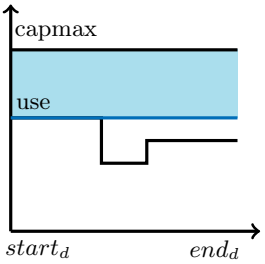

Fig. 4. Peak Area

nuisance measures the impact of the period with the least availability on this sequence. Thus, the pressure shows the filling state of the system, and the nuisance indicates the presence or absence of peaks that are detrimental to optimal filling. These new measures will provide three learning situations parameterized by τ_ρ and τ_η as shown in Figure 5, where each demand is plotted according to the pressure and nuisance of the system when it was received. Situation 1 corresponds to a rather empty system's filling, while situations 2 and 3 correspond to a more complete filling. The difference between these two situations is that in situation 2 the filling is fairly uniform, whereas in situation 3 there are large filling peaks.

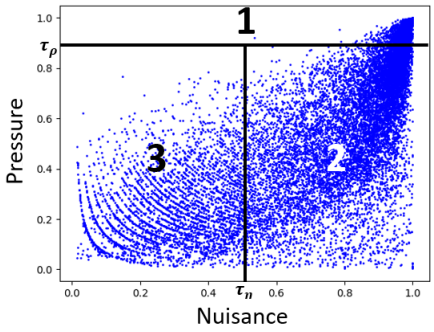

Fig. 5. Situation diagram according to nuisance value and pressure: each point corresponds to the nuisance and pressure values of one demand

For each of these three situations, a specific RL algorithm defined in Section 3.2 is executed based on the temporal state of the resource.

Note that an approach based on 2 times 3 situations has also been considered, considering a threshold value for $start_d - res_d$. The relevance of such a threshold could arise from the analysis of a data history presented in Section 4 (see Figure 7). Such an approach has not yet shown sufficient performance to be retained.

5 Experimental results

This section presents the experimental evaluation of the pricing methods defined in Section 3. First, real-world data collected from a car rental company are analyzed (in Section 4.1) to identify patterns and correlations among demand characteristics, such as rental duration, advance booking time, revenue per day, and utilization rate. Then, the experimental setup is defined in Section 4.2. Section 4.3 describes the evaluation metrics and test scenarios. Finally, the performance of the different approaches is compared in Section 4.4.

5.1 Real data analysis

The dataset used for the performance evaluation of the proposed dynamic pricing approaches comes from the past activity of a real world rental agent, a customer of the WeYield company, located on a French island. This company has a fleet of about 700 cars and faces many competitors of similar or larger sizes. The sequence of demands that serves for the experiments consists of three years of contracts, whose prices have not been negotiated in advance (called yieldable demands); note that demands received but not contracted are not known. The number of demands thus considered is about 13,000 per year. Each demand d has the values of res_d, $start_d$, and end_d, as well as the final price paid by the customer.

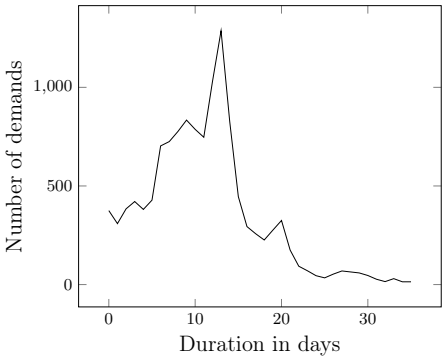

Fig. 6. Distribution of demand duration

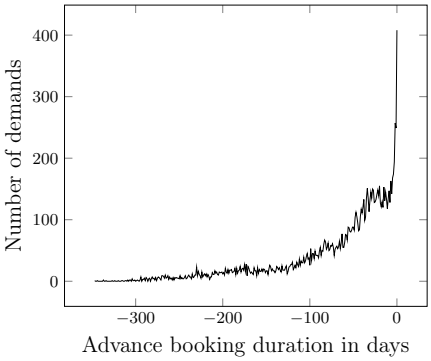

Fig. 7. Distribution of demands advance booking duration

These collected data are first analyzed by focusing on possible correlations between the negotiated prices and the values of demand characteristics. Contracted demands are mainly leisure demands with a duration between 7 and 14 days (in Figure 6). The average duration is 11.27 days. The average advance booking (i.e. $res_d - start_d$) duration is -70,48 days. This average is quite high

for car rentals, but that's due to the leisure clientele for a long-haul destination. These values can vary considerably from one rental company to another. Moreover, the car rental business on this French island is heavily impacted by seasonality, as is the case in most car rental locations. The high season is between July and December, with a spike during the fall vacation. In Figure 8, the number of checkouts per month is displayed with a color gradient according to the duration of the advance booking. So, in January, demands have a short advance booking duration, mostly less than a month, while in July, their advance booking duration may be zero days or six months. This is caused by the fact that July is the start of the high season, so customers book in advance, whereas January is the new year and the low season.

Possible correlations between the different demand characteristics: rental duration, advance booking duration, revenue per day (RPD), and utilization rate (defined as the maximum ratio between the capacity already used *use* and the total capacity *capmax* at period res_d) — are now being studied.

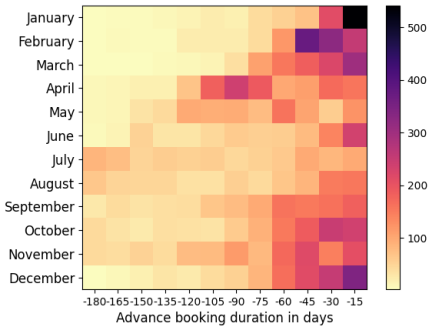

Fig. 8. Correlation between advance booking and rental start month

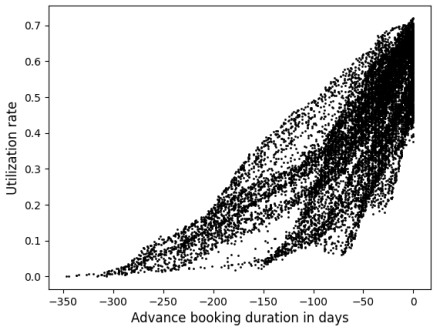

Fig. 9. Correlation between advance booking duration and utilization rate

In Figure 9, each point corresponds to a single demand located at the intersection between its advance booking duration and its maximum utilization rate. Thus, a strong correlation between utilization rate and advance booking duration can be seen, which is found in all rental companies. The correlation curves between RPD and the other characteristics show no correlation. To conclude this study, demands were clustered according to each characteristic considered as a function of the others. Each clustering step was performed using the OClustR algorithm [23]. This algorithm is a graph-based clustering algorithm for building overlapping clusters. It considers the complete graph, with demands as its vertices and aretes as its weight, the normalized difference of the characteristic we're looking at. Cluster overlapping allows demands to be in several clusters at the same time, which is important because several clusters can have similar behaviors and, therefore, have common demands. Each clustering is performed

according to one characteristic, and then the distribution of the other characteristics in each cluster is compared to the global distribution. No link between RPD and the other characteristics appears clearly, except for clustering based on the utilization rate, which shows a link between the advance booking duration, the utilization rate, and the duration.

5.2 Instance definition for experimentation

To carry out the performance evaluation of the algorithms defined in Section 4, the sequence of demands defined above is ordered by reservation dates res_d. These demands are those received in 2023, but they can have their rental period in 2023 and 2024. Demands received in 2022 will be considered as a bootstrapping year for the learning methods and as historic data for EMSR. Demands of 2022 will also impact the remaining capacities of periods during 2023. It would not be realistic to consider the start of the year (2023) without any reservations already made the previous year. Pricing performances are evaluated on demands received in 2023.

In each experiment, the two competitive agents A_1 and A_2 will have the same number of vehicles $capmax_i(p) = 150$, $i \in \{1, 2\}$ for each period p.

5.3 Measured parameters and scenarios

To evaluate and compare the performance of the different dynamic pricing methods defined above, the main metric used is the average total profit of each competing player, calculated over 20 runs on the dataset defined in Section 4.1. The performance evaluation of the algorithms will also be based on other metrics obtained for a single run and defined in the next sections.

Three repeated game scenarios are considered between two players having the same set of possible prices per day $P_r = \{15, 20, 25, 30, 35, 40, 45\}$, which were chosen considering existing prices in the dataset. Each scenario runs a specific algorithm: RL approach VS EMSR (to contrast a resource approach and a competitive approach), mixed approach VS RL approach, and mixed approach VS EMSR.

The input parameters of the used Reinforcement Learning algorithms are experimentally set as follows: for the RL approach, $b = 0.001$ and $v_{min} = 0.001$ and for the mixed method $b = 0.005$, $v_{min} = 0.005$, $\tau_\rho = 0.9$ and $\tau_\eta = 0.5$. The value of b was chosen experimentally by taking the value giving the best benefit between 0.1 et 0.0001. Then the value of v_{min} was chosen according to that of b.τ_ρ and τ_η were tested and on this data set, we took those with the best result.

5.4 Experimental results

The first experiment that is carried out is an evaluation of the RL method's performance, in comparison with EMSR. Figure 10 gives the average cumulative profit value over time, considering 20 executions for each of the two players. The

RL method performs better from the first to about the 10000^{th} demand, and then the performances reverse. This transition is a consequence of the seasonality phenomenon defined in Section 4.1. Indeed, after $8,000$ demands, the available capacity of the RL player is zero for the autumn holiday booking period, unlike the capacity of the EMSR player. Since there are demands for this period between the 8000^{th} and 10000^{th} demand, EMSR is the only one to accept reservations. This can be seen in Figure 11, which shows the non-availability ratio for each sequence of 100 demands. Around the $8,000^{th}$ demand, the number of refusals due to lack of capacity increases considerably for the RL method, in contrast to the EMSR method, where this number is around the 10000^{th} demand.

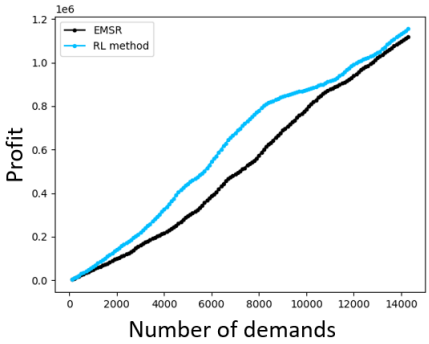

Fig. 10. RL method VS EMSR

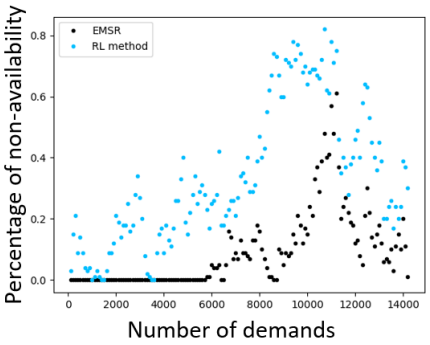

Fig. 11. Evolution of availability during demands

To illustrate this behavior, if the available capacity $capmax_i(p)$ of each player $i \in \{1, 2\}$ is increased so that each player can accept all the demands received during the considered year, then the cumulated profit of the RL method is significantly higher than the one of EMSR because the RL method undercuts the market (it captures the demands by proposing the lowest prices) and therefore obtains more demands than EMSR whose prices only evolve according to the utilization rate. On the other hand, if $capmax_i(p)$ is divided by 3 compared to its initial value, which creates a heavily constrained environment, then the profit of the RL method is lower than the profit of EMSR because the RL method fills its capacity too quickly with too low prices.

To evaluate the efficiency of the proposed mixed method, it first competes against the RL method. As illustrated in Figure 12, both methods have almost the same total profit since they run similar algorithms. In this case, learning vectors of the two methods evolve similarly over time.

When making the mixed method compete with EMSR, Figure 13 shows that the mixed method is more cost-effective than EMSR, with better performance than RL against EMSR, because, with its three situations (as explained in figure 5), the mixed method wins demands at a higher price than the RL method in

situations 2 and 3 which are situations where part of the agent's capacity is already used. In these situations, the prices chosen by this agent thus remain mostly lower or at the same level as the prices of the EMSR agent prices due to the fact that EMSR has also used almost all of its available capacity considering one random execution.

For the mixed method, the learning vector of situations 1 and 2 converges to a unit vector, unlike situation 3 (see Figure 16). Indeed, for situation 1 of the mixed method, the capacities of the EMSR actor are not very full, which implies that EMSR only offers prices of 15 and 20. The mixed method thus learns to choose a price of 15 to win all demands. For demands in situation 2 of the mixed method, the capacities of the EMSR actor are less free but still not very full, so the mixed method learns to choose the price of 20 as the price because it is lower than the average price, equal to 24.6, selected by EMSR, and because there are few prices chosen equal to 15. The behavior is very different in situation 3 as can be seen in Figure 16 because the prices chosen by EMSR are much more variable and higher since EMSR's capacities are fuller. Overall, the use of these three learning vectors (one per situation) that learn differently makes the mixed method outperform the RL method against EMSR. In Figure 14, very few of the demands are in situation 1, given the parameters τ_ρ and τ_η, but it is necessary to have learning vectors that learn higher prices in situations 2 and 3. The mixed method is, therefore, more profitable than the competition method and the resource method when they compete.

Situation 3, therefore, turns out to be the situation where both price and resource competition are confronted. This is reflected in the evolution of the learning vectors over time. While in situations 1 and 2, a price is very quickly selected with a probability that immediately converges to a value close to 1, Figure 16 shows that learning is slower in situation 3, which does not prevent it from learning high probability values for the three selected prices. Figure 16 includes the bootstrapping year before the vertical line to show the learning speed.

Moreover, Figure 14 shows that the start dates of the demands are distributed homogeneously over time from the beginning of the year as they occur, since the curves of the three situations increase in parallel and regularly, notably up to the 8000^{th} demand, and not consecutively. Figure 14 and Figure 15 show that when the mixed method is against the RL method or EMSR, filling does not take place in the same way, as there are significantly more demands in situation 3, which is a situation where the system is full when the mixed method is used against EMSR. This behavior change is due to the fact that on the one hand, the competitor has a resource-based approach, while on the other, the approach is based on competition. Both the competition-based and mixed methods tend to lower their prices sharply to meet demand. In the scenario shown in figure 15, there are two methods that lower their prices, so they get half as many demands, which means there are more low-pressure, low-nuisance demands. Given that the rate τ_ρ is quite high, some low-pressure, low-nuisance demands are in situation 2.

Figure 17 shows that there are many demands with a pressure of around 0.65 for EMSR. This is due to the fact that for these applications, the EMSR price is 25, while the mixed method is still in situation 1 or 2, giving a price of 15 or 20. So for these demands, EMSR loses and the situation remains the same for subsequent demands until the mixed method moves into situation 3.

These different experiments show that the mixed method can compete effectively with both a price competition-based method like RL and a resource-based method like EMSR. These various experiments have also been carried out on data from other rental companies and have produced similar results, which are favorable to the mixed method.

Finally, the efficiency of the mixed method is strongly dependent on the values of the parameters τ_ρ and τ_η which were experimentally fixed in this study.

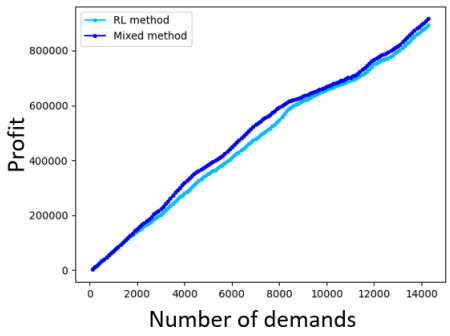

Fig. 12. Mixed method VS RL method

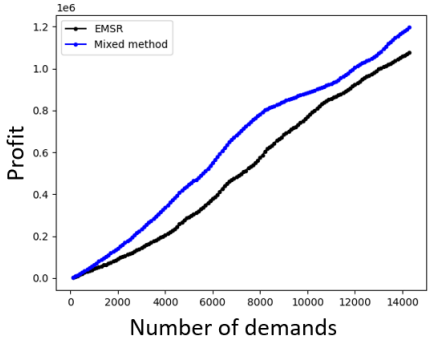

Fig. 13. Mixed method vs EMSR

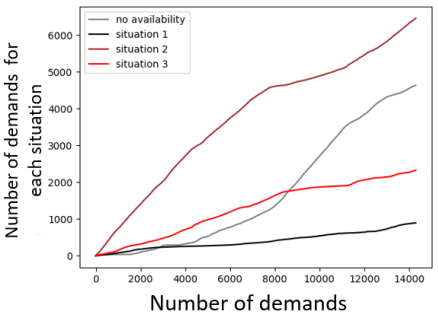

Fig. 14. Total number of demands by situation for the mixed method against EMSR

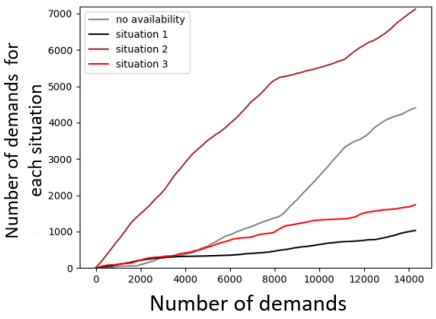

Fig. 15. Total number of demands by situation for the mixed method against the RL method

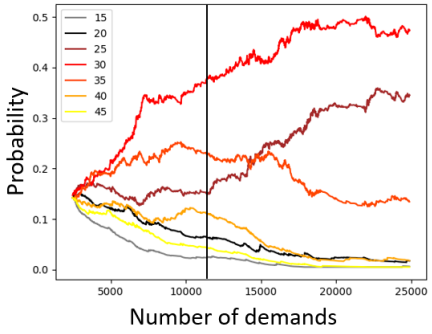

Fig. 16. Evolution of the learning vector for the situation 3

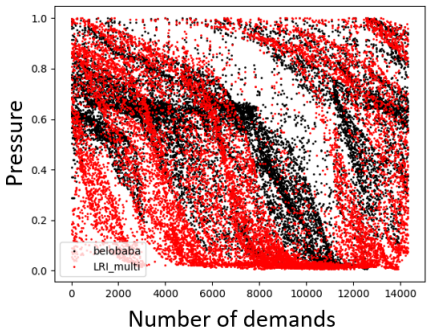

Fig. 17. The pressure of each demand for both methods: EMSR and the mixed method

6 Conclusion

This paper presents a Reinforcement Learning method for dynamic pricing with resource constraints in a competitive situation, a method that can be effective against both a method based on price competition and a method oriented towards resource evolution. Building on a classic reinforcement learning (RL) approach inspired by [20] values, this method stands out for its integration of resource state considerations, its unique utility function, and its ability to account for diverse situations, each leading to a distinct learning process. The real data used for performance evaluations show the impact on the behavior of the methods of demands seasonality phenomena often occurring in car rental context. The objective now is to integrate history-based seasonality prediction into a reinforcement learning method. Finally, the dynamic pricing approach proposed in this paper applies to a context with both resource constraints and price wars, modeled by a repeated game. Such an approach can therefore concern many other areas of dynamic pricing (transport, hotels, etc.) whose specific characteristics would influence the choice of the RL method retained.

References

1. Alabi M.: Data-Driven Pricing Optimization: Using Machine Learning to Dynamically Adjust Prices Based on Market Conditions, 2024.
2. Belobaba P., Odoni A., Barnhart C.: The Global Airline Industry. John Wiley & Sons, 2015.
3. Betancourt J. M., Hortaçsu A., Öry A., Williams K. R.: Dynamic price competition with capacity constraints. NBER Working Paper 32673, 2024.
4. Bondoux N., Nguyen A. Q., Fiig T., Acuna-Agost R.: Reinforcement learning applied to airline revenue management. Journal of Revenue and Pricing Management, 1–17, 2020.
5. Cross R. G.: Revenue Management: Hard-Core Tactics for Market Domination. Currency, 1997.
6. Deksnyte I., Lydeka Z.: Dynamic Pricing and Its Forming Factors. International Journal of Business and Social Science, Vol. 3 No. 23, 2012.
7. Ferrara M., Brandimarte P.: A reinforcement learning approach to dynamic pricing, 2018.
8. Fudenberg D., Levine D.: The Theory of Learning in Games, Cambridge: MIT Press, 1999.
9. Gallego, G., Topaloglu, H.: Revenue Management and Pricing Analytics, 2019.
10. Ghasemi S., Meybodi M. R., Takht-Fooladi M. D., and Rahmani A. M.: Dynamic Pricing of Applications in Cloud Marketplaces using Game Theory. Arxiv, 2023.
11. Ghasemkhani A., Yang L.: Reinforcement learning-based pricing for demand response,2018 IEEE International Conference on Communications Workshops, 2018.
12. Hou L., Li Y., Yan J., Wang C., Wang L., Wang B.: Multi-agent reinforcement mechanism design for dynamic pricing-based demand response in charging network, International Journal of Electrical Power and Energy Systems, 2023
13. Hu Y., Han C., Li H., Guo T.. Modeling opponent learning in multiagent repeated games. Appl Intell 53, 17194–17210, 2023.
14. Kastius A., Schlosser R.: Dynamic pricing under competition using reinforcement learning. Journal of Revenue and Pricing Management, 2022. https://doi.org/10.1057/s41272-021-00285-3
15. Khan R., Singh V., Zhu T.: Price Discrimination and Competition in the Auto Rental Industry, 2009.
16. Maestre R., Duque J., Rubio A., Arévalo J.: Reinforcement learning for fair dynamic pricing, 2018.
17. Mihailescu M., Teo Y. M.: Dynamic Resource Pricing on Federated Clouds. 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010.
18. Mobiag: Fundamentals of dynamic pricing model for car sharing and car rental businesses, 2017.
19. Mohan A., Soni T.K., Vamshisai P., Sateesh B.: A deep Q network framework on stock price prediction, International Journal of Health Sciences, 3050–3061, 2022.
20. Narendra K. S., Thathachar M. A. L.: Learning Automata: An Introduction, 1989.
21. Oliveira B. B., Carravilla M. A., Oliveira J.F.: Integrating pricing and capacity decisions in car rental: A matheuristic approach. Operations Research Perspectives, 334-356, 2018.
22. Paudel D., Das T. K.: Multi-agent Deep Reinforcement Learning for Dynamic Pricing by Fast-Charging Electric Vehicle Hubs in Competition. Arxiv, 2024.
23. Pérez-Suárez A., Martínez-Trinidad J. F., Carrasco-Ochoa J. A., Medina-Pagola J. E.: OClustR: A new graph-based algorithm for overlapping clustering. Neurocomputing, 234-247, 2013.

24. Shihab S.A.M., Wei P.: A deep reinforcement learning approach to seat inventory control for airline revenue management. J Revenue Pricing Manag 21, 183–199, 2022.
25. Strauss A. K., Klein R., Steinhardt C.: A review of the choice-based revenue management: Theory and methods. European Journal of Operational Research, 375–387, 2018.
26. Sutton R. S., Barto A. G.: Reinforcement Learning: An Introduction. Second Edition, MIT Press, Cambridge, MA, 2018.
27. Talluri K. T., Ryzin G. J.: The theory and practice of revenue management. Springer, 2004.
28. Truong-Huu T., Tham C. K.: A Game-Theoretic Model for Dynamic Pricing and Competition among Cloud Providers. IEEE/ACM 6th International Conference on Utility and Cloud Computing, 2013.
29. Tuncay G., Kaya K., Yılmaz Y., Yaslan Y., Gündüz Öğüdücü Ş.: A reinforcement learning based dynamic room pricing model for hotel industry, 2023.
30. Yavuz T., Kaya O.: Deep reinforcement learning algorithms for dynamic pricing and inventory management of perishable products, Applied Soft Computing, 2024.
31. Zhai Y., Zhao Q.: Oligopoly dynamic pricing: A repeated game with incomplete information. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016.
32. Zhu X., Jian L., Chen X., Zhao Q.: Reinforcement learning for Multi-Flight Dynamic Pricing. Computers & Industrial Engineering, 2024.
33. Zhydik O.: E-commerce Tech Trends: Reinforcement Learning for Dynamic Pricing, eleks, 2024.