Pricing in Dynamic Advance Reservation Games

Eran Simhon, Carrie Cramer, Zachary Lister and David Starobinski
College of Engineering, Boston University, Boston, MA 02215

***Abstract*—We analyze the dynamics of *advance reservation (AR) games*: games in which customers compete for limited resources and can reserve resources for a fee. We introduce and analyze two different learning models. In the first model, called *strategy-learning*, customers are informed of the strategy adopted in the previous iteration, while in the second model, called *action-learning*, customers estimate the strategy by observing previous actions. We prove that in the strategy-learning model, convergence to equilibrium is guaranteed. In contrast, in the action-learning model, the system converges only if an equilibrium in which none of the customers makes AR exists. Based on those results, we show that if the provider is *risk-averse* and sets the AR fee low enough, action-learning yields on average greater profit than strategy-learning. However, if the provider is *risk-taking* and sets a high AR fee, action-learning provably yields zero profit in the long term in contrast to strategy-learning.**

I. Introduction

Modern resource management software, such as Haizea[1] and IBM Platform Computing Solutions[2] include mechanisms that allow customers to reserve resources in advance. In these packages, an administrator can define an AR pricing scheme and may charge a reservation fee. Charging an AR fee is common in different venues and can be also found in cloud computing. For example, Amazon EC2 cloud offers *reserved instance* services, in which customers pay a fee that allow them to use resources later on for a lower cost.[3]

The strategic behavior of customers in such systems is studied in [1], which introduces the concept of Advance Reservation (AR) games. In AR games, customers know that they will need service at a future time slot. In order to increase the probability to get service when needed (referred to as the *service probability*), customers can make a reservation in advance for a fee, which is set by the provider.

Upon deciding whether to reserve a resource in advance or not, estimating the decision of other customers plays a key role. In this paper, we assume that such estimation can be made by observing historical data. Our goal is to understand how revealing different types of data impacts the profit of a service provider. Toward this end, we study the dynamics of AR games by analyzing several learning models. We assume that the game repeats many times and customers modify their strategy following the observation of the behavior of customers at the previous iteration.

The number and types of equilibria in AR games depend on the AR fee [1]. Low fees lead to a unique equilibrium and guaranteed AR fee profit. However, in many cases, the profit-maximizing fee is higher and may result in other equilibria, including one that yields zero profit. Thus, we distinguish between a *risk-averse provider*, who opts for a guaranteed profit albeit typically a non-optimal one, and a *risk-taking provider*, who wishes to maximize its profit and is willing to take the risk of ending up with zero profit.[4]

We define two types of learning: *strategy-learning* and *action-learning*. In strategy-learning, customers obtain information about the strategy adopted in the previous iteration, while in action-learning, customers estimate the strategy by obtaining information about the actions taken by customers in the previous iteration.

We separately analyze the dynamics of systems formed by a risk-averse provider (a system with a unique equilibrium) and by a risk-taking provider (a system with multiple equilibria). For each system, we analyze the outcomes of both strategy-learning and action-learning. The main results regarding convergence to equilibrium are: 1) When implementing strategy-learning, the customers' strategy always converges to an equilibrium. If multiple equilibria exist, the initial belief of customers determines to which equilibrium the strategy will converge. 2) When implementing action-learning, if there exists an equilibrium in which none of the customers makes AR (*none-make-AR equilibrium*), then the strategy converges to this equilibrium. Otherwise, the strategy cycles. Hence, a system formed by a risk-averse provider always cycles, while a system formed by a risk-taking provider always converges to a *none-make-AR equilibrium*. Furthermore, we show that in order to maximize profits, a risk-averse provider should reveal previous actions rather than strategies.

The rest of the paper is structured as follows: In Section II, we review related work. In Section III, we describe the AR game and provide necessary background. In section IV, we define and analyze the learning process. In Sections V and VI, we study the dynamics of systems with risk-averse pricing and risk-taking pricing, respectively. Section VII concludes the paper and suggests directions for future research.

II. Related Work

The concept of learning an equilibrium is rooted in Cournot's duopoly model [2] and has been extensively researched. Most works analyze fixed-player games (i.e., the same players participate at each iteration), see, [3], [4] and [5]. However, in practice the number of players may vary over

[1]See http://haizea.cs.uchicago.edu/.
[2]See http://www.redbooks.ibm.com/redbooks/pdfs/sg248073.pdf.
[3]See http://aws.amazon.com/ec2/purchasing-options/reserved-instances/.

[4]Note that, given a demand, the number of customers getting service is independent of the number of customers making AR. Hence, profits from service fees are ignored in this paper.

time. In our work, we assume that the number of customers at each iteration is a random variable.

Learning under stochastic settings was researched in [6]. In that paper, customers choose between buying a product at full price or waiting for a discount period. Decisions are made based on observing past capacities. In [7], the authors analyze a processor sharing model. In this model, customers choose between joining or balking by observing the performance history. In [8], the authors present a model of abandonment from unobservable queues. The decision is based on the expected waiting time which is formed through accumulated experience. To our knowledge, a learning model of a queueing system that supports advance reservation has not been studied yet.

Different learning models are distinguished by their learning type. While Cournot [6] assumes that decisions are made by observing the opponent's most recent action, the author of [9] assume that at each iteration, players know the actions made in all previous steps. This approach is known as *fictitious play*. In [10], the authors assume that players only observe their own payoffs and from those they deduce the actions of other players. In our work, we adopt Cournot learning model in the sense that customers only observe the behavior of the previous iteration.

III. Advance Reservation model

In this section we describe the AR game and summarize the main results presented in [1] that are relevant to this paper. We begin by describing the model assumptions.[5]

The system consists of N servers. The service time axis is slotted. The *demand*, which represents the number of customers that request service in a specific slot (each customer requests one server), is an independent random variable D with general distribution (supported in $\mathbb{N}$).[6] The lead time of a customer is the time between arrival and the slot starting time. All lead times are i.i.d continuous random variables with the same general distribution (supported in $\mathbb{R}^+$). Each customer chooses an action: make AR or not make AR, denoted AR and AR' respectively. If the demand is larger than N but not all servers are reserved, the unreserved servers are arbitrarily allocated to customers that did not make AR. The customers know the number of servers N and statistical information on the system (i.e., the distribution of the demand and the lead time). The provider charges customers that make AR and get service a fixed reservation fee C. All the customers have the same utility U from the service. Without loss of generality, we set $U = 1$.

We denote the value of the cumulative distribution of a customer's lead time as her *normalized lead time* (which is also the expected fraction of customers arriving after her). A strategy function $\sigma\colon \tau \to [0, 1]$ is defined as the probability that a customer with normalized lead time $\tau \in [0, 1]$ makes AR. Given that all customers follow strategy σ, the expected payoff of making AR for a tagged customer with normalized lead time τ is

$$U_\sigma(\tau, AR) = (1 - C)\mathbb{P}\left(S|\sigma, \tau, AR\right), \tag{1}$$

where $\mathbb{P}(S)$ is the service probability. The expected payoff of not making AR is

$$U_\sigma(\tau, AR') = \mathbb{P}\left(S|\sigma, AR'\right). \tag{2}$$

The service probability in both cases is calculated by conditioning on the number of customers and the lead times. A formula is given in [1].

Given a strategy σ and a normalized lead time τ, one can find the best response of a tagged customer (i.e, the decision that maximizes her expected payoff) by comparing the two possible expected payoffs:

$$BR(\sigma, \tau) = \underset{\alpha \in \{AR, AR'\}}{\operatorname{argmax}} \left(U_\sigma(\tau, \alpha)\right). \tag{3}$$

Next, we define a threshold strategy $p_e \in [0, 1]$ as a strategy in which only customers with normalized lead times greater than p_e make AR. Under the assumption that all customers follow the threshold strategy p_e, the service probabilities of the *threshold customer* (i.e., a customer that arrives exactly at the threshold) when making and not making AR are defined by $\pi_{AR}(p_e)$ and $\pi_{AR'}(p_e)$, respectively. Both probabilities are non-decreasing function of p_e.

AR games have two types of equilibria. The first type is *some-make-AR equilibrium*. A threshold strategy p_e leads to a *some-make-AR equilibrium* if and only if $p_e \in (0, 1)$ and the threshold customer is indifferent between the two strategies:

$$(1 - C)\,\pi_{AR'}(p_e) = \pi_{AR}(p_e). \tag{4}$$

The second type of equilibrium is a *none-make-AR equilibrium*.[7] A threshold strategy p_e leads to a *none-make-AR equilibrium* if and only if $p_e = 1$ and

$$(1 - C)\,\pi_{AR'}(1) \geq \pi_{AR}(1). \tag{5}$$

We define the ordered set of equilibria of a game with n equilibria by $P^e = \{p^e_1, ..., p^e_n\}$ where $p^e_{i+1} > p^e_i$. If *none-make-AR* is an equilibrium, then $p^e_n = 1$.

Throughout the paper we make the following assumption:

Assumption 1. *Customers that are indifferent between making and not making AR opt not to make AR.*

IV. Learning Model

In this section we define the learning model and analyze the behavior of customers at a specific iteration. We assume that at the first iteration, customers base their decisions on certain information they share regarding the strategy of other customers. We refer to that information as the *initial belief*.

Lemma 1. *Given any common initial belief, the set of best responses of all customers to that belief is a threshold strategy.*

[5] In [1] three different models of AR games are presented. In this paper we develop learning models of the first AR game.

[6] This is a generalization of [1], which assumes that the demand follows a Poisson distribution.

[7] It is shown in [1] that an equilibrium in which all customers make AR does not exist.

Proof: Since the servers are allocated in a first-reserve-first-allocated fashion, the expected payoff of making AR is a non-decreasing function of the lead time. On the other hand, when not making AR, the expected payoff as a function of the lead time is a constant. In case that the two expected payoff functions do not intersect, then all customers will either make AR or not make AR (depending on which function is greater). In the case that they do intersect, they can intersect on a single point γ or along an interval $[\gamma_1, \gamma_2]$. In the first case, the best response of a customer with lead time smaller than γ is not to make AR, while the best response of a customer with greater lead time is to make AR. Hence, all customers will follow the threshold strategy γ. In the second case, the best response of a customer with lead time smaller than γ_1 is not to make AR. Based on Assumption 1, we deduce that a customer with lead time within the interval $[\gamma_1, \gamma_2]$ also does not make AR. The best response of a customer with lead time greater than γ_2 is to make AR. Thus, all customers will follow the threshold strategy γ_2. ■

Next we make the following assumption:

Assumption 2. *The initial belief β_1 is that all customers follow the same threshold strategy.*

The assumption is crucial for analyzing the strategy-learning model when the provider is risk-taking. In this case, the initial belief determines to which equilibrium the strategy converges. Without this assumption, any arbitrary initial belief will require a separate analysis.

We denote the threshold strategy followed at iteration i by $p_i \in [0,1]$. An estimator for strategy p_i is denoted $\hat{p}_i$. Our learning rule is based on a Cournot model. In this model, the players believe that the strategy followed at the last iteration will also be followed in the current iteration. Hence, at iteration $i > 1$, the belief β_i is equal to $\hat{p}_{i-1}$. In the strategy-learning model, customers observe the previous strategy, i.e., at iteration i, $\hat{p}_i = p_i$. In the action-learning model, customers observe the fraction of customers that did not make AR and use that fraction as an estimator for the strategy. More formally, given D_i and $D_i^{AR'}$, which are respectively the demand and the number of customers that did not make AR at iteration i, then the estimator of p_i is $\hat{p}_i = D_i^{AR'}/D_i$. If at some iteration the demand is zero, no learning is being done and we assume that the belief remains the same as in the previous iteration, i.e., if at iteration i the demand $D_i = 0$, then $\beta_{i+1} = \beta_i$.

Since all customers follow a threshold strategy in each iteration, we redefine the best response function $\overline{BR} : [0,1] \to [0,1]$. The input of the new best response function is a belief regarding the threshold strategy that is followed by all customers. The output is the best response threshold strategy to that belief. Given β, the output p is the single value that satisfies the following:

- For any normalized lead time $\tau \geq p$

$$U_\beta(\tau, AR) \geq U_\beta(AR'). \tag{6}$$

- For any normalized lead time $\tau < p$

$$U_\beta(\tau, AR) \leq U_\beta(AR'). \tag{7}$$

In the next subsection, we analyze the response of customers to any given belief, regardless of how this belief has been established.

A. Learning Analysis

We begin the analysis with the following key observations:

1) Given a belief β, the service probability of a customer with normalized lead time $\tau > \beta$ that makes AR is only affected by the decisions of customers that arrived earlier than her. Therefore, her service probability is the same as a threshold customer in a system with threshold strategy τ. Hence,

$$P\left(S|\beta, \tau, AR\right) = \pi_{AR}(\tau) \quad \text{if } \tau \geq \beta. \tag{8}$$

2) Given a belief β, a customer with normalized lead time $\tau < \beta$ that makes AR believes that she is the only one deviating, and therefore, she has the same service probability as the threshold customer. Hence,

$$P\left(S|\beta, \tau, AR\right) = \pi_{AR}(\beta) \quad \text{if } \tau < \beta. \tag{9}$$

Based on these observations, we conclude that under the belief β, the expected payoff of making AR for a customer with normalized lead time τ is:

$$U_\beta(\tau, AR) = \begin{cases} (1-C)\,\pi_{AR}(\beta) & \text{if } \tau \leq \beta \\ (1-C)\,\pi_{AR}(\tau) & \text{if } \tau > \beta. \end{cases} \tag{10}$$

The expected payoffs of all customers that do not make AR are equal. Hence,

$$U_\beta(\tau, AR') = \pi_{AR'}(\beta). \tag{11}$$

Next, we show that under the belief β, if the threshold customer is better off not making AR, then the best response strategy to β is in between β and the smallest equilibrium that is greater than β.

Lemma 2. *Given a belief β, if*

$$(1-C)\pi_{AR}(\beta) < \pi_{AR'}(\beta), \tag{12}$$

then the best response strategy is in the interval $(\beta, p_m^{\mathrm{e}}]$, where $m = \min\{j : p_j^{\mathrm{e}} \geq \beta\}$.

Proof: First, we show that the best response strategy is greater than β. Given Eq. (12) and based on Eqs. (10) and (11), we deduce that all customers with normalized lead times smaller than β have greater expected payoffs when not making AR. A customer with normalized lead time τ greater than β has an expected payoff of $\pi_{AR'}(\beta)$ if not making AR and $(1-C)\pi_{AR}(\tau)$ if making AR. If

$$(1-C)\pi_{AR}(\tau) < \pi_{AR'}(\beta), \quad \forall \tau, \tag{13}$$

then all the customers are better off not making AR. Otherwise, since $\pi_{AR}(\cdot)$ is a continuous increasing function, there exists a single value $p \in (\beta, 1)$ such that

$$(1-C)\pi_{AR}(p) = \pi_{AR'}(\beta). \tag{14}$$

If making AR, the expected payoffs of customers with normalized lead times smaller than p is at most $(1-C)\pi_{AR}(p)$. Hence, they will not make AR. The payoff of customers with normalized lead time greater than p, if making AR, is at least $(1-C)\pi_{AR}(p)$, and hence they will make AR. Thus, the best response of all customers is the threshold strategy p.

Next we show that the best response strategy p is bounded by p_m^e. Assume by contradiction that $p > p_m^e$. In this case, there exists $\epsilon > 0$ such that a customer with normalized lead time $\tau \in (p_m^e, p_m^e + \epsilon)$ is better off not making AR, namely:

$$(1-C)\pi_{AR}(\tau) < \pi_{AR'}(p). \tag{15}$$

Based on Eq. (12) and since

$$(1-C)\pi_{AR}(p_m^e) = \pi_{AR'}(p_m^e), \tag{16}$$

we deduce that

$$(1-C)\pi_{AR}(\tau) \geq \pi_{AR'}(\tau). \tag{17}$$

Since $\pi_{AR'}(\cdot)$ is a monotonic increasing function and since $\tau > p$, we deduce that

$$\pi_{AR'}(\tau) \geq \pi_{AR'}(p). \tag{18}$$

By combining Eqs. (17) and (18), we get that

$$(1-C)\pi_{AR}(\tau) \geq \pi_{AR'}(p), \tag{19}$$

which contradicts the assumption stated in Eq. (15). Thus, we have shown that $p \leq p_m^e$. ■

Next, we show that under the belief β, if the threshold customer is better off making AR, then the best response strategy to β is to make AR.

Lemma 3. *Given a belief β, if*

$$(1-C)\pi_{AR}(\beta) > \pi_{AR'}(\beta), \tag{20}$$

then its best response is $p = 0$.

Proof: Given that Eq. (20) holds and based on Eqs. (10) and (11), we deduce that customers with normalized lead times greater than β that make AR have at least the same payoff as the threshold customer. Thus, they are better off making AR. The expected payoffs of customers with normalized lead times smaller than β are $\pi_{AR'}(\beta)$ if not making AR. However, if making AR (each customer naively assumes that she is the only one deviating), the expected payoff is equal to the expected payoff of the threshold customer, namely to $(1-C)\pi_{AR}(\beta)$. Hence, the best response of all customers is to make AR. ■

In the next two sections, we use Lemmas 1-3 to separately analyze the dynamics of a system that is formed by a risk-averse provider and those of a system that is formed by a risk-taking provider.

V. Risk-averse Pricing

We define a risk-averse provider as one that advertises a fee which is not necessarily optimal, but has a unique *some-make-AR equilibrium*, and hence a guaranteed expected profit. Next, we show that for any system there is a low enough fee such that the game has a unique *some-make-AR equilibrium.*

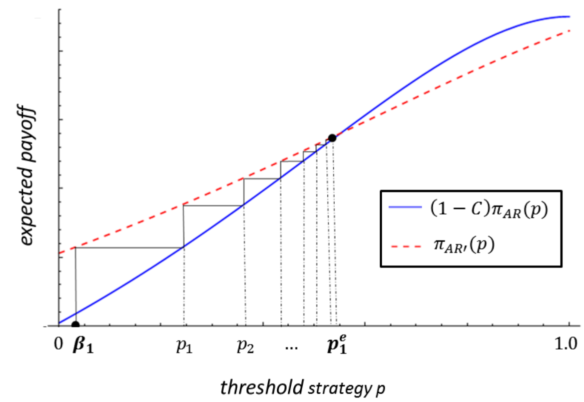

Fig. 1: The expected payoff functions of the threshold customer when the provider is risk-averse. The intersection point is the equilibrium strategy p_1^e. When using strategy-learning, after each iteration, the strategy followed by all customers is getting closer to p_1^e.

The uniqueness of the equilibrium (given that an appropriate fee C has been chosen) is an outcome of the following observations. 1) For any threshold $p > 0$, $\pi_{AR}(p) > \pi_{AR'}(p)$; 2) $\pi_{AR}(0) = \pi_{AR'}(0)$; 3) both $\pi_{AR}(\cdot)$ and $\pi_{AR'}(\cdot)$ are monotonic increasing functions. From those three observations, we deduce that there is a range of fees $(0, C^*]$, such that if C is in that range, the functions $(1-C)\pi_{AR}(\cdot)$ and $\pi_{AR'}(\cdot)$ intersect at a single point and therefore, the equilibrium is unique.

Next, we analyze the strategy-learning model and the action-learning model. We then compare between the provider's profits in the two models.

A. Strategy-learning

We begin with presenting our main result regarding converges to equilibrium.

Theorem 1. *Under strategy-learning, convergence to equilibrium is guaranteed.*

Proof: We split the proof into two cases. In the first case we assume that

$$(1-C)\pi_{AR}(\beta_1) < \pi_{AR'}(\beta_1). \tag{21}$$

From Lemma 2 we deduce that $p_1 \in (\beta_1, p_1^e]$. Since, $\beta_2 = p_1$, we deduce that if $\beta_2 \neq p_1^e$, then Eq. (21) will hold true, if β_1 is replaced by β_2. By induction, we deduce that, for any $j > 0$,

$$p_j \geq \beta_j \geq \beta_{j-1}, \tag{22}$$
$$p_j \leq p_1^e. \tag{23}$$

The set $\{p_i, i = 1, 2...\}$ is a monotonic increasing sequence bounded by p_1^e. Thus, it has a limit and $\lim_{i\to\infty} \overline{BR}(p_i) = p_i$. Since, the limit is a fixed point of $\overline{BR}$ we conclude that the limit is the equilibrium strategy p_1^e. Fig. 1 illustrates the convergence process.

In the second case, we assume that Eq. (21) does not hold. Therefore, based on Lemma 3, $p_1 = 0$ and $\beta_2 = 0$. Under the assumption that all customers make AR, making AR at time zero has no impact on the service probability. Hence,

$$(1 - C)\pi_{AR}(0) < \pi_{AR'}(0). \tag{24}$$

We conclude that $(1 - C)\pi_{AR}(\beta_2) < \pi_{AR'}(\beta_2)$. Therefore, from the second iteration, the system behaves as in the first case and converges to the unique equilibrium. ■

B. Action-learning

When action-learning is used, the strategy at each iteration is a random variable. Even if at some iteration the belief is equal to the equilibrium strategy p_1^e, it is not guaranteed that the fraction of customers not making AR at that iteration will be equal to p_1^e. Hence, with probability one, at some future iteration customers will not follow the equilibrium strategy. Once the belief is that the fraction of customers not making AR is greater than p_1^e, at the next iteration, all customer will make AR. Therefore, the strategy fluctuates between zero and p_1^e. We deduce that the expected number of customers making AR at each iteration is greater with action-learning that with strategy-learning. Thus, the provider is better off if customers use action-learning. Due to space limitation, the formal proof of this claim is omitted.

Theorem 2. *The expected profit of a risk-averse provider is greater in a dynamic system with action-learning than in a dynamic system with strategy-learning.*

Next, we present a simulated example that compares between the profit in the action-learning model and the strategy-learning model.

Example 1. *We consider a system with $N = 10$ servers and Poisson distributed demand with parameter $\lambda = 10$. We set the fee to $C = 0.125$. The unique equilibrium is $p_1^e = 0.116$, which yields the maximum profit that can be achieved when using risk-averse pricing. We perform 10 runs of the simulation. Each run consists of 10,000 iterations. At each run, the initial belief is $\beta_1 = p_1^e$. Using action-learning, the average profit per iteration is 1.067 and about 95% of the customers make AR. When using strategy-learning, the profit of the same realization is 1.013. Thus, the profit using action learning is about 5% greater than when using strategy-learning.*

VI. Risk-taking Pricing

It was shown in [1] that when advertising a high enough fee, the system has at least two *some-make-AR equilibria* and a *none-make-AR equilibrium*. In this section, we focus on systems with such fees. As in the previous section, we first analyze the strategy-learning model, followed by an analysis of the action-learning model.

A. Strategy-learning

Lemmas $1 - 3$ and the arguments used within the proof of Theorem 1 are sufficient for obtaining the following corollary.

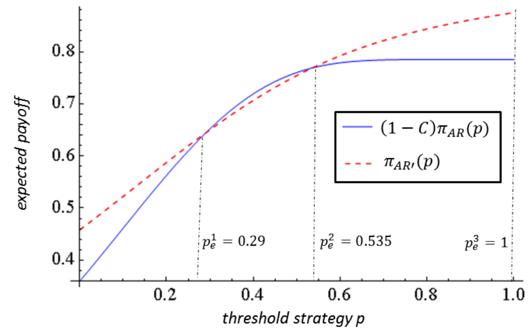

Fig. 2: The fee that leads to the equilibrium with the maximum possible profit, also leads to another *some-make-AR equilibrium* and to a *none-make-AR equilibrium*.

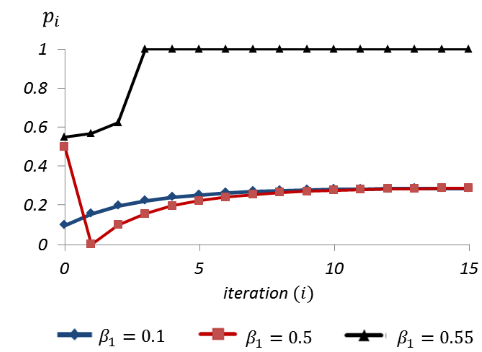

Fig. 3: Risk-taking pricing with strategy-learning. The initial belief determines to which equilibrium the strategy will converge.

Corollary 1. *In strategy-learning, given an initial strategy β_1, if $(1-C)\pi_{AR}(\beta_1) < \pi_{AR'}(\beta_1)$, then the system will converge to a some-make-AR equilibrium p_m^e where $m = \min\{j : p_j^e > \beta_1\}$. Otherwise, it converges to p_1^e.*

From Theorem 1 and Corollary 1, we deduce that the initial belief has no effect on the expected profit at steady state (after convergence to equilibrium) of a risk-averse provider while it determines the expected profit of a risk-taking provider.

Example 2. *We consider the same system as described in the previous example. But this time, we set the AR fee to $C = 0.215$. This fee has three equilibria $P^e = \{0.291, 0.535, 1\}$. The first equilibrium yields the maximum possible profit. From Fig. 2, we observe that if the initial belief is smaller than p_2^e, then the strategy will converge to p_1^e. Otherwise, it will converge to p_3^e. We set three different initial thresholds and apply strategy-learning. As Fig. 3 shows, within a few iterations, the strategy converges to the appropriate equilibrium.*

If the risk-taking provider has no control over the initial belief but the initial belief is a random variable with known distribution, then the provider can calculate the probability of convergence to each equilibrium. In this way, it can compute the overall expected profit.

Example 3. *We consider the same system as in Example 2 and assume that the distribution of the initial belief is uniform in [0,1]. In this case, with probability 0.535, the strategy* *will converge to p_1^e. The expected profit per iteration at that equilibrium is 1.478.[8] With probability 0.465, the strategy will converge to none-make-AR equilibrium, which yields zero profit. Thus, the expected profit at steady state with regards to the initial belief is 0.790, which is smaller than the maximum expected profit of a risk-averse pricing 1.016. In this example, being risk-averse is optimal, however, different distributions on the initial belief can lead to different conclusions.*

B. Action-learning

We showed in the previous section that when using action-learning, the strategy cannot converge to a *some-make-AR equilibrium*. If a *none-make-AR equilibrium* exists, we show next that the strategy eventually converges to that equilibrium.

Theorem 3. *Under action-learning and risk-taking pricing, the strategy converges to a none-make-AR equilibrium, with probability one.*

Proof: When using Cournot learning rule, the belief is the previous estimator. Hence, it is sufficient that in one iteration the estimator will be equal to one in order to converge to a *none-make-AR equilibrium* (this will occur if the threshold is smaller than one and no one arrived before the threshold).

For any given $p_i > 0$ the probability that $\hat{p}_i$ will be equal to one is given by

$$\mathbb{P}\left(\hat{p}_i = 1 | p_i)\right) = \sum_{n=1}^{\infty} \mathbb{P}(D = n) p_i^n. \tag{25}$$

This probability is positive for any value of $0 < p_i \leq 1$. At the boundary case of $p_i = 0$, the probability that none of the customers will make AR at the next iteration is zero. However, $p_i = 0$ is not an equilibrium point, and hence in the next iteration p_{i+1} will be greater than zero. Since a *none-make-AR equilibrium* is the only steady state and since for any given $p_i > 0$ there is a positive probability that $\hat{p}_i$ will be equal to one, we deduce that, with probability one, the strategy will converge to a *none-make-AR equilibrium*. ■

In the next example, we show that while the strategy can cycle many times, it eventually converges to a *none-make-AR equilibrium*.

Example 4. *We consider the same system as in Example 2, but this time with action-learning. The initial belief is that all customers make AR. We perform a simulation with 10 runs consisting each of 1000 iterations. In all 10 runs, the strategy converges to a none-make-AR equilibrium. The fastest convergence is within 8 iterations while the longest is within 242. On average, it takes 110 iterations to converge. Fig. 4 shows the convergence to a none-make-AR equilibrium in a typical run of the system.*

VII. Conclusion and Future Work

In this paper we studied the dynamics of a reusable resource system that supports advance reservations. We used a game-theoretic framework to analyze the behavior of customers in the system, while assuming that customers observe the behavior of the previous iteration and respond accordingly.

An interesting question that the paper addresses is whether the provider should reveal historical strategies or actions. We showed that if the provider is risk-averse and chooses a low fee that leads to a unique equilibrium, revealing the actions yields on average greater profit than revealing the strategies. On the other hand, if the provider is risk-taking and charges a fee that leads to multiple equilibria, then revealing previous actions will eventually cause all customers not to make AR. The concept of steering the system to a more desirable output, by controlling the information provided to customers, should be of interest to many other problems.

[8] A formula to calculate the expected profit of *some-make-AR equilibria* is given in [1].

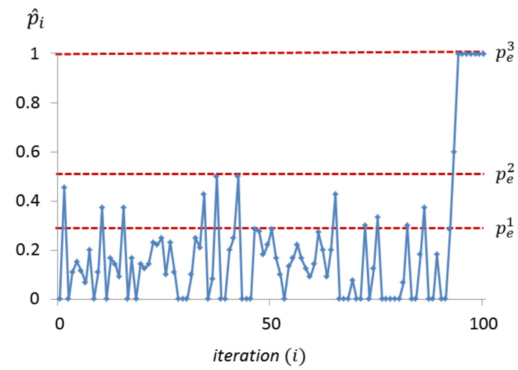

Fig. 4: Risk-taking pricing with action-learning. If a *none-make-AR equilibrium* exists, then at some point the strategy converges to it.

Acknowledgment

This work was supported in part by the U.S. National Science Foundation under grant CNS-1117160.

References

[1] E. Simhon and D. Starobinski, "Game-theoretic analysis of advance reservation services," in *Information Sciences and Systems (CISS), 2014 48th Annual Conference on*. IEEE, 2014, pp. 1–6.
[2] A. A. Cournot, "Recherches sur les principes mathematiques de la theorie des richesses, paris 1838," *English transl. by NT Bacon under the title Researches into the Mathematical Principles of the Theory of Wealth, New York*, 1897.
[3] S. Lakshmivarahan, *Learning algorithms theory and applications*. Springer-Verlag New York, Inc., 1981.
[4] D. Fudenberg, *The theory of learning in games*. MIT press, 1998, vol. 2.
[5] P. Milgrom and J. Roberts, "Adaptive and sophisticated learning in normal form games," *Games and economic Behavior*, vol. 3, no. 1, pp. 82–100, 1991.
[6] Q. Liu and G. van Ryzin, "Strategic capacity rationing when customers learn," *Manufacturing & Service Operations Management*, vol. 13, no. 1, pp. 89–107, 2011.
[7] E. Altman and N. Shimkin, "Individual equilibrium and learning in processor sharing systems," *Operations Research*, vol. 46, no. 6, pp. 776–784, 1998.
[8] E. Zohar, A. Mandelbaum, and N. Shimkin, "Adaptive behavior of impatient customers in tele-queues: Theory and empirical support," *Management Science*, vol. 48, no. 4, pp. 566–583, 2002.
[9] G. W. Brown, "Iterative solution of games by fictitious play," *Activity analysis of production and allocation*, vol. 13, no. 1, pp. 374–376, 1951.
[10] R. Cominetti, E. Melo, and S. Sorin, "A payoff-based learning procedure and its application to traffic games," *Games and Economic Behavior*, vol. 70, no. 1, pp. 71–83, 2010.