Costly Price Adjustment and Automated Pricing: The Case of Airbnb

Qi Pan *

Wen Wang †

[First Draft: Aug, 2019][Current Draft: Nov 2021]

Abstract

On many e-commerce platforms such as Airbnb, StubHub and TURO, where each seller sells a fixed inventory over a finite horizon, the pricing problems are intrinsically dynamic. However, many sellers on these platforms do not update prices frequently. In this paper, I develop a dynamic pricing model to study the revenue and welfare implication of automated pricing which allows sellers to update their prices without manual interference. The model focuses on three factors through which automated pricing influences sellers: price adjustment cost, buyer's varying willingness to pay and inventory structure. In the model, I also take into account competition among sellers. Utilizing a unique data set of detailed Airbnb rental history and price trajectory in New York City, I find that the price rigidity observed in the data can be rationalized by a price adjustment cost ranging from 0.9% to 2.2% of the listed price. Moreover, automated pricing can increase the platform's revenue by 4.8% and the hosts' (sellers') by 3.9%. The renters (buyers) could be either better off or worse off depending on the length of their stays.

Keywords: dynamic pricing; price-adjustment cost; automated pricing; two-sided platforms

*Email: panqi@cuhk.edu.cn, Marketing Department, School of Management and Economics, The Chinese University of Hong Kong (Shenzhen).

†Email: wenwangww@ust.hk, School of Humanities and Social Science, The Hong Kong University of Science and Technology.

1 Introduction

In markets where sellers have fixed inventories and limited selling time, optimal prices respond to both the remaining inventory and time (Gallego & van Ryzin, 1994). Empirical studies (Williams, 2018; Cho et al., 2018) examining pricing problem in these environments also support Gallego and van Ryzin's theoretical prediction (1994). However, on many e-commerce platforms where small sellers are facing a fixed inventory and limited selling time problem, we commonly observe price rigidity, which seems to contradict Gallego and Ryzins theory. Indeed, I find this to be the case in the data set I study from Airbnb. Furthermore, the pattern is not limited to new sellers. I find that some sellers with 1-2 years of experience on Airbnb also demonstrate a reluctance to adjust prices. There are some possible reasons for this. Most sellers on these platforms are not large firms and likely have other jobs, thus, they might not have enough time or find it too costly to manage their listings. Automated pricing, which uses machine learning algorithms to automatically price products, is becoming a standard feature on some of these platforms. For instance, Airbnb allows hosts to use "Smart Pricing" which automatically updates prices everyday according to the market conditions. Turo, a peer-to-peer car rental platform, and Stubhub, a ticket resale platform, allow sellers to use "Automatic Pricing"(Turo) or "Manage Price for Me"(Stubhub) to manage price setting without manual interference. One key feature of automated pricing is that it reduces seller's burden; the seller does not need to carry out a price-optimization problem every day.

In this paper, I investigate sellers' pricing behavior on the leading peer-to-peer accommodation platform Airbnb and examine the revenue and welfare effects of automated pricing. I develop an empirical framework to address the following research questions: What are the main factors that determine seller's pricing behavior? How does automated pricing affect the pricing behavior through these factors; and what are the revenue and welfare impacts of automated pricing?

For hosts on Airbnb, determining the "right" price to charge is a complex task. It requires that a host know not only how much the current renters value her listing, but also what future demand will be and her availability of supply. Since each host only has limited time to rent out her listing for a given check-in date, the more time the host has the more likely she will rent out her listing. The host may need to lower down her price tomorrow if she fails to rent out her listing today, unless renter who will arrive tomorrow is willing to pay more than the renter arrives today. Therefore, renter's willingness to pay would be one of the main driving forces for the pricing dynamics. In addition, listing's inventory structure is also an important element in determining the pricing dynamics. If the inventory of a listing is shrinking, the host might have an incentive to increase her price. However, smaller inventory can also imply smaller demand, thus, lower price. This is because rentals are generally in the form of different combinations of consecutive days, and the realization of one rental type may exclude a host from accepting other rental types.

While renter's willingness to pay and listing's inventory structure are two important factors that affect the optimal price, additional building block is needed to explain why many hosts on Airbnb do not adjust their prices. In the literature, "price rigidity" has been associated with the cost of price adjustments (Rotemberg, 1982; Levy et al., 1997; Slade, 1998; Aguirregabiria, 1999; Bergen et al., 2003; Zbaracki et al., 2004; Kano, 2006; Merlo et al., 2015 and Huang et al., 2018). This cost is due to the time and effort required in making price-adjustment decisions. In general, a host needs to collect information about demand and her competitors before she adjusts her price. This information collection process is sometimes time-consuming. In addition, calculating and justifying the optimal price require additional effort, which may further increase the host's cost for adjusting her price. The cost of price adjustment can also include laziness, ignorance or other unobserved factors that prevent a host from changing her price. If the gain from the price adjustment exceeds the price adjustment cost, the host will adjust her price according to the evolution of renter's willingness to pay and the inventory structure; otherwise, the price will stay the same.

Failing to respond to the change in renter's willingness to pay or inventory structure may result in revenue lost for the hosts. For instance, if a host's inventory is booked up too quickly, this may indicate that the host set her price too low. Therefore, the host should have raised her price earlier in the selling period. The price adjustment cost blocks the channels for price movement, thus, both the hosts and the platform may benefit from reducing the cost of price adjustment. There are many ways to do this. For example, the platform can provide tips and information to assist hosts to make their pricing decisions, or the platform can help hosts adjust their prices directly – through automated pricing. This paper specifically focuses on automated pricing. Ultimately, the automated pricing problem is a dynamic pricing problem which reduces the price adjustment cost and allows the price to respond to changes in the demand and inventory structure.

In order to study the pricing behavior of hosts on Airbnb and to evaluate the welfare impact of automated pricing, I build a dynamic pricing model with price adjustment cost. I allow heterogenous renters to arrive stochastically. I assume that they know when they want to travel, and how long they want to stay and that they choose among all the available listings without considering delaying their purchases. Given the expectation about how renters value their listings, forward-looking hosts maximize their expected revenues taking into account the cost of price adjustment. When hosts themselves have to bear the cost of price adjustment, they are reluctant to adjust their prices unless the benefit of the price adjustment exceeds the cost.

There are several challenges in the model setup and the estimation. First, each listing can only provide one unit of rental for a given check-in date. Therefore, I need to impose additional structure on the choice process to solve the listing assignment problem. Second, the competition among hosts causes the dimension of the dynamic pricing problem to grow exponentially with the number of listings. In order to reduce the dimension, I assume that each host only considers her own state and a set of system states, which summarize the information of her competitors, when making pricing decision. Third, even with the system state setting, the dimension of the choice variable and the number of unique listings still make the dynamic pricing problem intractable. In this scenario, I employ several clustering methods to shrink the dimension of the pricing problem.

Once the model is estimated, I focus on price adjustment cost as the first channel through which automated pricing affects pricing behavior. With automated pricing, price adjustment cost will be reduced, thus prices can be adjusted as frequently as necessary. In the case of automated pricing, the necessary price adjustments then depend on the change in renter's willingness to pay and inventory structure. Since renters arrive randomly and they have different stay length requirements, automated pricing will take into account the tradeoff between renting out a listing today versus tomorrow through these two channels. If renters arriving tomorrow are less price sensitive, automated pricing may need to increase the prices tomorrow to capture the opportunity for higher rental revenue. Additionally, if a host's inventory becomes scattered and the host does not have many consecutive days available, automated pricing will decrease the prices to help hosts attract more renters. Even though each listing on Airbnb is unique, there is still competition among hosts, thus, automated pricing will also consider the competition in the market. Although the impact of automated pricing depends on the interaction among those factors, the competition among hosts creates a significant computation burden. Since there are thousands of hosts in the market, it is not feasible to solve for the strategic interaction among them. To make the problem tractable, I follow the ideas developed in Weintraub et al. (2008), Sweeting (2015) and Buchholz (2018). I assume, in equilibrium, that each host's decision only depends on her own state, assuming that the rest of the market evolves deterministically along the path implied by these strategies. Therefore, the multi-agent dynamic pricing problem is reduced to a single-agent dynamic programming problem.

Since there is no theoretical result about the price adjustment cost reduction under the environment of varying renter's willingness to pay, special inventory structure and competition, the actual impact of automated pricing on revenue and welfare is unclear. This paper is the first one to use a structural dynamic pricing model to study the pricing behavior of hosts on Airbnb. The structural model can inform what factors automated pricing should consider. More importantly, this model also provide quantitative results about the revenue and welfare impact of the automated pricing.

The model is estimated using a novel data set of Airbnb pricing and rental history. The data set includes all the accessible characteristics including listing location, number of bedrooms, reviews and amenities. The data set also includes detailed pricing information which enables me to model the intertemporal price dynamics for given check-in dates. The demand function is formulated as the combination of the Poisson processes with different rental types and the discrete choice model which describes hosts' preferences among listings. Since some listing characteristics are not observable, they might contaminate the estimation result. In order to alleviate this problem, I supplement the analysis with information such as the characteristics of the building that the listing is in and the history of the host's operation. This information together with other observed listing characteristics, acts as the proxy to listing quality. Given the demand function, host's pricing decision is formulated as a dynamic discrete choice problem. When solving the pricing problem, the host only considers her own states and a system state that summarizes the information of her competitors. Under this setting, the "curse of dimensionality" problem caused by the competition among hosts is minimized. I also employ clustering methods to reduce the number of value functions in the system.

The parameters of the model are estimated via maximum likelihood in two steps. In the first step, I recover the parameters in the Poisson process and utility function. In the second step, I recover the price adjustment cost. I find that the price adjustment cost decreases over time and that it is between 0.9% and 2.2% of the average listed price. For instance, if the average listed price is \$195 the price adjustment cost is about \$1.77 to \$4.28. I also find that renters with longer stays tend to book early and they are more sensitive to price changes when they are compared to those who plan for shorter stays. This finding suggests that the hosts have an incentive to increase their prices to capture those with higher willingness to pay. However, because of the diminishing time to sell and the price adjustment cost, the increasing trend is suppressed and prices are frictionally decreasing.

Given the parameter estimates, I evaluate the impact of automated pricing, thereby providing managerial insights. Automated pricing is formulated as a dynamic pricing process without price adjustment. I find that the automated pricing increases Airbnb's revenue by 4.8% and the hosts' revenue by 3.9%. This result indicates that both the platform and the hosts benefit from reducing the cost of price adjustment. From Airbnb's perspective, this would be a strong evidence for supporting automated pricing. I also find that, without the price adjustment cost, the hosts tend to first increase their prices and then reduce them as the check-in date draws near. Renters who arrive at the beginning or late during the selling period benefit from automated pricing, since they face lower prices. However, those who arrive in the middle of the selling period are worse off due to higher prices. Additionally, I compute a counterfactual of reducing the price adjustment cost to one half of its original level and calculate the revenue and welfare impacts. This counterfactual is meant to represent the case where Airbnb only provides pricing suggestion instead of automated pricing. Finally, I allow the automated pricing to update host's price weekly rather than daily during the selling period. In practice, a daily price update could be computationally burdensome to the platform. Therefore the platform may have an incentive to consider weekly automated pricing instead of daily. In this counterfactual, weekly automated pricing increases Airbnb's revenue by 4.1% and the hosts' revenue by 3.4%. Although the revenue improvement under weekly automated pricing is less than daily automated pricing, the difference is small.

To my knowledge, this is the first empirical study that explores the pricing dynamic and automated pricing in the context of Airbnb. This paper contributes to the literature in the following three aspects. First, I develop a novel application of dynamic pricing under price adjustment cost using a high frequency data set that allows the modeling of intertemporal pricing behavior. Most of the existing literature on Airbnb (Zervas et al., 2016; Barron et al., 2018; Li & Srinivasan, 2018) uses snapshot data that only allows researchers to observe one price for each check-in date. This paper also provides empirical evidence to support automated pricing whose revenue effect is not clear to both the platform and researchers. Second, in the hospitality industry, modeling different lengths of stay is increasingly appealing to both hotel managers and researchers. In this paper, I allow the modeling for multi-day rentals which are essential for understanding the pricing dynamics on Airbnb. Lastly, I empirically solve the dynamic pricing problem with many sellers by using an equilibrium concept similar to the oblivious equilibrium. This equilibrium concept significantly reduces the computational burden.

The rest of the paper is organized as follows. Section 2 reviews the related literature. In section 3, I use a simple dynamic pricing model to demonstrate what potential factors affect the optimal price path. Section 4 introduces the data and provides some model-free evidence regarding the pricing dynamics. Section 5 sets up the model. In Section 6 and 7, I describe the estimation method and results. The counterfactual experiment is presented in Section 8 and Section 9 presents the conclusion.

2 Related Literature

The model used in this paper builds on the literature of dynamic pricing, which has been widely studied in economics, marketing and operation research communities both theoretically and empirically (Stokey, 1979; Besanko & Winston, 1990; Gallego & van Ryzin, 1994; Talluri & van Ryzin, 2004; Nair, 2007; Hendel & Nevo, 2013). More specifically, I focus on the dynamic pricing in the markets where sellers only have limited time to sell a limited amount of inventory. Intertemporal price discrimination naturally arises in these markets because of the restriction on selling time and quantity. Under different situations, price can be either decreasing or increasing (Su, 2007). For example, if renter's willingness to pay of the listing is constant or decreasing over time, a seller should decrease her price if there is less time to sell for a given size of the inventory, or at a given point in time, the optimal price will increase as the size of the inventory decreases (Pashigian, 1988; Pashigian & Bowen, 1991; Gallego & van Ryzin, 1994; Bitran & Mondschein, 1997; McAfee & te Velde, 2008). If renter's willingness to pay is increasing or non-monotonic over time, there is no general prediction about the direction of the optimal price. Sweeting (2012) adopts a simple dynamic pricing model to capture the pricing behavior of event ticket sellers on a secondary event ticket website. He concludes that a dynamic pricing model can fit the data well and explain why there is a drop in price as the event date approaches. Under the assumption that the continuation value does not depend on current price, Sweeting's model naturally generates the non-increasing pattern of the price path. Lazarev (2018) estimates a dynamic pricing model using airline data and analyzes the welfare effect of airline pricing. Neither Sweeting (2012) nor Lazarev (2018) allows for stochastic arrival of consumers; therefore, their analyses mainly center around the intertemporal price discrimination. Williams (2018), however, puts stochastic arrival and intertemporal price discrimination under the same framework and successfully disentangles the effects of these two forces. In his model, business travelers, who tend to arrive late, are more likely to be willing to pay higher prices. In order to maximize profit, airline companies have incentive to reserve seats for late arrival customers. Williams demonstrates that intertemporal price discrimination is then complementary to the price change according to stochastic demand. In this paper, both stochastic demand and intertemporal price discrimination exist but in a more complicated manner due to the extreme limited capacity and the different length of stays. Additionally, I allow price to depend on inventory structure. The mixed effect of these elements may generate a non-monotonic price path. Therefore, this study supplements the literature by providing extra empirical evidence that there is no general prediction about the price under the dynamic pricing framework.

This paper also relies on the literature about the cost of price adjustment. Many empirical studies have shown that both managerial costs and physical costs for price adjustment are significant in retailing and other industries (Rotemberg, 1982; Levy et al., 1997; Slade, 1998; Aguirregabiria, 1999; Bergen et al., 2003; Zbaracki et al., 2004; Kano, 2006). More recent empirical papers by Merlo et al. (2015) and Huang (2018) incorporate price adjustment cost into the dynamic pricing setting to explain the price stickiness observed in the data. Little is known, however, about the impact of removing the price adjustment cost in a competition environment. A theory paper by Dana and Williams (2018) mentions that competition in the market can prevent sellers from using intertemporal price discrimination when less elastic buyers arrive late in the selling period. Empirically, it can be prohibitive to calculate the equilibrium of a dynamic pricing model with competition because of the exponentially growing state space. Taking advantage of the large number of players, the oblivious equilibrium (Weintraub et al., 2008) and similar equilibrium concepts used in Buchholz (2018) and Sweeting (2015) simplify the calculation by assuming that each individual's decision only depends on her own state and a system state but not the behavior of each competitor in the market. By using a similar equilibrium concept, this paper extends the dynamic pricing under competition environment to accommodate price adjustment cost.

Other relevant papers include literature on empirical analysis of Airbnb. Airbnb, which was founded in 2008, is growing extremely quickly; however, there are only a few empirical papers on this topic. Zervas et al.(2016) is the first paper that focuses on Airbnb and hotel interaction. Using hedonic regression analysis, Zervas et al.(2016) examines the impact of Airbnb on hotel revenue in Austin, TX and concludes that the existence of Airbnb reduces hotel revenue by 8%-10%. Li and Srinivasan (2018) studies hoetls' strategic responses to Airbnb. They discover that Airbnb's presence help recover the lost underlying demand due to hotel seasonal pricing, therefore, some hotel may benefit from less seasonal or even counter-seasonal pricing. Farronato and Fradkin (2018) examines the welfare impact of Airbnb on travelers, hosts and hotels. Combining a data set that includes listings from the entire United States with data from Zillow on housing and rental prices, Barron et al.(2018) finds that a 1% increase in Airbnb listing leads to a 0.018% increase in rents and a 0.026% increase in house prices. My paper investigates Airbnb through the lens of pricing dynamics and provides insights into the automated pricing feature.

3 Illustrative Examples

In this section, I provide three examples to illustrate how optimal price patterns change in response to renter's willingness to pay, inventory structure and price adjustment cost. These examples frame the pricing problem of hosts and show what factors automated pricing should consider.

Suppose there is one seller who owns one unit of a product. The product is perishable; if it is not sold by time $t = 0$, the value of the product becomes zero. The seller starts selling at time $t = T$ and can set a different price in each period. The objective of the seller is to maximize the expected value defined by the Bellman equation:

$$V_t = \max_{p_t} p_t D_t(p_t) + (1 - D_t(p_t))V_{t+1}$$

V_t is the expected profit for selling the product. With probability $D_t(p_t)$, the item will be sold at price p_t. With probability $1 - D_t(p_t)$, no one purchase the item and the seller enters into next period when she can set new price for the product. $D_t(p_t)$ is changing over time and buyer's willingness to pay of the product might also be time-varying. V_{t+1} is the continuation value

Table 1: Parameter Setting of the Examples

	μ_t	α_t
Case 1	0	$\frac{0.02}{T}t + 0.06$
Case 2	0	$\frac{0.06}{T}t + 0.02$
Case 3	0	$\frac{0.07}{T}t + 0.01$

which describes the opportunity cost if the item is sold at period t. The optimal price p_t^* is determined by the first order condition:

$$\frac{p_t^* - V_{t+1}}{p_t^*} = \underbrace{\frac{D_t/p_t^*}{-\partial D_t/\partial p_t}}_{\text{Lerner index}}$$

where the right-hand side of the equation is the first order condition is the Lerner index which is the inverse of price elasticity. V_{t+1} is the opportunity cost for selling the product at period t. It is easy to prove that the opportunity cost is decreasing in time t if $D_t(p_t)$ is decreasing in p_t (Sweeting 2012). While the opportunity cost is decreasing, the price elasticity could move in any direction.

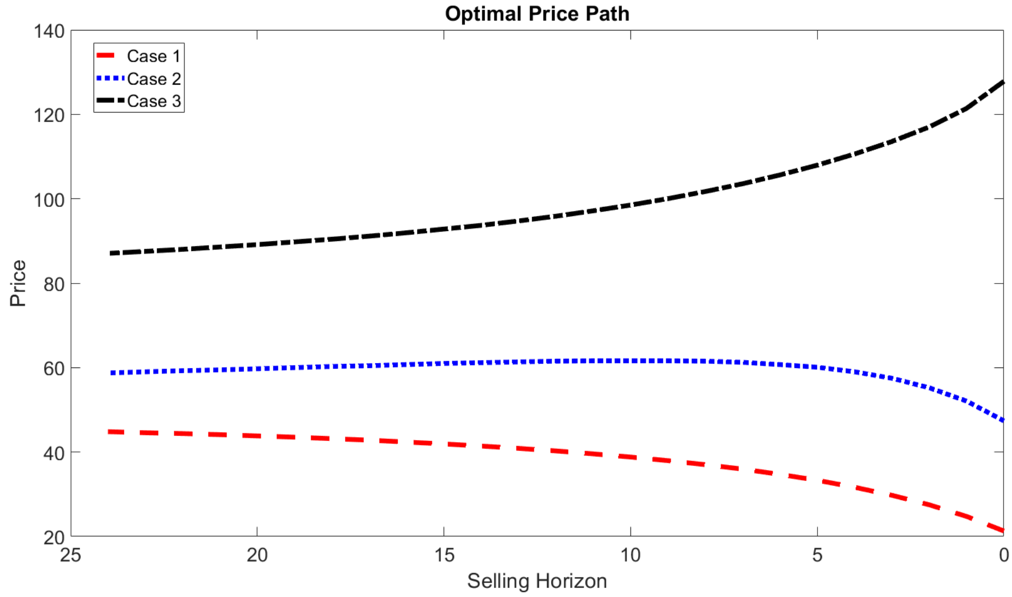
Varying Buyer's Willingness to Pay

I assume that there is only one buyer in each period and her willingness to pay of the product is v_t. If v_t is greater than the price of the product p_t, she will purchase the product. I assume that v_t follows a logistic distribution with location parameter μ_t and scale parameter $\frac{1}{\alpha_t}$. Therefore, the probability for the buyer to purchase the product is

$$\Pr(v_t > p_t) = D_t(p_t) = \frac{\exp(c_t - \alpha_t p_t)}{1 + \exp(c_t - \alpha_t p_t)}$$

where $c_t = \mu_t \alpha_t$. I consider 3 cases with 3 different set of parameters that are summarized in Table 1: In each of these 3 cases, the seller has $T = 25$ period to sell and α_t decreases linearly from $t = T$ to $t = 0$, so that the buyer's willingness to pay is higher toward the end of the selling period. When the buyer's willingness to pay is increasing fast enough, the seller has an incentive to increase her price over time. Otherwise, the decreasing opportunity for sale dominates, and the seller will reduce her price to encourage the transaction. In Case 1, the price sensitivity parameter α_t decreases from 0.08 when $t = T$ to 0.06 when $t = 0$. The optimal price decreases even though the buyer's willingness to pay increases. In Case 2 (α_t decreases

Figure 1: Optimal Price Paths under Different Price Sensitivity



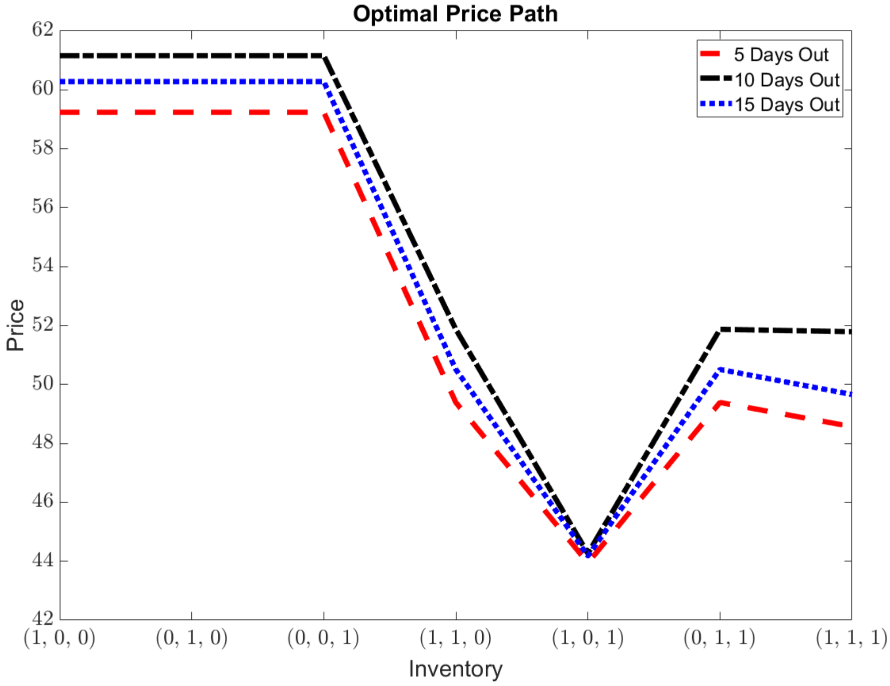
from 0.08 to 0.02), the optimal price is increasing at the beginning and then decreasing. The increasing trend of buy's willingness to play dominates the decreasing trend of opportunity cost in the first 15 period. After that, the opportunity cost has the upper hand and pulls down the price. In Case 3 (α_t decreases from 0.08 to 0.01), the optimal price is increasing because the demand elasticity is dropping fast enough to compensate for the decreasing opportunity for sale. In this example, the optimal price paths can be increasing, decreasing or non-monotonic. This indicates that renter's willingness to pay is crucial in determining the price pattern.

Inventory Structure

Next, I extend the above simple dynamic pricing model to accommodate the inventory structure. Instead of selling one product, the seller has 3 products a, b and c in this example. There is only 1 unit of each product. I also assume that the buyer can purchase one product or multiple products. If the buyer purchases multiple products, they must be in the form of (a,b), (b,c) or (a,b,c). This is similar to room booking on Airbnb – a renter books a single day or consecutive days. Since the seller only has one unit of each product, the size of the inventory may not provide sufficient information for pricing. Which product(s) is(are) left would be more informative. Figure 2 demonstrates the optimal price path under different inventory[1]. (1, 1, 1) indicates that product a, b and c are all available, and (0, 1, 1) indicates that only product

[1]The detail of the setting is in the Appendix

Figure 2: Optimal Price Paths across Different Inventory



b and c are available. The optimal price is not monotonic in the size of the inventory. For example, (1, 0, 1) excludes buyers who want to buy (a, b) or (b, c) and potentially faces a smaller market than (1, 1, 0), therefore the seller has an incentive to cut the price. This result is essential in understanding the price dynamics on Airbnb. In many industries where there is no structure restriction on inventory, price is generally a non-increasing function in the size of inventory. However, a host on Airbnb may need to reduce her price when "orphan days" – short period between bookings – appear, because her listing is less likely to be rented on those isolated days .

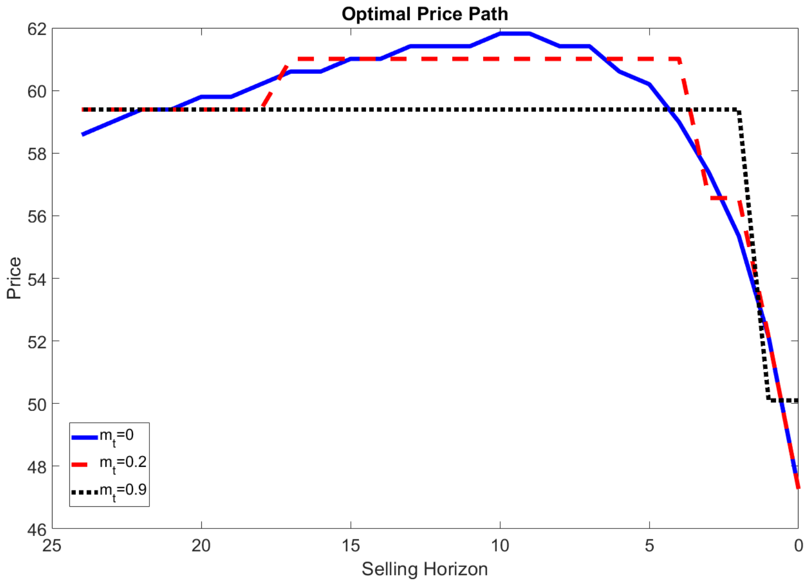
Price Adjustment Cost

In this last example, I assume that price adjustment is costly.

$$V_t = \max_{p_t} p_t D_t(p_t) + (1 - D_t(p_t))V_{t-1} - m_t * 1(p_t \neq p_{t+1})$$

where m_t is the price adjustment cost. Using the parameter setting in Case 2 in the first example, I calculate the optimal price path under different m_t (Figure 3). The price adjustment cost causes the price to underreact to the change in the demand. When the price adjustment can generate greater revenue improvement than its cost, the seller will then adjust the price. If

Figure 3: Optimal Price Paths under Different Price Adjustment Cost



the price adjustment cost is large enough, the optimal price does not respond to the change in the renter's willingness to pay or the inventory structure.

The existence of the price adjustment cost becomes an obstacle for efficient pricing. Therefore, reducing this cost and restoring the pricing mechanism, which reflects the change in the market, are the essential task for automated pricing. The examples described in this section abstract away from many other factors such as competition and renter heterogeneity, and hence I develop a more realistic model in Section 5. The factors discussed here remain the key drivers of the price dynamics on Airbnb.

4 Data and Descriptive Analysis

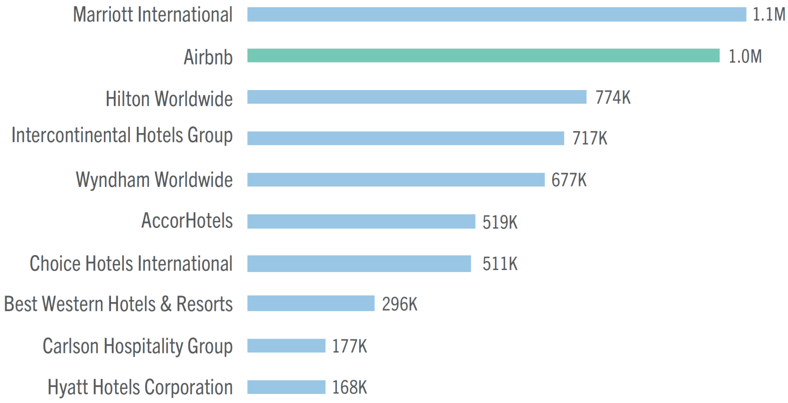
4.1 The Platform: Airbnb

Airbnb, an online platform established in 2008, allows people to list or rent short-term lodging in residential properties. Airbnb now operates in more than 190 countries and has more than 3 million listings; one third of these listings are comparable to hotel rooms (entire apartment with less than one bedroom, Figure 4). Airbnb is the dominant player in the peer-to-peer apartment rental market and it is growing rapidly. Airbnb's market share of room nights sold in the U.S. lodging market grew from less than 1% in 2014 to more than 8% in 2017[2].

If a renter wants to rent an apartment on Airbnb, she can search properties in a certain

[2] http://www.hotelappraisers.com/airbnbs-market-share-of-u-s-lodging-demand/

Figure 4: Number of Comparable Rooms

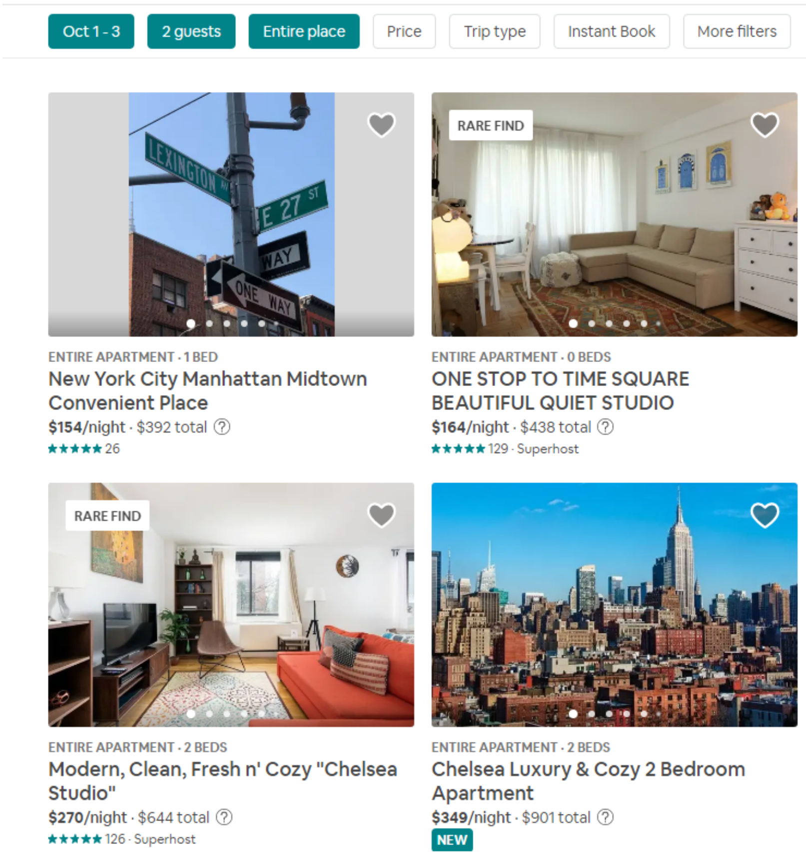


area using a specific search filter such as listing type and number of bedrooms (Figure 5 is an example of a search result). If a property has the "instant booking" sign, the renter can rent this apartment instantly without communicating with the host or waiting for permission; otherwise the transaction will not be finalized before the host confirms. After the transaction, the renter can cancel or alter the rental with or without penalty depending on the host's cancellation policy. The host, on the other hand, can also cancel the rental. If the cancellation is initiated by the host, Airbnb will impose a $50 fine on the host and post a message in the review system stating that the host cancelled the rental. After the stay, the renter can leave review for this listing.

The hosts have complete control over the rental prices, descriptions, cancellation rules and availability of their apartments. Airbnb allows hosts to set prices 12 months in advance. Hosts can also decide whether to add a cleaning fee to the rental price. In principle, a host can have more than one listing on Airbnb. While a multiple-listing owner might be a concern in the analysis; it is believed that none of these multiple-listing owners has dominant power in the market. Airbnb has also restricted the operation of this type of owners after May 2015, though it is possible that a host uses different family members' names to operate multiple properties. Since it is impossible to discern these multiple-listing owners in the data, I assume that each listing is owned by a separate owner. Airbnb collects commission fees from both hosts and renters. The commission fee from the host is fixed at 3% of the transaction (excluding tax but including cleaning fee) and the service fee collected from renters ranges between 6% - 12%[3]. In New York City, for instance, this service fee is always around 12%.

[3]https://www.airbnb.com/help/article/1857/what-are-airbnb-service-fees

Figure 5: An Example of Search Result



Beginning in November 2015, Airbnb launched an automated pricing algorithm called "S-mart Pricing". If a listing owner chooses to use this algorithm, Airbnb adjusts the price automatically. For instance, Airbnb may increase the host's price if other similar listings are renting out rapidly, or it may reduce the price if demand is low. With "Smart Pricing", listing owners no longer bear the burden of monitoring their prices. Changing price becomes a more frequent behavior. Motivated by this new feature, I build a dynamic pricing model to investigate the factors that automated pricing should consider and the welfare implication of the automated pricing.

4.2 Observed Variables

The main data set is collected from Airbnb's website. For each listing in the sample, I observe listing type (shared room, private room or entire apartment), number of bedroom/bathroom, amenities (such as air conditioning, wifi, TV, washer), location, cancellation policy, minimum stay requirement, number of photos, rating, number of reviews, length of reviews, whether the listing can be booked instantly, whether the owner is a superhost[4], cleaning fee, price sequence of each check-in date, when the listing was booked, transaction price and a unique listing ID.

The main data set does not include any individual information about the renters except for the time when they booked the listings, their check-in dates and the length of their stays. This data limitation reduces the room for renter heterogeneity. However, the length of stay can still play an important role to capture the heterogeneity: renters with different length of stay may arrive very differently across time and have different level of price sensitivity.

In addition to the information mentioned above, I further observe when each listing was created. This variable is constructed by combing the main data set with the information provided by Inside Airbnb[5] through unique ID matching. This variable provide a measure of how experienced the host is. Moreover, I augment my data with the age and the value of the property at the building level from Infogroup. To construct these two variables, I use the location information to match the Infogroup data to the main data set[6]. This additional information helps control for the individual specific characteristics. However, the matching between Airbnb data and the Infogroup data is not straightforward. Airbnb does no provide the exact location of a listing. In general, the distance between the true location and the data location ranges from 0-400 feet. I assign each listing to the nearest residential building and use this to match the Infogroup data set.

4.3 The Sample

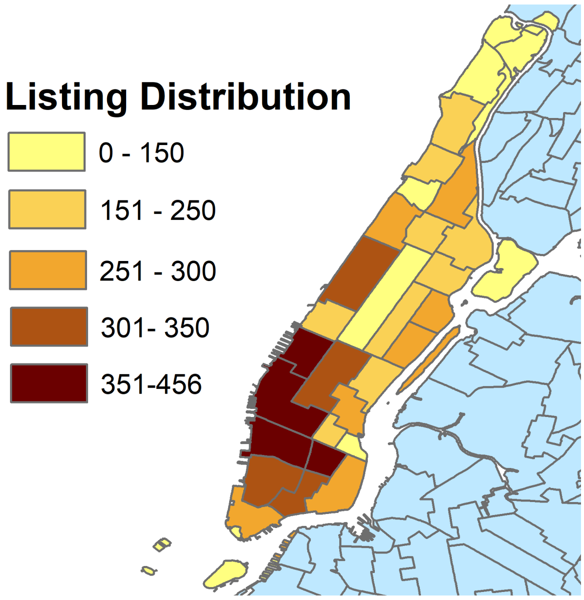
In order to understand the pricing behavior and evaluate the impact of automated pricing, it is necessary to have pricing data that track the same listing and the same check-in date across time. The price calendar for Oct 4, 2015 to Oct 31, 2015 for each host on Manhattan in New York City was scanned on a daily basis starting from Aug 2, 2015. For each host in this sample, I observe at least two months (56 days) of price movement for each check-in date between Oct 4, 2015 and Oct 31, 2015. In principle, Airbnb allows hosts to set price 12 months in advance. Since 89% of the rentals were booked within two months of the check-in date, the length of the price sequence considered in this paper is sufficient for capturing the major price

[4]The superhost rating is an indicator for good reputation. To be qualified as superhost, a host has to meet serval criteria. https://www.airbnb.com/superhost

[5]http://insideairbnb.com/. This website provides historical Airbnb listing information in many cities around the world

[6]There could be multiple apartments in one building. In this case, the property age and value are the same across these apartments

Figure 6: Listing Distribution



dynamic. More importantly, Airbnb launched "Smart Pricing" at the end of Nov 2015 which was later than the sample period. Therefore, this sample allows me to examine the implications for automated pricing during a time when hosts were not exposed to this pricing tool. It is possible that some hosts used other third-party automated pricing services in my sample, but this number would be very small. This is also confirmed by the pricing team at Airbnb.

To deliver the final sample for analysis, I restrict my investigations to listings that are provided as "entire apartments". The reason for this restriction is twofold. First, listing type is the most used filter on Airbnb when renters are searching for listings. Therefore, listings with different type may not be considered at the same time by renters. Second, hosts with "shared room" or "private room" type are more likely to block their listings (remove their listings from the market). This behavior might add extra complication to the model. Among listings in the "entire apartment" category, I keep those with two or fewer bedrooms, that have a two or fewer minimum nights requirement and a price range between \$50 and \$350. As a result of applying these criteria, the total number of listings used in the analysis is 5,434. Figure 6 shows the geographic distribution of these 5,434 listings. The number of listings has substantial variation across different neighborhoods.

Table 1 provides price statistics on the distribution of prices across listings and days.. The average price for a listing, regardless of rental status, is \$195 per night. This number is about one half of the hotel average price of \$315 in the same area. The average transactional price, which is conditional on being rented, is \$193 per night. Meanwhile, for those listings which

Table 2: Price Statistics

	Mean	SD	Median	Min	Max	Obs.
All	195.3	125.6	192	50	350	162,834
Rented	192.6	121.3	189	50	350	52,106
Not Rented	201.2	151.2	195	50	350	110,728
Weekend	205.3	156.7	201	55	350	45,592
Weekday	190.5	144.3	188	50	350	117,242

An observation is the combination of a listing and a check-in date. "All": price regardless of rental status. "Rented": transaction price. "Not Rented": lowest price for the check-in date. "Weekend and Weekday": weekend price and weekday price regardless of rental status

were not rented, the average price is $201 per night. While it indicates that listings with lower prices are more likely to be rented, this result could be misleading without controlling for the characteristics of the listings. In addition, the rate per night for the weekend is higher than the weekday. This can be explained by the fact that during the sample period, Airbnb primarily targeted leisure travellers rather than business travelers who are less price sensitive and would mostly stay on weekdays.

While the pricing information can be accurately collected from the website, obtaining accurate supply and demand is challenging for three reasons. First, listing owners or Airbnb can block some check-in dates[7]. Second, a host can reject a renter's request even though the apartment is available. Lastly, renters can cancel their reservation before check-in. The accuracy of the data, however, is not affected by the first concern because I can distinguish whether a check-in date is blocked or booked. Before Nov 2015, blocked dates and booked dates were coded differently; after that, however, it is not possible to distinguish between blocked rooms and booked rooms[8]. Since the time period in the sample is Oct 2015, this change has no impact on the analysis. Moreover, I do not model rejection in this paper because the communication between host and renter is not accessible. In the data collection process, I do not have access to the cancellation data on either the hosts or the renters. However, Table 2 shows that about 73% of the hosts implemented strict cancellation policies. A strict cancellation policy means once a renter has paid for the rental, any modification or cancellation will result in a 50% fine. Although ignoring cancellation might bias the estimation, the effect is believed to be small due to the high cost of cancellation.

[7]If a host cancel a rental or violate the platform's rules, Airbnb can block the host too

[8]Some researchers or data collection companies use machine learning technique and review information to infer the booking status. According to Fradkin et al. (2017), around 70% of the renters actually leave reviews which partly confirm whether a check-in date is booked or blocked.

Table 3: Cancellation Policy

	Flexible	Moderate	Strict
Freq.	815	653	3,966
Percent	15.00%	12.02%	72.98%

"Flexible": full refund if cancelled at least one day before check-in date. "Moderate": full refund if cancelled at least 5 days before check-in date. "Strict": 50% refund if cancelled at least 7 days before check-in date otherwise no refund

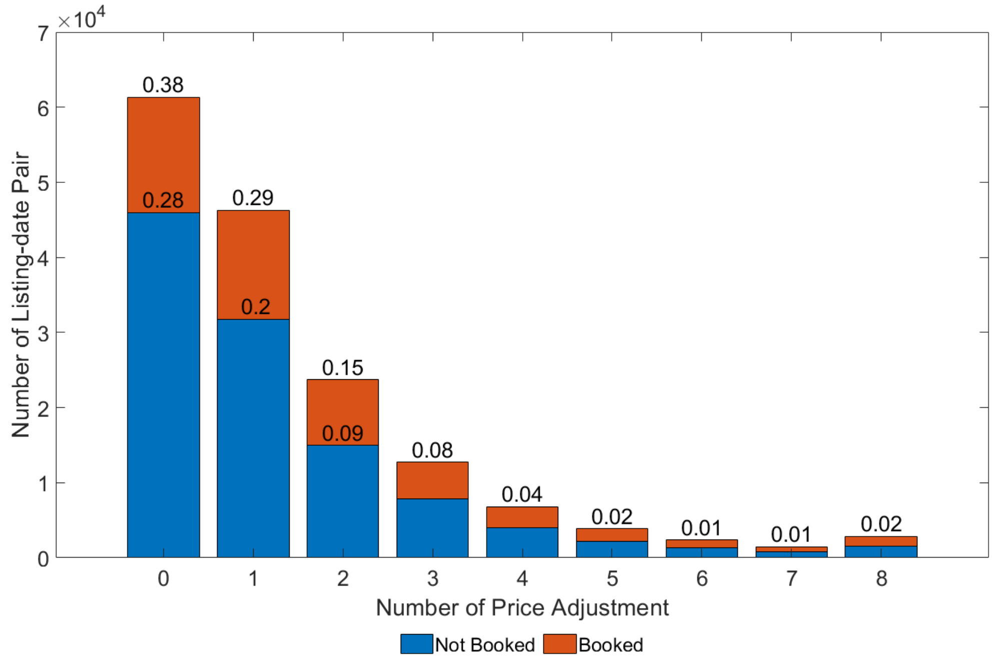
4.4 Price Rigidity and Rental Patterns

While the summaries in the previous section were essentially cross-sectional, this subsection presents the main price and rental patterns. These patterns are crucial for building the model for evaluating automated pricing. In particular, the low price change frequency strongly suggests that automated pricing may have a substantial impact on host's pricing behavior and revenue by reducing the price adjustment cost.

In Figure 7, I provide the distribution of price adjustments. In 38% of the listing-check-in-date pairs there is no price adjustment. No price adjustment means that a host keep her price constant throughout the selling period. Among all the listing-check-in-date pairs where the number of price adjustment is zero, 74% (0.28/0.38) of them are not rented. This ratio is decreasing in the number of price adjustment, which indicates that more price adjustment is positively related to higher rental probability. Moreover, one can observe that over 80% of the listing-check-in-date pairs have 2 or less price adjustments during the selling period.

The above observation indicates that hosts are reluctant to adjust their prices. When price adjustments happen, most of them are price reductions. Table 3 reveals that the majority of price adjustments are price drops. For instance, more than 47% of the observations (listing-check-in-date pair) show price reduction over the 8-week long period and only 14% actually raise their prices. Conditional on being rented, the average and median price drop are -$48 and -$35 respectively, and the average and median price change are -$23 and $-5. Conditional on listings that are not rented, the average overall price change is -$15 and the average price drop is -$40. Similar to the results shown in Table 1, rented listings tend to have greater price cut than listings that are not rented. Furthermore, Figure 7 and Table 3 together provide a more integrated picture of hosts' pricing behavior – some hosts do not adjust their prices frequently and the overall trend of price is decreasing. This observation is also confirmed by Figure 8, which shows the average listing price over the 56-day selling period. The relatively flat part at

Figure 7: Price Adjustment Frequency



the beginning indicates that hosts do not adjust their prices frequently. However, the average price is gradually decreasing from \$210 to \$190

In order to obtain a more accurate price adjustment pattern, I regress the log-price on a set of characteristics, location dummies and time dummies.

$$\log(P_{t,k}) = D_t\beta_1 + X_k\beta_2 + Z_{t,k}\beta_3 + C_m\beta_4 + \epsilon_{t,k} \tag{1}$$

where $P_{t,k}$ is the listed price for listing k, t days before the check-in date m. C_m is the check-in date dummy. D_t denotes the time dummy that measures the days prior to the check-in date. X_k is the characteristics of listing k including room type, number of bedrooms, amenities, cleaning fee, deposit, number of photos, cancelation policy, reviews and location. $Z_{t,k}$ are the variables that measure the number of competitors and average price of those competitors.

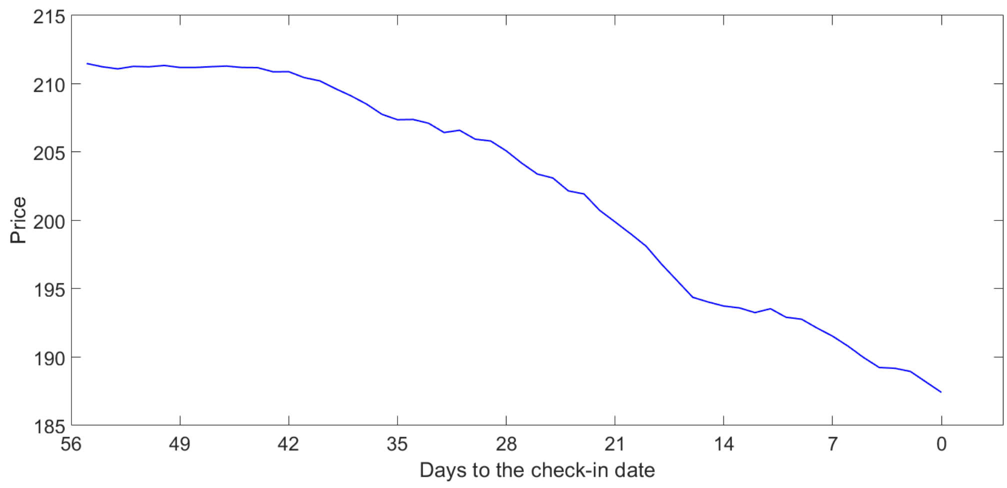
Table 4 shows the results from utilizing all observations regardless of whether the listings are rented or not. The coefficients in Column (1) are estimated without adding check-in date fixed effect. The price falls about 9% from 8 weeks away to the check-in date and the price drop speed becomes more rapid when it approaches the check-in date. Column (2) is the result with check-in date fixed effect. The price also drops about 9% from 8 weeks away to the check-in date, and the increasing speed pattern is preserved. This result indicates that the price trend

Table 4: Price Adjustment Direction & Magnitude

	Mean	SD	10th	25th	50th	75th	90th	% to Group
All								
Increase	$19.2	43.5	$5	$9	$15	$35	$50	14.23%
Decrease	$-45.1	122.1	$-54	$-45	$-25	$-20	$-5	47.56%
Entire Group	$-20.1	98.2	$-50	$-25	$0	$0	$5	100%
Booked								
Increase	$16.2	35.9	$5	$10	$15	$25	$30	17.13%
Decrease	$-48.3	111.3	$-65	$-60	$-35	$-25	$-15	52.13%
Entire Group	$-23.5	85.1	$-60	$-35	$-5	$0	$5	100%
Not Booked								
Increase	$20.1	65.7	$5	$10	$15	$35	$55	15.17%
Decrease	$-40.1	133.5	$-65	$-60	$-30	$-15	$-5	45.41%
Entire Group	$-15.2	112.2	$-60	$-20	$0	$0	$5	100%

An observation is the combination of a listing and a check-in date. "All": all observations regardless of rental status. "Booked": observations that are rented. "Not Booked": observations that are not rented. "Increase": the transaction price or the "last minute" price is higher than the initial price. "Decrease": the transaction price or the "last minute" price is lower than the initial price

Figure 8: Average Price



during the selling period does not vary much across different check-in dates. One difference between the result in Table 4 and Figure 8 is that in Figure 8 the average price drops about 13% from the beginning of the selling period to the end, however, the price decrease shown in Table 4 is about 9%. Since $Z_{t,k}$, the number of competitors and the average price of those competitors, are the only time-varying variables other than the time dummies, the competition among hosts contributes to the price dynamics and is an important element in determining the listing price. Lastly, in order to investigate the price variation across different price levels, 5 quantile (10%, 25%, 50%, 75%, 90%) regressions are run. Most of the price decreases happen at the lower 25th percentile, which means listings with lower prices tend to have more price adjustment than those with higher prices.

From the regression analysis, one can see the overall price pattern. However, this is not enough to inform the dynamic pricing modeling. As it is mentioned in previous section, renters with different length of stay requirements may have very different price preferences. In addition, the size and the structure of the inventory are also crucial in the pricing procedure. The key ingredient to connect inventory structure and pricing decision is the modeling of multi-day rental. If each check-in date is treated separately, like it is in most of the prior literature, the estimated demand will not accurately capture renters' true preferences. Figure 9 shows the number of rentals with different length of stay for the check-in dates between Oct 4, 2015 and Oct 10, 2015. The total number of rentals, regardless of rental length, was around 4400. Multi-day rentals (≥ 2 days) account for 74% of the total rentals. Therefore, ignoring the multi-day structure may bias the demand estimation.

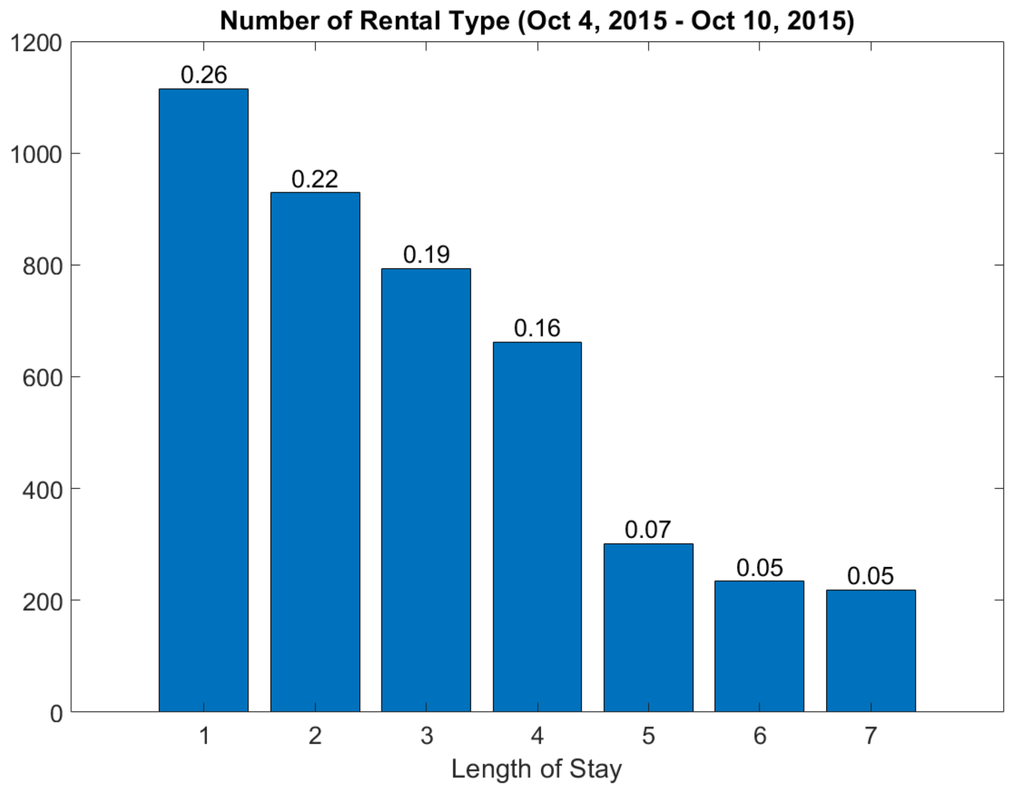
Moreover, there is substantial variation in the length of stay across the selling period. Figure 10 shows that renters who preferred longer stays would book early while renters preferred shorter stays would book late. The composition of rentals with different length varied substantially across the selling period. For instance, more than 50% of the rentals last from 5 to 7 days when it was two months before the check-in date. However, when the bookings were made one or two weeks before the check-in date, the majority of the rental types were 1 and 2 days. These rental patterns strongly suggest that renters with different stay lengths might have different preferences, therefore, the model used to evaluate automated pricing should take these patterns into account as well.

Table 5: Price Regression

Days to Check-in	(1) No FE	(2) FE	(3) 10th Quantile	(4) 25th Quantile	(5) 50th Quantile	(6) 75th Quantile	(7) 90th Quantile
7	.0312***	.0289***	.0262***	.0399***	.0405***	.0301***	.0089***
	(.0025)	(.0015)	(.0025)	(.0015)	(.0011)	(.0021)	(.0013)
14	.0456***	.0395***	.0411***	.0613***	.0435***	.0596***	.0095***
	(.0029)	(.0012)	(.0025)	(.0033)	(.0012)	(.0010)	(.0032)
21	.0576***	.0461***	.0587***	.0769***	.0452***	.0613***	.0103***
	(.0035)	(.0016)	(.0021)	(.0012)	(.0023)	(.0029)	(.0031)
28	.0687***	.0589***	.0612***	.0775***	.0488***	.0656***	.0112***
	(.0011)	(.0023)	(.0032)	(.0017)	(.0031)	(.0027)	(.0016)
35	.0724***	.0699***	.0765***	.0781***	.0521***	.0671***	.0120***
	(.0021)	(.0031)	(.0023)	(.0027)	(.0028)	(.0032)	(.0031)
42	.0786***	.0785***	.0769***	.0782***	.0532***	.0713***	.0129***
	(.0022)	(.0021)	(.0028)	(.0033)	(.0016)	(.0022)	(.0029)
49	.0822***	.0862***	.0771***	.0831***	.0571***	.0722***	.0196***
	(.0015)	(.0011)	(.0029)	(.0019)	(.0024)	(.0012)	(.0035)
56	.0895***	.0912***	.0789***	.0946***	.0589***	.0796***	.0265***
	(.0028)	(.0008)	(.0011)	(.0039)	(.0029)	(.0022)	(.0034)
Constant	3.1254***	3.2112***	3.6133***	4.661***	4.1331***	4.8961***	4.8322***
	(.0021)	(.0189)	(.0021)	(.0025)	(.0012)	(.0019)	(.0010)
Adj. R^2	.3046	.7611	-	-	-	-	-
Other Controls	Room Type, Amenities, Photo, Reviews etc.						
Obs.	1,302,672						

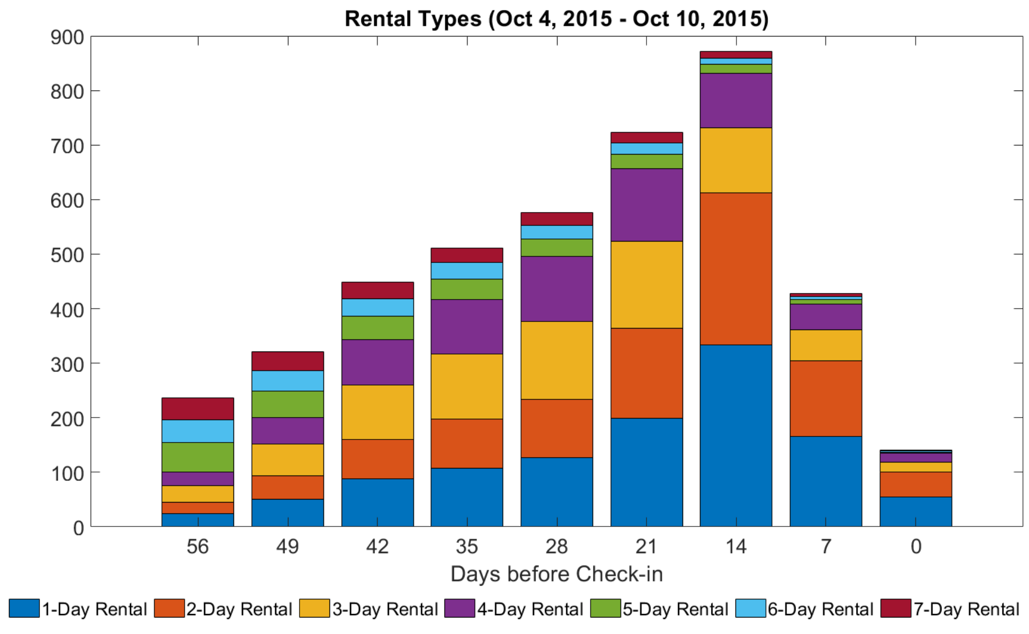
Price is log price. One observation is the combination of a listing, a check-in date and number of week(s) to check-in date

Figure 9: Number of Rentals



Note: Y axis is the number of rentals. The number on top of each bar represents the fraction of each rental type

Figure 10: Rentals across Time



Note: Y axis is the number of rentals.

5 An Empirical Model of Rental Choice and Price Setting

In the previous section, I showed that some hosts do not adjust their prices frequently and that the overall trend of price is decreasing. Moreover, the rental patterns indicate that there is substantial variation in the length of rentals across the selling period. In order to capture these features, I build a dynamic pricing model with price adjustment cost. There are several challenges in the model setup. First, each listing can only provide one unit of rental for a given check-in date. Therefore, I need to impose more structure on the choice process to solve the listing assignment problem. Second, the competition among hosts causes the dimension of the dynamic pricing problem to grow exponentially with the number of listings in the market. In order to reduce the dimension, I assume that each host only considers her own state and a set of system states, which summarize the information of her competitors, when making pricing decision.

This empirical model allows the identification of the demand with different renal type and the price adjustment cost. Moreover, the model considers factors and their interactions with the price adjustment cost. In particular, those factors are renter's varying willingness to pay and inventory structure; these are the key ingredients in evaluating automated pricing because automated pricing releases the price from the restriction imposed by the cost. Before presenting the model, I first introduce several notations which are related to the timing and the rental structure.

5.1 Setup

I define a check-in week as a calendar week beginning on Sunday and ending on Saturday. In Figure 11, there are four check-in weeks: week 1 to week 4. For a given check-in week, renters can book a listing up to T days before this check-in week. T is defined as the days before the Sunday of the check-in week. For example, in Figure 10, time t is 11 days before check-in week 1 and 18 days before check-in week 2. In general, a rental can be within one check-in week or overlap different check-in weeks. In this model, I assume that if a rental spans over multiple check-in weeks, this rental will be split into multiple rentals and each rental is within a check-in week. This assumption maintains the multi-day rental structure while significantly reducing the computational burden. Moreover, renters are categorized by rental type which is defined by the check-in day and the length of stay. For instance, type 1 renters are those who want to stay

Figure 11: Price Calendar

SUNDAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY	
			1 t	2	3	4	
5	6	7	8	9	10	11	
12 $100	13	14 $100	15 $100	16 $100	17 $120	18 $120	week 1
19 $105	20 $105	21	22 $105	23 $105	24 $125	25 $125	week 2
26 $110	27 $110	28 $110	29	30	31 $130	1 $130	week 3
2	3	4 $115	5 $115	6 $115	7 $135	8 $135	week 4

on Sunday, type 2 renters are those who want to stay on Monday and type 8 renters are those who want to stay on Sunday and Monday (the detail of the rental types is in the Appendix).

5.2 Listing Choice

The model of listing choice characterizes how a renter chooses a listing and how a listing is eventually rented. I assume that a renter's travel date and length of stay are exogenously determined. Because of each host's limited inventory, the rental type can describe how the mix of different renter's willingness to pay varies across the selling period.

Renter's arrival and listing choice t days before a given check-in week, renters with rental type j randomly arrive according a Poisson distribution with parameter λ_t^j. A type j renter selects her preferred listing among all the available listings by solving a static discrete choice problem. The indirect utility for a type j renter to choose listing k at time t is

$$v_{t,k}^j = X_k\beta - \alpha^j P_{t,k}^j + \epsilon_{t,k}^j \tag{2}$$

where X_k includes the observed characteristics of listing k such as amenities, location, rating and photo. $P_{t,k}^j$ is the effective nightly rate of type j rental of listing k at time t. The effective nightly rate $P_{t,k}^j$ is defined as the average nightly rate including the cleaning fee and the service fee paid to Airbnb. $\epsilon_{t,k}^j$ is the unobserved idiosyncratic utility shock that follows and iid type 1 extreme value distribution. If the renter leaves the market without purchase, the utility is normalized to $\epsilon^j_{t,0}$. Let $G^j_{t,k} = \Pr(v^j_{t,k} > \max_{r\neq k}\{v^j_{t,r}\})$ define as the probability that a renter intends to choose listing k, therefore with the logit error assumption

$$G^j_{t,k} = \frac{\exp(v^j_{t,k})}{1 + \exp(v^j_{t,k}) + \sum_{r\neq k}^{N^j_t} \exp(v^j_{t,r})} \tag{3}$$

where N^j_t is the number of all the available listings which can accept type j rental. For instance, if j is a Mon-Wed rental type, a listing with Monday sold is not in N^j_t.

Allocation Mechanism If multiple renters, with the same rental type, prefer the same listing, each of them receives the listing with equal probability. Let $Q^j_{t,k}(M^j_t)$ be the the probability that among all M^j_t potential type j renters, at least one of them is interested in listing k. Then

$$Q^j_{t,k}(M^j_t) = 1 - (1 - G^j_{t,k})^{M^j_t} \tag{4}$$

Hosts do not know the actual number of renters that seeks to book but only the distribution which is assumed to be Poisson. Therefore, $Q^j_{t,k}$, which is $Q^j_{t,k}(M^j_t)$ without conditional on the number of arrived renters, has an analytical form

$$\begin{aligned} Q^j_{t,k} &= \sum_{M^j_t=1}^{\infty} (1 - (1 - G^j_{t,k})^{M^j_t}) \Pr(M^j_t) \\ &= \sum_{M^j_t=0}^{\infty} (1 - (1 - G^j_{t,k})^{M^j_t}) \frac{(\lambda^j_t)^{M^j_t} e^{-\lambda^j_t}}{M^j_t!} \\ &= 1 - e^{-\lambda^j_t(G^j_{t,k})} \end{aligned} \tag{5}$$

It is also possible that renters with different types prefer the same listing. I assume that a host can only process one rental at a time at most. Let $\tilde{g}^j_t$ be the probability that determines how likely type j rental will be processed by a host

$$\tilde{g}^j_t = \frac{1(\text{j is available})}{\sum_{k=1}^{n} 1(\text{k is available})} \tag{6}$$

where n is the number of all possible rental types. The indicator function in Equation (6) guarantees that if type j rental is not available it is excluded from this probability. $\tilde{g}^j_t$ serves two purposes in the model. First, since hosts rarely process more then one rental at a time, this setting is a parsimonious way to determine the rental type that is actually processed by a host. Each possible rental type will be considered by a host with equal probability. Second, with $\tilde{g}^j_t$

I can write down an analytical form of the demand function. Combining Equation (5) and (6), I can define

$$D_{t,k}^{j} := \tilde{g}_t^j Q_{t,k}^j \tag{7}$$

as the probability that a type j rental is realized at listing k. $D_{t,k}^0 = 1 - \sum_{j=1}^{n} D_{t,k}^j$ is the non-rental probability. $D_{t,k}^j$ not only characterizes the rental probability, it is also the transition probability describes how the inventory evolve over time.

5.3 Price Setting by Hosts

The rental probability $D_{t,k}^j$ depends on time, rental type, price, listing characteristics, renter's preference and inventory structure. Given the information of $D_{t,k}^j$, forward-looking hosts maximize the expected revenues of each check-in week in the selling period series. In each of the sequential selling periods, a host can either choose to set a new price or leave the price unchanged from the previous period. However, if the host chooses to adjust her price she incurs a price adjustment cost. I assume that prices take on discrete value. Moreover, there exist price specific idiosyncratic shocks, which are only observable to the hosts. The per-period payoff function is

$$\sum_{j=1}^{n} D_t^j r_t^j$$

where n is the number of all possible rental type within a week, D_t^j is the type j rental probability defined in Equation (7) and r_t^j is the revenue associated with type j rental, with the cleaning fee subtracted. While the cost for renting out an apartment could include cleaning, utilities, maintenance and other related cost, cleaning is the most direct and primary cost associated with each rental. Considering the relatively short length of each stay, I believe that other costs for operating Airbnb listing are in the form of fixed cost. If I assume that the cleaning fee collected by the host is the actual cleaning cost, r_t^j is the profit for a type j rental. Therefore, for a given check-in week, the optimal price setting is summarized by the following Bellman equation

$$\begin{aligned} V_t(s_t, u_t) &= \max_{P_t} \Bigg\{ \sum_{j=1}^{n} \Big[D_t^j(P_t, s_t) r_t^j(P_t, s_t) \Big] \\ &+ E_{s_{t-1}, u_{t-1} | s_t, u_t, P_t} V_{t-1}(s_{t-1}, u_{t-1}) - m_t \cdot 1(P_t \neq P_{t+1}) + u_t(P_t) \Bigg\} \end{aligned} \tag{8}$$

In Equation (8), s_t are the observed state variables. u_t is the price specific idiosyncratic shock which is assumed to be i.i.d with type 1 extreme value distribution and scale parameter σ. m_t is the price adjustment cost and $E_{s_{t-1},u_{t-1}|s_t,u_t,P_t}V_{t-1}(s_{t-1},u_{t-1})$ is the expected continuation value after type j rental is realized and the expectation is over (s_{t-1},u_{t-1}) conditional on (s_t,u_t,P_t).

5.3.1 State Variables

Since the number of listings is large, one host's behavior would have little impact on others. Similar to the idea of oblivious equilibrium developed by Weintraub et al. (2008), I assume that each host's decision is based on her own state and a set of system states which evolve deterministically. In Equation (3), the probability for choosing listing k is

$$G^j_{t,k} = \frac{\exp(v^j_{t,k})}{1+\exp(v^j_{t,k})+\sum_{r\neq k}^{N^j_t}\exp(v^j_{t,r})}$$

if the rental type is j. Let $ss^j_{t,k} = \log(\sum_{r\neq k}^{N^j_t}\exp(v^j_{t,r}))$. In my application, N^j_t is a large number, thus $ss^j_{t,k}$ would be "almost" invariant across k. Hence, I can rewrite Equation (3) as

$$G^j_{t,k} = \frac{\exp(v^j_{t,k})}{1+\exp(v^j_{t,k})+\exp(ss^j_t)} \tag{9}$$

where ss^j_t is defined as the system state of type j rental. Hosts only consider their own states and the system states $\mathbf{ss_t} = \{ss^j_t\}^n_{j=1}$ which summarize the number of the competitors and their price choices. The advantage of this simplification is that while the model becomes tractable, competition is not entirely ignored.

With this simplification, a host's state variables (s_t,u_t) can be decomposed into $(P_{t+1},a_{t+1},\mathbf{ss_t},u_t)$. P_{t+1} is the host's own price at time $t+1$; and it affects the host's pricing decision through the price adjustment cost. For instance, a host will change her price only when the revenue improvement (over not adjusting the price) exceeds the price adjustment cost. a_{t+1} is the inventory of a listing. It describes availability status of each day within a check-in week. If a listing can only be booked for Sunday, a_{t+1} can be represented by a vector $(1,0,0,0,0,0,0)$ meaning that Sunday is bookable (1) but all the other days are not (0). Once a_{t+1} becomes $(0,0,0,0,0,0,0)$, the value function $V_t(s_t,u_t)$ is equal to zero. In addition, when t reaches zero $V_0 = 0$ – unsold inventory after the check-in date represents no revenue opportunity.

5.3.2 State Transition

Let $\nu_t(s_{t-1}, u_{t-1} \mid s_t, u_t, P_t)$ be the transition probability that describes a host's belief about how the states evolve over time. I assume that conditional independence is satisfied (Rust 1987), meaning that $\nu_t(s_{t-1}, u_{t-1} \mid s_t, u_t, P_t) = \omega(u_{t-1}) g_t(s_{t-1} \mid s_t, P_t)$. ω is the density function of the idiosyncratic shock u_{t-1} and g_t is the transition probability of the observed states s_{t-1} given s_t and P_t. This probability can be further decomposed as

$$\begin{aligned} g_t(s_{t-1} \mid s_t, P_t) &= g_{1t}(\mathbf{ss_{t-1}} \mid a_t, s_t, P_t) g_{2t}(a_t \mid s_t, P_t) g_{3t}(P_t \mid s_t, P_t) \\ &= g_{1t}(\mathbf{ss_{t-1}} \mid \mathbf{ss_t}) g_{2t}(a_t \mid s_t, P_t) \end{aligned} \tag{10}$$

The second equality in Equation (10) is due to the facts that the system states are not affected by individual's behavior and P_t is deterministic given P_t. Since the system states evolve deterministically, the path of $\mathbf{ss_t}$ is perceived by all the hosts. Introducing stochastic system states makes the model computationally intensive, and hence I opt to maintain a simple structure. Given this transition structure, $g_{2t}(a_t \mid s_t, P_t)$ is then determined by the rental probabilities $\{D_t^j\}_{j=1}^n$. Equation (8) can be written as:

$$V_t(s_t, u_t) = \max_{P_t} \left\{ \sum_{j=1}^{n} D_t^j(P_t, s_t) \left[r_t^j(P_t, s_t) + E_{u_{t-1}} V_{t-1}(s_{t-1}, u_{t-1}) \right] - m_t \cdot 1(P_t \neq P_{t+1}) + u_t(P_t) \right\} \tag{11}$$

5.4 Equilibrium

A host's pricing decision depends on her own states, a set of system states and beliefs about how the states will change in the future. The host's own states include her inventory status and the price she set in last period. The system states summarize all relevant information about the competition. I can define an equilibrium under this structure as follows:

Definition. *The equilibrium consists of the demand probabilities $\{D_{t,k}\}$, the pricing functions $\{P_{t,k}\}$, the initial availability vectors $\{a_{T+1,k}\}$ and a perceived path of system state $\{s\hat{s}_t\}$ such that*

1) Given the current period availability vector $\{a_{t+1,k}\}$, last period price $\{P_{t-1,k}\}$ and the perceived system state $\{s\hat{s}_t\}$, each listing owner k chooses her price according to the pricing function $\{P_{t,k}\}$ which solve Equation (11)

2) In each period t, listings are rented according to the demand probability $\{D_{t,k}\}$. $\{D_{t,k}\}$ also determine the next period availability vector $\{a_{t,k}\}$.

3) $\{a_{t+1,k}\}$ *determine the number of players in each rental type* j *and* $\{P_{t,k}\}$ *determine the pricing decision. The aggregate movement of* $\{a_{t+1,k}\}$ *and* $\{P_{t,k}\}$ *imply the actual system state* $\{ss_t\}$*:* $ss_t^j = \log(\sum^{N_t^j} \exp(v_{t,r}^j))$ *which should be consistent with the perceived* $\{\hat{ss}_t\}$

Existence of this equilibrium is a direct consequence of the finite horizon and finite action-space (Maskin and Tirole 2001). The equilibrium determines the evolution of the system state such that each host's pricing decision is compatible with the system state. This equilibrium concept permits the calculation of the optimal price under different counterfactual scenarios.

6 Identification and Estimation

6.1 Identification

The parameter set can de decomposed into demand parameters $\theta_1 = \{\beta, \{\alpha_t^j\}, \{\lambda_t^j\}\}$ and supply parameters $\theta_2 = \{\{m_t\}, \sigma\}$. In order to take full advantage of the data variation and reduce the number of parameters, I impose additional restrictions on the parameters. Given rental type j, each of $\{\{\alpha_t^j\}, \{\lambda_t^j\}, \{m_t\}\}$ can only take four values depending on the number of days before the check-in date. For example:

$$m_t = \begin{cases} m_1 & \text{less than 7 days before check-in} \\ m_2 & \text{7 to 21 days before check-in} \\ m_3 & \text{21 to 35 days before check-in} \\ m_4 & \text{more than 35 days before check-in} \end{cases} \tag{12}$$

Given time t, each of $\{\{\alpha_t^j\}, \{\lambda_t^j\}\}$ can takes 7 values which are based on the length of stay (within a week, there can be 7 different length of stay).

The key challenge for identification is separately identifying the demand parameters from the arrival process parameters λ_t^j. Without search data to pin down the arrival process, an increase in arrivals could instead be seen as a change in the demand (Talluri and Van Ryzin 2004, Williams 2018). For instance, a listing was rented could be the result of large number of arrivals with low willingness to pay or small number of arrivals with high willingness to pay. However, I argue that with the pricing and rental information across different check-in weeks I can use the mean and variance of the rental probability to identify the demand parameters and the arrival process.

In the model, $Q^j_{t,k}(M^j_t)$ is the probability that at least one of the M^j_t arrived renters is interested in listing k. Equation (4) denotes that $Q^j_{t,k}(M^j_t)$ has the following form.

$$Q^j_{t,k}(M^j_t) = 1 - (1 - G^j_{t,k})^{M^j_t}$$

$G^j_{t,k}$ is defined in Equation (3) and is the probability for choosing listing k. Since the number of arrived renters M^j_t follows a Poisson distribution with parameter λ^j_t, I can calculate the mean and variance of $Q^j_{t,k}(M^j_t)$:

$$\begin{aligned} E(Q^j_{t,k}(M^j_t)) &= 1 - \exp\left(-\lambda^j_t G^j_{t,k}\right) && (13) \\ Var(Q^j_{t,k}(M^j_t)) &= \exp\left(-2\lambda^j_t G^j_{t,k}\right)\left(\exp\left(-\lambda^j_t (G^j_{t,k})^2\right) - 1\right) && (14) \end{aligned}$$

λ^j_t and $G^j_{t,k}$ can be uniquely pinned down by $E(Q^j_{t,k}(M^j_t))$ and $Var(Q^j_{t,k}(M^j_t))$ through Equation (13) and (14). Therefore, I can separately identify the demand parameters from the arrival process.

Finally, the identification of σ (the variance of the idiosyncratic shock to the host) relies on the price variance among those listings with similar observed characteristics and states. The identification of $\{m_t\}$ (the price adjustment cost) comes from the price adjustment likelihood and the magnitude of the adjustments. The changes in the demand and time lead to price adjustments only when they are sufficient to justify the price adjustment cost. In the data, even though some hosts do not have many price adjustments, the total number of price adjustments during each of the four time periods described in Equation (12) is sufficient to identify $\{m_t\}$. In practice, I did not encounter any issues in identifying the price adjustment cost parameters.

6.2 Estimation

In order to avoid searching over a large set of parameters while solving the dynamic pricing problem, I decide to split the estimation process into two steps. In the first step, I estimate the demand parameters. Once the parameters from the demand function are recovered, I estimate the price adjustment cost in the dynamic pricing step. The estimation of the price adjustment cost is challenging. First, the decision variable, price, is a vector of size 7 since each host needs to determine a price for each day within a week. I impose additional restriction on the price vector and reduce to dimension from 7 to 2. Second, no two listing are the same, which means the value function is listing specific. In order to reduce the number of the value functions, I employ a clustering method to group similar listings together.

6.2.1 Step 1: Demand Estimation

Let $y_{t,k}^j$ denote the type j rental indicator for listing k at time t. If $y_{t,k}^j = 1$, it means listing k is rented at selling period t and the rental type is j. Let $y_{t,k} = (y_{t,k}^0, y_{t,k}^1, ..., y_{t,k}^n)$ be the rental indicator vector. Since each host can only process one rental type in each selling period, $\sum_{j=1}^n y_{t,k}^j = 1$. $a_{t,k}$ is the inventory at selling period t. The likelihood function on rental incident is

$$\begin{aligned}
L_k &= P(y_{\tau_k,k}, y_{\tau_k+1,k}, ..., y_{T,k}|a_{T+1,k}; \theta_1) \\
&= P(y_{\tau_k,k}|a_{T+1,k}, y_{\tau_k+1,k}, ..., y_{T,k}; \theta_1) P(y_{\tau_k+1,k}|a_{T+1,k}, y_{\tau_k+2,k}, ..., y_{T,k}; \theta_1)...P(y_{T,k}|a_{T+1,k}; \theta_1) \\
&= P(y_{\tau_k,k}|a_{\tau_k+1,k}; \theta_1) P(y_{\tau_k+1,k}|a_{\tau_k+2,k}; \theta_1)...P(y_{T,k}|a_{T+1,k}; \theta_1) \\
&= \prod_{t=\tau_k}^{T} \prod_{j=1}^{n} \left(D_{t,k}^j\right)^{y_{t,k}^j} \left(D_{t,k}^0\right)^{y_{t,k}^0} \qquad (15)
\end{aligned}$$

where $D_{t,k}^j$ is defined in Equation (7), $D_{t,k}^0 = 1 - \sum_{j=1}^n D_{t,k}^j$ and θ_1 is the set of demand parameters. τ_k is the time when listing k is sold out or the check-in date approaches.

One potential problem in the demand estimation is that the effective nightly rate $P_{t,k}^j$ is potentially correlated with some unobserved factors which are included in the error $\epsilon_{t,k}^j$ in Equation (2). I deal with the price endogeneity by including a rich set of attributes and extra information about the listings. In the data set, I can observe all the displayed attributes of a listing. In addition to the observed attributes, I supplement with the information such as the age of the listing and the estimated house value provided by Infogroup. Even though I use a rich set of attribute to alleviate price endogeneity, it cannot be ruled out since no attribute set is all-encompassing. However, most of the existing econometric methods dealing with price endogeneity is not feasible in my model. First, the Berry et al. (1995) contraction mapping is not applicable because of the inventory structure of each listing. Given rental type j, each host only has one unit of "product" to sell. Therefore, the empirical market share would be extremely noisy with a large number of zeros. Second, while control function approach could potentially solve this issue, this method introduces an extra random term to each listing choice. If the choice set is large, which is true in this paper, the integration over the extra random terms becomes infeasible[9]. As a robustness check for the performance of the attributes I include in the model, I estimate two linear probability models – one with individual fixed effect and the other one without individual fixed but with a larger set of attributes. The result (shown in the Appendix) indicates that after controlling for enough attributes, the price endogeneity might be alleviated.

6.2.2 Step 2: Pricing

Once the demand parameters θ_1, are recovered, I estimate the price adjustment cost. In order to make this estimation feasible I need to impose further structure on the problem. In the problem, as formulated above, price is of dimension 7, namely there is a different price for each day of the week. This makes the problem very difficult to solve so in order to simplify the calculation, I assume that there are only two prices within a week: the weekday price and the weekend price. Equivalently, a listing owner is choosing a weekday average price and a weekend average price instead of selecting a price for each day.

Although the dimension of the decision variable is reduced, the large attribute space is also problematic, especially when the demand function, the cleaning fee and price adjustment cost are host specific. In the sample, there are 7662 different listings, which means I have to solve 7662 different dynamic programming problems. To overcome the problem caused by the idiosyncrasy of the listings, I apply a clustering method using k-means based on each neighborhoods listing characteristics. I categorize each listing into one of the 252 clusters (9 clusters in each of the 28 neighborhoods).

Based on the clustering result, the price adjustment cost has the form $m_t = \bar{m}_t \bar{P}$, and $\bar{P}$ is a price index that is calculated as the average price in each cluster. In order to estimate the price adjustment cost, Let

$$v_t^{P_t} = \sum_{j=1}^{n} D_t^j(P_t, s_t) \left[r_t^j(P_t, s_t) + E(V_{t-1}(s_{t-1}) \mid s_t, j) \right] - m_t \bar{P} \cdot 1(P_t \neq P_{t+1}) \tag{16}$$

be the choice specific value function. If P_t can take H_t different values, then the expected value function can be expressed as

$$EV_t(s_t) = \sigma \log \sum_{h=1}^{H_t} \exp\left(\frac{v_t^h}{\sigma}\right) + \sigma\gamma$$

[9]If there are 100 listings, then I need to take integration over 100 random variables.

where γ is the Euler constant. And the choice probability for the h^{th} price is

$$\frac{\exp(\frac{v_t^h}{\sigma})}{\sum_{r=1}^{H_t}\exp(\frac{v_t^r}{\sigma})} \tag{17}$$

The log-likelihood function for the pricing decision can be written as

$$\begin{aligned} LL_k &= \log \Pr(P_{\tau_k,k}, P_{\tau_k+1,k}, ..., P_{T,k}, P_{T+1,k} | \{a_{r,k}\}_{r=\tau_k}^{T+1}; \theta_2) \\ &= \log \left(\prod_{t=\tau_k}^{T-1} \Pr(P_{\tau_k,k} | P_{\tau_k+1,k}, a_{\tau_k+1,k}; \theta_2) \Pr(P_{T,k} | a_{T+1,k}; \theta_2) \right) \\ &= \sum_{t=\tau_k}^{T-1} \left\{ \left(v_t^{P_{t,k}}/\sigma - \log \left[\sum_{h=1}^{H_t} \exp(v_t^h/\sigma) \right] \right) \right\} + v_T^{P_{T,k}}/\sigma - \log \left[\sum_{h=1}^{H_T} \exp(v_T^h/\sigma) \right] \end{aligned} \tag{18}$$

where the singled-out term in the last line

$$v_T^{P_{T,k}}/\sigma - \log \left[\sum_{h=1}^{H_T} \exp(v_T^h/\sigma) \right] \tag{19}$$

represents the choice probability for price P_T. The initial price P_T is called the "base price" which reflects a listing owner's belief about the value of her listing and is set before the selling period. In the estimation, P_T is the price that maximizes the value function, assuming that prices are identical for each selling period and the idiosyncratic shock u_t only appears in period T. Since this is a finite period problem, backward induction can be employed. Further detail of the estimation is provided in the Appendix.

7 Estimation Results and Model Fit

The parameters are estimated using the estimation sample with 7,662 listings, and the model fit simulation is based on the same sample. In the estimation step, the system states ss_t, which summarize hosts' pricing decisions and reduce the dimension of the pricing problem, is assumed to be deterministic. However, in the model fit simulation, since the system states should be consistent with the pricing decision of all the hosts, I use a backward induction and forward simulation method to update the system state. In this section, I presents the results of the parameter estimation and the model fit.

7.1 Parameter Estimates

In the demand estimation I assume that rental types that have the same length of stay share the same parameters. This simplification significantly reduces the number of parameters while maintains the necessary renter segmentation. I first show the estimation results of the arrival process in Table 5. All parameters are significant at the 1% level. Each column in Table 5 shows the estimated parameters λ_t^j that govern the Poisson arrival processes. For instance, in the first column, λ $(0, 7]$ is average number of arrivals per day for all one-day rental from 1 to 7 days prior to the check-in date. The parameter estimates suggest substantial variations in the arrival rate across different length of stay and time. Renters who are interested in longer stay tend to arrive early to the booking. This is compatible with the facts that 1) longer stay involves more complicated planning such as flight ticket booking and 2) the number of hosts who can accommodate long stay renters is decreasing thus it is harder for those renters to find their desired listings if they arrive late. Shorter stay renters make their booking decision relatively late. Moreover, one week before check-in (first row in Table 5), the arrival number for 1-day rental is about 128 per day (18*7) which is about 16 times higher than the number for 7-day rental. However, the daily arrival number for 1-day rental is only one half of the 7-day rental two months prior to the check-in.

In Table 6, I show the estimated price coefficients. The price coefficients are all significant and of the expected sign. The implied demand elasticities range from 1.82 to 4.66 under the estimate in Table 6. The result in this paper is in the reasonable range[10]. Long stay renters are more price sensitive than short stay renters. Turning to the intertemporal change in the price coefficients, renters are significantly more price sensitive when it is more than 5 weeks away from the check-in date than they are when it is only a week before the check-in date. Even though renters are becoming less price sensitive as time approaches to the check-in date, prices are still decreasing. This indicates that other factors such as price adjustment cost, the diminishing opportunities to sell and the inventory structure are crucial in the analysis, otherwise less price sensitive demand should suggest price increase. As for other attributes of the listings, renters prefer listings with higher rating, more reviews, less restrictive cancellation policies and more photos. The details of other parameters in the demand function are in the Appendix.

With the demand parameters in hand, I estimate the price adjustment cost. In principle,

[10]Also see Cho et al. (2018) for hotel pricing, Williams (2018) for Airline pricing and Li & Srinivasan (2018) for hotel and Airbnb pricing

Table 6: Parameters of the Arrival Process

	one-day	two-day	three-day	four-day	five-day	six-day	seven-day
λ (0,7]	18.37***	16.71***	14.52***	12.34***	7.46***	7.36***	8.26***
	(.86)	(.79)	(.69)	(.96)	(.59)	(.43)	(.67)
λ (7,21]	25.73***	22.95***	28.33***	27.26***	11.12***	14.42***	26.75***
	(.88)	(.73)	(1.12)	(.96)	(.63)	(.84)	(.90)
λ (21,35]	16.44***	16.81***	24.88***	24.71***	14.36***	17.90***	34.69***
	(1.06)	(1.13)	(.84)	(1.18)	(.77)	(.76)	(1.34)
λ (35,56]	3.14***	7.15***	9.23***	9.74***	17.13***	20.73***	44.75***
	(.22)	(.46)	(.63)	(.81)	(1.19)	(.87)	(1.23)

The arrival process parameter λ can vary across selling period and length of stay. Given length of stay, λ can take 4 values based on the days prior to check-in: 0-7 days, 8-21 days, 22-35 days and 35-56 days. *90% **95% ***99%.

Table 7: Demand Estimation (price coefficients)

	one-day	two-day	three-day	four-day	five-day	six-day	seven-day
price (0,7]	-.0197***	-.0216***	-.0338***	-.0463***	-.0528***	-.0601***	-.0605***
	(.0008)	(.0018)	(.0021)	(.0012)	(.0011)	(.0015)	(.0016)
price (7,21]	-.0219***	-.0256***	-.0362***	-.0168***	-.0536***	-.0625***	-.0636***
	(.0007)	(.0011)	(.0021)	(.0018)	(.0009)	(.0020)	(.0021)
price (21,35]	-.0242***	-.0265***	-.0384***	-.0473***	-.0544***	-.0635***	-.0652***
	(.0011)	(.0015)	(.0007)	(.0012)	(.0015)	(.0018)	(.0010)
price (35,56]	-.0292***	-.0303***	-.0413***	-.0503***	-.0586***	-.0688***	-.0692***
	(.0010)	(.0011)	(.0008)	(.0012)	(.0013)	(.0018)	(.0019)

The price coefficients can vary across selling period and length of stay. Given length of stay, the price coefficients can take 4 values based on the days prior to check-in: 0-7 days, 8-21 days, 22-35 days and 35-56 days. *90% **95% ***99%.

each host should have her own price adjustment cost. The price index $\bar{P}$ serves as a tool to reduce the dimensionality caused by the host specific price adjustment cost. Therefore the cost m_t is in the form of $\bar{m}_t\bar{P}$ and $\bar{m}_t$ is invariant across hosts. Under this setting, all the hosts within the same cluster have the same price adjustment cost while this cost varies across different clusters. presents the estimated price adjustment cost. In Table 7, the cost is about 2% of the price index two months before the check-in date, and it drops to under 1% of the price index one week before the check-in date. This result indicates that hosts are more likely to adjust their prices if the check-in date is close. If I assume that price index is \$195 which is the average price in the sample, the price adjustment costs are approximately \$1.8 - \$4.2. Although \$1.8 and \$4.0 are not huge numbers compared to the rental price, they generate substantial price stickiness in the data.

7.2 Model Fit

In order to evaluate how well the model fits the data, I simulate the model using the estimated parameters. Unlike the estimation step, where it is not necessary to solve the pricing

Table 8: Price Adjustment Cost and Price shock

	Coef.	Std.	t-stat
Price Adjustment Cost			
$\bar{m}(0,7]$	.0089***	.0023	3.8696
$\bar{m}(7,21]$	.0141***	.0041	3.4390
$\bar{m}(21,35]$	.0175***	.0033	5.3030
$\bar{m}(35,56]$	.0229***	.0052	4.4038
Logit shock			
σ	1.5123***	.0211	71.6730

*90% **95% ***99%.

equilibrium, simulating the model requires solving the value function and the pricing function under different states. One challenge in the simulation is to calculate the path of the system states $\{\mathbf{ss_t}\}$. Although I have assumed that $\{\mathbf{ss_t}\}$ evolve deterministically, the perceived state $\{\mathbf{s\hat{s}_t}\}$ have to be consistent with the actual state $\{\mathbf{ss_t}\}$. In fact, $\{\mathbf{s\hat{s}_t}\}$ can be updated via backward induction and forward simulation. The backward induction is required for the value function and the forward simulation is utilized to calculate the prices and update the system state. The solution method is summarized in Algorithm 1 in the Appendix. I start the algorithm using the system state calculated in the estimation procedure. Then, I calculate the expected value function using backward induction. Once the value function is calculated, I simulate both the prices and the purchase decision. With these two pieces of information, the system state ss_t^j is updated through $ss_t^j = \log(\sum_{r=1}^{N_t^j} \exp(v_{t,r}^j))$

Given the current price, the parameters in the demand model and the price adjustment cost, I simulate 100 cases of pricing and rental decision for the first week in October, 2015 using all 5,343 listings in the sample. In each case, I simulate hosts pricing decisions, renters arrival and their choices in each period. Since price used in the mode is discretized and adjusted to weekday average and weekend average, I transform all the prices using the same procedure. I simulate the model keeping the initial availability of each listing unchanged. In Figure 12. I show the comparison between the data and the model prediction of the number of rentals in each selling period. The model fits the data relatively well. Figure 13 addresses the model fit by comparing mean prices with model predicted prices, by day before check-in. Since I only consider weekday and weekend average price in the calculation, this figure also demonstrates the comparison by weekday and weekend price.

Figure 12: Model Fit of Rental Number

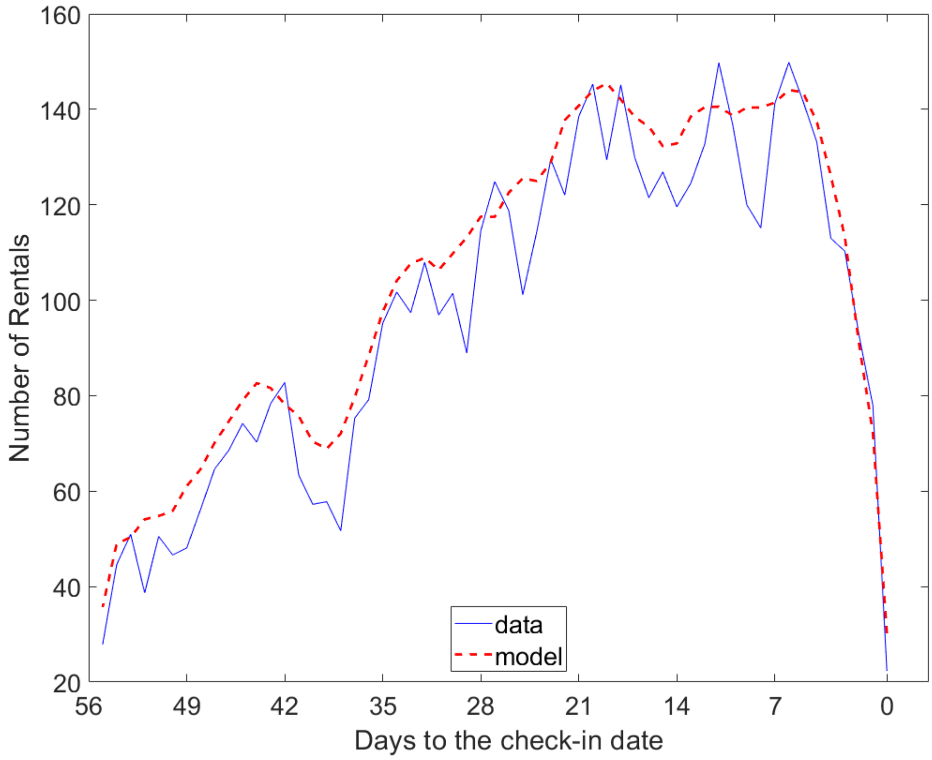
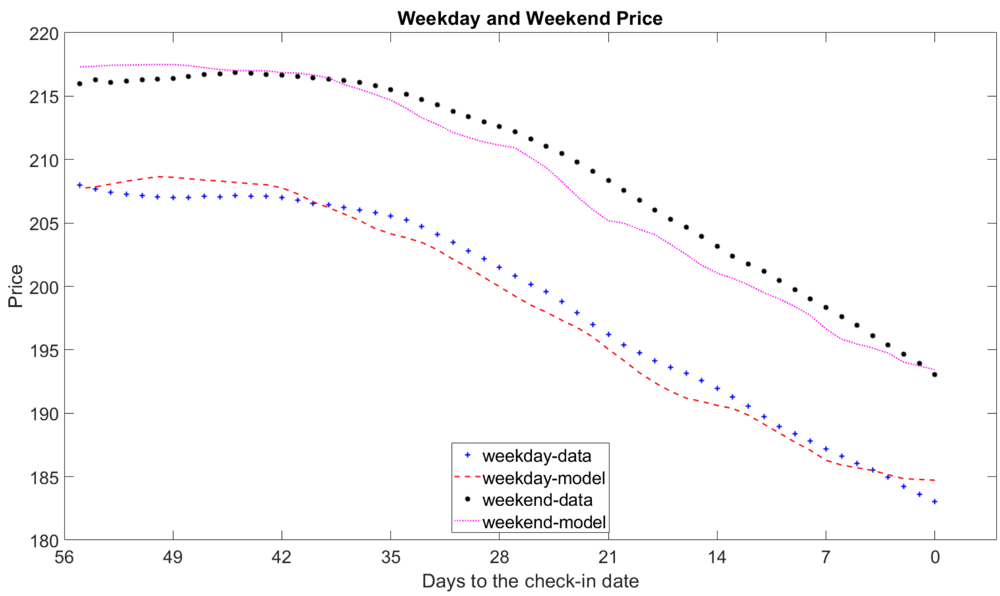


Figure 13: Model fit of average weekday price and weekend price



8 Managerial Implications: Automated Pricing

In this section, I evaluate the revenue and welfare implications of automated pricing using the estimated demand parameters. Automated pricing will influence both the timing and the magnitude of price changes by reducing the price adjustment cost and taking into account renter's varying willingness to pay and inventory structure. When the price adjustment cost exists, hosts's pricing decisions are less responsive to these factors. However, if automated pricing makes the pricing decision, it has the potential to react to any change in the market, therefore the interaction among those factors may suggest a different optimal price path.

I conduct three counterfactual experiments exploring different forms of automated pricing. First, I assume that automated pricing eliminates the price adjustment cost completely. In the second counterfactual, I examine the possibility that automated pricing only reduces the price adjustment cost but does not eliminate it. Therefore, I set the price adjustment cost to half of its original level. Among other reasons, the reason for conducting this experiment is that Airbnb also provide price suggestion for hosts. In the case of price suggestion, hosts still need to adjust their price manually. Therefore, price adjustment cost may still exist, but the cost of price adjustment may be smaller than if there were no pricing suggestion from Airbnb. Finally, in the last experiment, I assume that automated pricing eliminates the price adjustment cost but adjusts the price every week instead of every day during the selling period. Under this setting, the price adjustment frequency is 8 during the selling period. The purpose of this experiment is to examine whether daily price adjustment is necessary for automated pricing.

8.1 No Price Adjustment Cost

In this counterfactual, I assume that automated pricing eliminates the price adjustment cost entirely. Although this counterfactual does not replicate the exact algorithm implemented by Airbnb, it provides important insight into how automated pricing can help hosts set their prices and the potential consequences of using such a pricing tool. From the platform's perspective, if hosts' prices are adjusted more frequently and it results in higher revenue, the platform should put more effort to encourage the use of automated pricing, such as, for example, promoting "smart pricing" more vigorously or sending out daily reminder about price checking to the hosts.

In order to simulate the counterfactual result, I follow a modified version of Algorithm 1 (in the Appendix) with random initial availability. Since hosts' prices can be changed without any friction in each period, the price in the previous period is no longer a state variable in the problem. Therefore, the expected value function becomes

$$E_{u_t}V_t(s_t,u_t) = \max_{P_t}\left\{\sum_{j=1}^{n} D_t^j(P_t,s_t)\left[r_t^j(P_t,s_t) + E_{u_{t-1}}V_{t-1}(s_{t-1},u_{t-1})\right] + u_t(P_t)\right\} \tag{20}$$

The solution method for this problem is described in Algorithm 2 in the Appendix.

Compared with Algorithm 1, Algorithm 2 does not need to keep track of the price P_{t+1} in period t. In addition, the initial price P_{T+1} is not necessary in the simulation. In this frictionless regime, I simulate the optimal price path without the price adjustment cost for each listing as well as the purchase decisions. With the simulated price path and purchase pattern, I calculate the revenue of the hosts and the platform. I also calculate the surplus of the renters.

Table 8 presents the simulation results. When price adjustment cost is turned off, the average weekly revenue of the hosts is increased by approximately 3.9%. The platform also benefits from the elimination of the friction, and its revenue rises by 4.8%. There are two reasons why the revenue improvement for the platform is greater than the hosts. First, the price adjustment cost does not enter into the platform's revenue. Second, the cleaning fee is included in the platform's revenue but not in the hosts' since I assume that the cleaning fee is used to cover the actual cleaning cost for the hosts. The results in Table 8 also show that the utilization rate, defined as the number of booked days over the number of available days, increases from 31% to 36%.

The Airbnb's and the hosts' revenues both increase if automated pricing is employed. However, the impact of automated pricing on the renters depends on the length of their stays, or more precisely, when the renters arrive. In Table 8, consumer surplus for renters who only stay for one or two days increase by about 4.8% and 3.8% respectively. Renters who stay for 5 or more days are better off under the automated pricing. For those renters stay for 3 or 4 days, they are worse off compared to the case where price adjustment is costly.

These interesting findings are the direct result of the new optimal price path implied by automated pricing. Figure 14 shows the simulation results of the prices. One interesting pattern observed in this experiment is that the elimination of the price adjustment cost generates a price path that first increases then decreases. This result is different from most empirical papers on dynamic pricing – the realization of the price path is either increasing or decreasing. At the

Table 9: Revenue & Surplus: Case 1

	With Price Adjustment Cost	Without Price Adjustment Cost	Change
Host Profit	\$432.2 (\$401.3, \$473.5)	\$449.1 (\$421.3, \$499.3)	+3.9%
Platform Revenue	\$73.5 (\$62.1, \$82.7)	\$77.0 (\$69.4, \$88.6)	+4.8%
Utilization Rate	0.31 (0.27, 0.36)	0.36 (0.30, 0.39)	+16.1%
CS_1	\$35.3 (\$28.2, \$45.8)	\$37.0 (\$30.1, \$49.2)	+4.8%
CS_2	\$63.2 (\$55.4, \$70.7)	\$65.6 (\$58.4, \$75.3)	+3.8%
CS_3	\$100.2 (\$90.3, \$119.3)	\$97.3 (\$87.5, \$109.1)	-2.9%
CS_4	\$121.1 (\$108.9, \$133.1)	\$118.4 (\$101.7, \$127.2)	-2.2%
CS_5	\$156.3 (\$132.9, \$172.6)	\$159.1 (\$135.3, \$175.0)	+1.8%
CS_6	\$192.9 (\$173.6, \$212.5)	\$196.7 (\$177.5, \$218.6)	+2.0%
CS_7	\$205.2 (\$182.2, \$243.1)	\$210.0 (\$190.2, \$255.0)	+2.3%

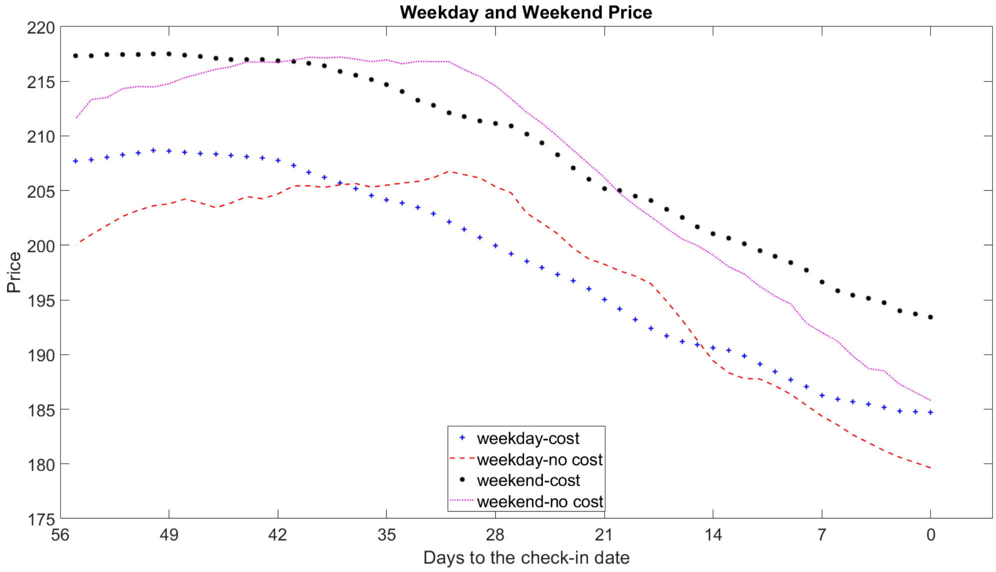
Calculation is based on 100 simulations. Revenue is the revenue for a check-in week. CS_i is the per-person consumer surplus for rentals with length of i. The 95% confidence interval is in the parenthesis.

beginning of the selling period (two months away), most of the renters are those who would stay for a relatively long time. These renters' price elasticity is higher than those who want shorter stays. As time approaches closer to the check-in date, more short stay renters arrive. The optimal prices account for this trend – the mixed price elasticity is decreasing which creates room for price hike. On the other hand, hosts' expectations for a future rental opportunity is decreasing. This force will pull down the price. In addition, short rentals may create "orphan days" that are less likely to be rented. Therefore, even though the inventory of a host is decreasing she might still want to lower down her price to account for those "orphan days". If the mixed price elasticity is not decreasing fast enough, the optimal price should drop.

8.2 Half Price Adjustment Cost

In the previous section, I assume that automated pricing eliminates the price adjustment cost. In practice, in addition to automated pricing, Airbnb also provides price suggestions to the hosts. This price suggestion can be treated as a semi-automated pricing tool. In this section, I assume that the price friction is reduced to half of its current level. The expected value function

Figure 14: Average Price: Case 1



becomes

$$V_t(s_t,u_t) = \max_{P_t}\left\{\sum_{j=1}^{n} D_t^j(P_t,s_t)\left[r_t^j(P_t,s_t) + E_{u_{t-1}}V_{t-1}(s_{t-1},u_{t-1})\right] - \frac{1}{2}m_t \cdot 1(P_t \neq P_{t+1}) + u_t\right\}$$

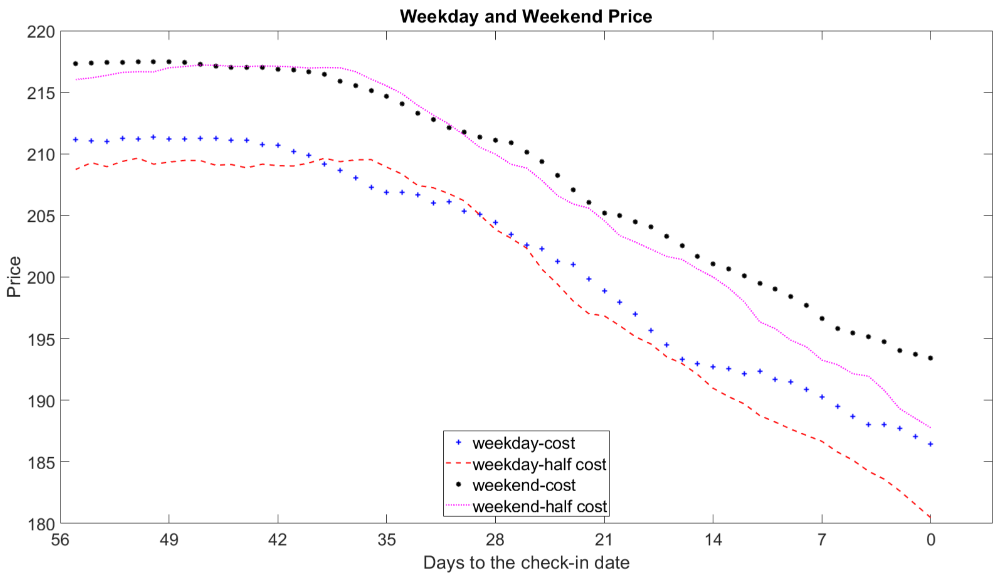
Since the price adjustment cost is not completely eradicated, the solution method to simulate this counterfactual is Algorithm 1 (in the Appendix). Table 9 shows that the revenues of the hosts and the platform are slightly improved under this experiment. The revenue of host increases by 1.3% and the revenue of the platform increases by 1.9% if price adjustment cost is reduced by 50%. Meanwhile, only the renters who are seeking 4-day or 5-day rental are worse off if the price adjustment cost is reduced to half of its original level. Figure 15 is the average weekday price and weekend price. Since the price adjustment cost is not eliminated entirely, the hosts still fail to fully respond to the change in the price elasticity especially the decreasing trend at the beginning of the selling period. Furthermore, hosts also fail to reduce their prices sufficiently to accommodate the forces that are driving down the prices. Although price suggestions may not necessary reduce the price adjustment cost by 50%, this experiment provides evidence that if automated pricing does not eliminate the price adjustment cost entirely the revenue improvement might be limited.

Table 10: Revenue & Surplus: Case 2

	With Price Adjustment Cost	Half Price Adjustment Cost	Change
Host Profit	\$432.2 (\$401.3, \$473.5)	\$437.8 (\$412.4, \$484.1)	+1.3%
Platform Revenue	\$73.5 (\$62.1, \$82.7)	\$74.9 (\$67.1, \$86.7)	+1.9%
Utilization Rate	0.31 (0.27, 0.36)	0.34 (0.26, 0.37)	+9.5%
CS_1	\$35.3 (\$28.2, \$45.8)	\$35.7 (\$29.3, \$47.6)	+1.1%
CS_2	\$63.2 (\$55.4, \$70.7)	\$64.5 (\$57.1, \$74.5)	+2.1%
CS_3	\$100.2 (\$90.3, \$119.3)	\$101.1 (\$92.3, \$118.2)	+0.9%
CS_4	\$121.1 (\$108.9, \$133.1)	\$119.6 (\$103.6, \$129.3)	-1.2%
CS_5	\$156.3 (\$132.9, \$172.6)	\$155.0 (\$130.8, \$171.3)	-0.8%
CS_6	\$192.9 (\$173.6, \$212.5)	\$194.8 (\$176.1, \$219.7)	+1.0%
CS_7	\$205.2 (\$182.2, \$243.1)	\$208.0 (\$187.7, \$251.3)	+1.4%

Calculation is based on 100 simulations. Revenue is the revenue for a check-in week. CS_i is the per-person consumer surplus for rentals with length of i. The 95% confidence interval is in the parenthesis.

Figure 15: Average Price: Case 2



8.3 Weekly Price Adjustment

In this section, I assume that automated pricing eliminate the price adjustment cost. However, instead of changing host's price daily, automated pricing adjusts host's price every week during the selling period. For instance, during the period between 14 days to 7 days prior to the check-in date, automated pricing sets a fixed price for this 7-day period. Ideally, automated pricing should adjust prices as frequently as necessary. However, if weekly price adjustments can achieve revenue and welfare results that are similar to those produced by daily price adjustment, weekly price adjustment might be sufficient for automated pricing. In addition, computation and maintenance cost of automated pricing may make weekly price adjustment attractive to the platform.

I simulate the dynamic pricing model without a price adjustment cost and update the listing price every week during the selling period. In Table 10, I shows the revenue and welfare results under two different price update frequencies. When the price is updated weekly during the selling period is updated weekly, the host's revenue is slightly less than the revenue produced by daily update. However, the revenue under these two cases are close to each other based on the 95% confidence interval. The platform's revenue is also higher under daily price update, but the difference is small. If I compare the revenue under weekly automated pricing with the revenue under the baseline model (price adjustment is costly), the result shows that weekly automated pricing increases the revenue of host by 3.4% and the platform by 4.1%. Similar to the daily automated pricing, weekly automated pricing increases the consumer surplus for renters who do not stay for 3 or 4 days.

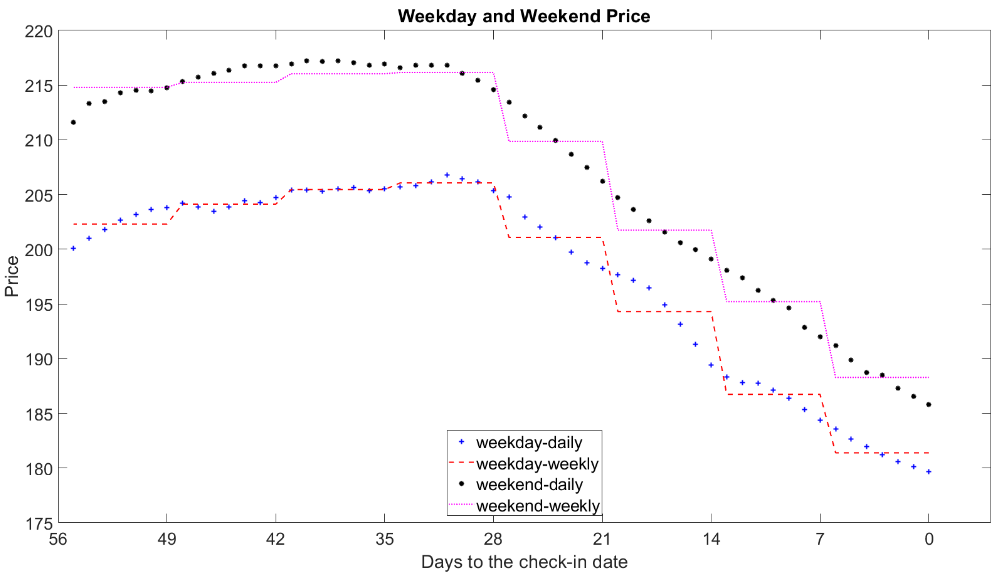
In the first experiment, automated pricing induces a hump shaped price path. In Figure 16, one can see that the average weekend price and weekday price under weekly automated pricing also demonstrate a first increasing then decreasing trend. Although weekly automated pricing can captures the overall trend as daily automated pricing, it misses some price movement due to the restriction of price update frequency. However, the revenue and welfare differences between weekly and daily automated pricing is small. This result suggests that if the daily update is costly for the platform, the weekly update can be a good alternative choice.

Table 11: Revenue & Surplus: Case 3

	Daily Price Adjustment	Weekly Price Adjustment	Change
Host Profit	\$449.1 (\$421.3, \$499.3)	\$446.9 (\$429.2, \$487.5)	-0.5%
Platform Revenue	\$77.0 (\$69.4, \$88.6)	\$76.5 (\$71.5, \$86.3)	-0.6%
Utilization Rate	0.36 (0.30, 0.39)	0.35 (0.31, 0.38)	-2.7%
CS_1	\$37.0 (\$30.1, \$49.2)	\$36.1 (\$32.2, \$46.3)	-2.4%
CS_2	\$65.6 (\$58.4, \$75.3)	\$64.2 (\$58.4, \$75.3)	-2.1%
CS_3	\$97.3 (\$87.5, \$109.1)	\$98.9 (\$88.6, \$107.9)	+1.6%
CS_4	\$118.4 (\$101.7, \$127.2)	\$120.3 (\$100.1, \$129.4)	+1.6%
CS_5	\$159.1 (\$135.3, \$175.0)	\$158.2 (\$130.2, \$171.2)	-0.6%
CS_6	\$196.7 (\$177.5, \$218.6)	\$194.2 (\$179.6, \$214.8)	-1.3%
CS_7	\$210.0 (\$190.2, \$255.0)	\$208.9 (\$196.5, \$251.5)	-0.5%

Calculation is based on 100 simulations. Revenue is the revenue for a check-in week. CS_i is the per-person consumer surplus for rentals with length of i. The 95% confidence interval is in the parenthesis.

Figure 16: Average Price: Case 3



9 Concluding Remarks

In this paper, I have taken advantage of the availability of a rich data set for Airbnb that includes all the available characteristics, price trajectory and transaction history. This enables me to explore the pricing behavior of the hosts across the selling period. A large proportion of the listing owners do not change their prices and this is modeled under the combination of intertemporal price discrimination, stochastic demand, competition and price adjustment cost. By using a dynamic pricing model, I estimate that this price adjustment cost is around 0.98% to 2.1% of the price index. Because of the price adjustment cost, hosts' pricing decisions may fail to respond to changes in the market or even to their own states. This implies that automated pricing could potentially improve their revenues by influencing both when to adjust the price and how much to adjust the price.

The counterfactual experiments that simulate automated pricing offer several managerial insights. I discover that automated pricing increases the revenue of Airbnb by about 4.8% and the hosts by 3.9% if the price adjustment cost is eliminated by automated pricing. Renters who book either late or early during the selling period benefit from the automated pricing; specifically, renters with shorter stays (1 or 2 days) tend to arrive late but renters with longer stay (5,6 or 7 days) tend to arrive early. However, renters who arrive around the middle of the selling period (rentals of 3 or 4 days) actually become worse off. Although the information about the exact number of hosts who are using an automated pricing is not accessible, this analysis provides useful and important evidence for supporting automated pricing. If automated pricing is provided in the form of price suggestion, the price adjustment cost is not reduced to zero. In this case, the revenue improvement might be limited. Lastly, the counterfactual experiment suggests that weekly price update for automated pricing could be an alternative choice for the platform if daily updates are costly.

The methodology presented in this study is applicable to other cases where the market is highly decentralized and each seller only has limited items to sell. For instance, in the secondary sport and event ticket market, prices are also decreasing but rigid. In the peer-to-peer rental car market, sellers face similar problem as the hosts on Airbnb. Therefore, the model in this paper can also help model the sellers' pricing behavior in those market and evaluate the impact of automated pricing or other features that will affect sellers' pricing choices.

There are two limitations in this paper. The first one is the price endogeneity. I use a very rich set of characteristics to mitigate this problem, but price endogeneity is yet to be ruled out. In order to address the price endogeneity, valid instrument and implementable method are both required. The second limitation is the assumption about non-strategic renters. The timing of purchase is of paramount importance in various industries. The existence of strategic customers may have a substantial impact on a firm's decision. Because of the complexity of my pricing model, I have to abstract away strategic customers. However, this is absolutely an important direction for future research.

Reference

Ackerberg, D. (2003), " Advertising, Learning, and Consumer Choice in Experience Good Markets: An Empirical Examination, " *International Economic Review*, Vol. 44, No. 3, 1007-1040.

Aguirregabiria, V. (1999), "The Dynamics of Markups and Inventories in Retailing Firms, " *Review of Economic Studies*, 66(2) 275-308.

Bergen, M., Dutta, S., Levy, D., Ritson, M., Zbaracki, M. (2003), "Shattering the Myth of Costless Price Changes, " *European Management Journal*, 21(6) 663-669.

Berry, S., Levinsohn, J. and Pakes, A. (1995), "Automobile Prices in Market Equilibrium, " *Economatrica*, Vol.63, No.4, 841-890.

Besanko, D., Winston, W. L. (1990), "Optimal Price Skimming by a Monopolist Facing Rational Consumers, " *Management Science*, 36(5), 555-567.

Bitran, G. & Mondschein, S. (1997), "Periodic Pricing of Seasonal Products in Retailing, " *Management Science*, 43(1), 64-79.

Buchholz, N. (2018), " Spatial Equilibrium, Search Frictions and Efficient Regulation in the Taxi Industry, " *Working Paper*.

Cho, S., Lee, G., Rust, J., Yu, M. (2018), "Optimal Dynamic Hotel Pricing, ", *Working Paper*

Corgel, J., Lane, J. & Woodworth, M. (2012), "Hotel Industry Demand Curves," *Journal of Hospitality Financial Management*, Volume 20, Issue 1, Article 6.

Crawford, G. & Shum, M. (2005), "Uncertainty and learning in pharmaceutical demand," *Econometrica*, Vol. 73, No. 4, 1137-1173.

Dana Jr., J. & Williams, K. (2018), "Oligopoly Price Discrimination: The Role of Inventory Control, " *Working Paper*, Cowles Foundation Discussion Paper 2136R

Erdem, T. and Keane, M. (1996), "Decision Making under Uncertainty: Capturing Dynamic Brand Choice Processes in Turbulent Consumer Goods Markets," *Marketing Science*, 15, 1-20.

Erdem, T., Keane, M. & Sun, B. (2008), "A Dynamic Model of Brand Choice When Price and Advertising Signal Product Quality," *Marketing Science*, 27 (6), 1111-1125.

Feng, Y. & Gallego, G. (1995), "Optimal Starting Times for End-of-Season Sales and Optimal Stopping Times for Promotional Fares Youyi Feng, Guillermo Gallego, " *Management Science*, Vol. 41(8), 1371-1391.

Fradkin, A. (2017), "Search, Matching, and the Role of Digital Marketplace Design in Enabling Trade: Evidence from Airbnb," *Working Paper*.

Fradkin, A., Grewall, E. & Holtz, D. (2017), "The Determinants of Online Review Informativeness: Evidence from Field Experiments on Airbnb," *Working Paper*

Gallego, G., and van Ryzin, G. (1994), "Optimal Dynamic Pricing of Inventories with Stochastic Demand Over Finite Horizons, " *Management Science*, 40(8), 999-1020.

Hendel, I., Nevo, A. (2013), "Intertemporal Price Discrimination in Storable Goods Markets, " *American Economic Review*, 103(7), 2722-2751.

Hotz, V. J. & Miller, R. A. (1993), "Conditional Choice Probabilities and the Estimation of Dynamic Models," *Review of Economic Studies*, Vol. 60, 497-529.

Huang, Z. (1998), "Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values, " *Data Mining and Knowledge Discovery*, 2, 283-304

Huang, G., Luo, H. & Xia, J. (2018), "Invest in Information or Wing It? A Model of Dynamic Pricing with Seller Learning," *Working Paper*.

Kano, K. (2006), "Menu Costs, Strategic Interactions, and Retail Price Movement, " *Working paper*.

Keane, M. & Wolpin, K. (1994), "The Solution and Estimation of Discrete Choice Dynamic Programming Models by Simulation and Interpolation: Monte Carlo Evidence, " *Review of Economics and Statistics*, Vol.76, No.4, 648-672

Levy, D., Bergen, M., Dutta, S., Venable, R. (1997), "The Magnitude of Menu Costs: Direct Evidence from Large U.S. Supermarket Chains, " *The Quarterly Journal of Economics*, 112(3) 791-825.

Liu, Q. & Ryzin, G. (2008), "On the Choice-Based Linear Programming Model for Network Revenue Management," *Manufacturing & Service Operations Management*, Vol. 10, No. 2, 288-310.

McAfee, P. & te Velde, V. (2008), "Dynamic Pricing with Constant Demand Elasticity, " *Production and Operations Management*, Vol. 17(4), 432-438.

Merlo, A., Ortalo-Magne, F. & Rust, J. (2015), "The Home Selling Problem: Theory and Evidence," *International Economic Review*, Vol. 56, No. 2, 457-484.

Nair, H. (2007), "Intertemporal Price Discrimination with Forward-looking Consumers: Application to the US Market for Console Video-games, " *Quantitative Marketing and Economics*, 5(3), 239-292.

Pashigian, P. (1988), "Demand Uncertainty and Sales: A Study of Fashion and Markdown Pricing, " *American Economic Review*, Vol. 78(5), 936-953.

Pashigian, P. & Bowen, B. (1991), "Why Are Products Sold on Sale?: Explanations of Pricing Regularities, " *The Quarterly Journal of Economics*, Vol. 106(4), 1015-1038.

Petrin, A. and Train, K. (2010). "A control function approach to endogeneity in consumer choice models,". *Journal of Marketing Research*, 47 (1), 3-13.

Rotemberg, J. (1982), "Sticky prices in the United States, " *Journal of Political Economy*, 90(6) 1187-1211.

Rust, J. (1987), "Optimal Replacement of GMC Bus Engines: An Empirical Model of Harold Zurcher," *Econometrica*, Vol. 55, 999-1033.

Slade, M. (1998), "Optimal Pricing with Costly Adjustment: Evidence from Retail-Grocery Prices, " *Review of Economic Studies*, 65(1) 87-107.

Stokey, N. L. (1979), "Intertemporal Price Discrimination, " *The Quarterly Journal of Economics*, 93(3), 355-371.

Su, X. (2007), "Intertemporal Pricing with Strategic Customer Behavior, " *Management Science*, 53(5), 726-741.

Sweeting, A. (2012), "Dynamic Pricing Behavior in Perishable Goods Markets: Evidence from Secondary Markets for Major League Baseball Tickets," *Journal of Political Economy*, Vol. 120, No. 6, 1133-1172.

Sweeting, A. (2015), "A Model of Non-Stationary Dynamic Price Competition with An Application of Platform Design," *Working Paper*, NET Institute, #15-03.

Talluri, K., & van Ryzin, G. (2004), "Revenue Management Under a General Discrete Choice Model of Consumer Behavior, " *Management Science*, 50(1), 15-33.

Talluri, K. T., & van Ryzin, G. (2005), "The Theory and Practice of Revenue Management, " *Springer*, vol. 68.

Weintraub, G., Benkard, L. & Van Roy, B. (2008), "Markov Perfect Industry Dynamic with Many Firms,", *Econometrica*, Vol. 76, No. 6, 1375-1411.

Williams, K.R. (2018), "Dynamic Airline Pricing and Seat Availability,", *Working Paper.*

Zbaracki, M., Ritson, M., Levy, D., Dutta, S., Bergen, M. (2004), "Managerial and Customer Costs of Price Adjustment: Direct Evidence from Industrial Markets. " *The Review of Economics and Statistics*, 86(2) 512-533.

Zervas, G., Proserpio, D. & Bayer, J.W. (2016), "The Rise of the Sharing Economy: Estimating the Impact of Airbnb on the Hotel Industry," *Working Paper.*

Zhang, D. & Adelman, D. (2009), "An Approximate Dynamic Programming Approach to Network Revenue Management with Customer Choice," *Transportation Science*, 43(3), 381-394.

Appendix

A Model Specification of the Illustrative Example

In this simple model, the seller has an inventory in the form of (a,b,c). There is only 1 unit of each product. I also assume that only single product, adjacent products or all 3 products can be sold in each purchase. In total, there are six combinations: (a), (b), (c), (a,b), (b,c) and (a,b,c). In each period, only one customer arrives. The customer is equally likely to be interested in each of the six product combinations. Let x_t be the inventory vector and b^j denote one of the six product combination. $D_t(p_t)$ is the probability that determines whether the customer makes the final purchase decision. Therefore, the revenue maximization problem of the seller can be formulated as:

$$V_t(x_t) = \max_{p_t} \underbrace{\sum_j \pi_t D_t(p_t) \left[|j| * p_t + V_{t-1}(x_t - b^j)\right]}_{\text{has purchase}} + \underbrace{\left(1 - \sum_j \pi_t D_t(p_t)\right) V_{t-1}(x_t)}_{\text{no purchase}}$$

where π_t is the probability determines which product combination the customer is interested and $|j|$ denotes the number of products in product combination j. In this example, I assume $D_t(p_t)$ use the parameter setting in case 2.

B Sample Construction

The current analysis focuses on listings in Manhattan borough in New York City. New York City is Airbnb's largest market in the North America and Manhattan concentrates more than 50% of all the listings in NYC. Therefore this sample should be relatively representative for large cities. In addition, because Airbnb is a highly decentralized market, the idiosyncrasy of the listings significantly increases the computation burden. Restricting the sample to this area, therefore, strikes a balance between reality and implementability.

To further restrict the number of observations, only listings classified as "entire apartment'" are considered. The reason for this restriction is twofold. First, listing type is the most used filter on Airbnb when renters are searching for listings. Therefore, listings with different apartment types may not be considered at the same time. Second, hosts with shared room or private room listings are more likely to block their listings. This behavior might add extra complication to the model. After eliminating the listings that do not fit these criteria, there are 6,910 listings in the sample. A few extra sample restrictions are also imposed. First, I only include listings with 2 bedrooms or fewer, and this filtering reduces the number of listings to 6,311. Moreover, in order to simplify the modeling of inventory transition, I only include listings with minimum stay requirements that are not greater than 2 days in my sample. In those 5,992 listings which satisfy the minimum stay filtering, I keep 5,343 of them whose prices are between $50 and $350.

In principle, the prices set by the hosts can vary along two dimensions. One is the variation across different check-in date, the other one is the variation across time for a given check-in dates. While this paper addresses both of these variation, more emphasis is placed on the second. Therefore, the sample only includes the check-in dates between Oct 4, 2015 and Oct 31, 2015. A longer period would make this analysis more credible, however, the complexity brought by the longer period may outweigh the benefit.

In this paper, I also supplement the analysis with the data from Infogroup that provides estimated value of the apartment and the data on the age of the apartment. The additional information help control the individual specific characteristics. However, the matching between Airbnb data and the Infogroup data is not straightforward. Airbnb does not provide the exact location of a listing. In general, the distance between the true location and the data location ranges from 0-400 feet. I assign each listing to the nearest residential building and use this to match the Infogroup data set.

C Estimation Details of the Pricing Model

One challenge in the pricing model estimation is the number of listing. Since each listing has its own characteristics and cleaning fee, the number of value functions grows with the number of listings. In order to bypass this obstacle, I use clustering and interpolation method.

First, each listing belongs to one of the 28 neighborhoods in Manhattan borough. In each neighborhood, a listing can be characterized by a two-dimensional vector $(X\beta, f)$ where $X\beta$ measures how renters value the attributes X and f is the cleaning fee.

Second, I use k-means method to categorize $(X\beta, f)$ into 3*3=9 different levels. I only calculate the value function at these levels in each neighborhood. The number of the value functions reduces to 28*9=252. Although this is still a large number, parallel computing can handle the problem of this size without difficulty.

The other challenge in the estimation is the dimension of the price vector. Price P_t is a vector with length 7. Even when each element can take 20 values, the total possible combinations are $20^7 \approx 1.2$ billion. In order to reduce the calculate burden but maintain certain flexibility in the price, I assume that hosts only choose weekday average price and weekend average price. In my sample, 76% of the hosts either set one price for the entire week or one price for the weekend and one price for the weekday. With this simplification, the dimension of P_t is reduced to 2. In the estimation, P_t can only take integer number and the total number of grid points in P_t is 25*25=625. Therefore, the total number of states in each cluster is $T * |P_t| * |a_t| = 56 * 625 * 128 = 4,480,000$

D Price Endogeneity

In this section, I estimate the demand using a linear probability model.

$$y_{t,k} = X_k\gamma_1 + Z_{t,k}\gamma_2 + FE_k + FE_t + FE_L + \epsilon_{t,k}$$

where $y_{t,k}$ is a binary variable denoting whether listing k is rented or not at t days prior to the given check-in date. X_k are the time-invariant characteristics of the listing and $Z_{t,k}$ are the number of competitors and the average price in the neighborhood. FE_k, FE_t and FE_L are the individual listing fixed effect, time fixed effect and neighborhood fixed effect. In this model, I can control unobserved individual characteristics by adding fixed effect. In Table 12, I show that, when sufficient number of observed characteristics are controlled for, the price endogeneity problem can, to some extent, be alleviated. Column (3) in Table 12 is the result with individual fixed effect and Column (5) is without fixed effect but with extra controls. The price coefficients in both columns are close to each other, which indicates that the extra variables added to Column (5) can control for the unobserved factors which may affect both the price and the demand. This example serves as an evidence to support that controlling for additional characteristics in the model can alleviate the price endogeneity. One caveat of this example is that the results should not be over-interpreted. The demand model used in this paper is highly non-linear and far more complicated than this linear model, therefore, even though I have controlled for these additional characteristics shown in Column (5), the price could still be correlated with other unobserved factors which also influence the demand.

Table 12: Demand Estimation (linear probability)

	(1)	(2)	(3)	(4)	(5)	(6)
price	-0.0005***	-0.0011***	-0.0021***	-0.0009**	-0.0019***	-0.0025***
	(0.0001)	(0.0002)	(0.0001)	(0.0002)	(0.0003)	(0.0003)
superhost	0.1680***	-0.0964	0.1161*	0.2450***	0.2115***	0.1280**
	(0.0581)	(0.0655)	(0.0656)	(0.0624)	(0.0544)	(0.0524)
bedroom	0.0929	-0.0667	0.1312*	-0.0973	0.2069***	0.2587***
	(0.0573)	(0.0787)	(0.0754)	(0.0704)	(0.0648)	(0.0640)
cancellation	-0.1424***	-0.0892**	-0.0922**	-0.0666*	0.0209	-0.0767***
	(0.0327)	(0.0379)	(0.0362)	(0.0340)	(0.0311)	(0.0291)
rating	0.0434***	0.0356***	0.0222***	-0.0068*	0.0223***	0.0337***
	(0.0021)	(0.0031)	(0.0030)	(0.0039)	(0.0036)	(0.0029)
photo	0.0113***	-0.0017	0.0040***	0.0053**	-0.0039	0.0008
	(0.0023)	(0.0035)	(0.0015)	(0.0022)	(0.0031)	(0.0019)
basic (air+wifi)	0.0598	0.0456	0.2430***	0.1799*	0.2018**	0.1837***
	(0.0622)	(0.0970)	(0.0746)	(0.0982)	(0.0979)	(0.0582)
comfortable (TV or washer)	0.0586	0.0233	0.0761*	0.0661*	0.2048***	0.1692***
	(0.0394)	(0.0572)	(0.0389)	(0.0379)	(0.0592)	(0.0354)
avg. price of competitors	0.0018***	0.0010***	0.0007	0.0015***	0.0006	0.0001
	(0.0004)	(0.0003)	(0.0005)	(0.0003)	(0.0004)	(0.0003)
number of competitors	0.0008***	0.0004***	0.0003**	0.0005***	0.0002**	0.0001**
	(0.0002)	(0.0001)	(0.0001)	(0.0002)	(0.0001)	(0.0000)
instant booking	-	-	-	0.1333**	0.1523**	0.1666**
				(0.0580)	(0.0635)	(0.0715)
number of review	-	-	-	0.0096**	0.0065	0.0073*
				(0.0044)	(0.0041)	(0.0041)
length of review	-	-	-	0.0034***	0.0024***	0.0029***
				(0.0011)	(0.0008)	(0.0009)
months on the platform	-	-	-	0.0013**	0.0006	0.0010***
				(0.0006)	(0.0005)	(0.0002)
property value (in thousand)	-	-	-	0.0003***	0.0003***	0.0002***
				(0.0000)	(0.0000)	(0.0000)
property age	-	-	-	-0.0023***	-0.0035***	-0.0029***
				(0.0002)	(0.0002)	(0.0003)
const.	0.2737***	0.4416**	1.0162***	0.6884	0.7763***	0.6882***
	(0.0524)	(0.1815)	(0.3325)	(0.6634)	(0.1133)	(0.0961)
time fixed effect	Yes	Yes	Yes	Yes	Yes	Yes
individual fixed effect	No	No	Yes	No	No	Yes
location fixed effect	No	Yes	No	No	Yes	No
obs.	1,302,672					

*90% **95% ***99%.

E Solution Method in Detail

In order to solve the pricing equilibrium, one needs to calculate the path of the system states $\mathbf{ss_t}$. From the prospective of the hosts, $\mathbf{ss_t}$ are given when they are making their pricing decisions. However, the perceived $\hat{\mathbf{ss}}_\mathbf{t}$ should be consistent with the actual $\mathbf{ss_t}$. This condition guarantees that each host is setting their price according to the equilibrium.

The solution method starts with a guess of the system states $\{ss_t^{j,(0)}\}$ which can be calculated from the baseline model. With the system states $\{ss_t^{j,(k)}\}$ in the k^{th} iteration, I can calculate the expected value function and the policy function through backward induction. Given the information of the expected value function and the policy function I can simulate the pricing decision and rental decision. For each simulation path, I can calculate one path for the system states. The new system states $\{ss_t^{j,(k+1)}\}$ are updated as the average over all the simulation paths. The details of this solution method is summarized in Algorithm 1.

Algorithm 1 Solution Method (with Friction)

Initialize $ss_t^{j,(0)}$ for all t and j, iteration indicator k=0
2: **for** $t = 1 \to T$ **do**
  Calculate the expected value function $EV_t^{(k+1)}(p_{t+1}, a_{t+1})$ assuming $ss_t^{j,(k)}$ is known
4:   Calculate the value function $V_t^{(k+1)}(p, a_{t-1})$
end for
6: Simulate the initial price $p_0^{(k+1)}$ according to the choice probability in Equation (17)
for $t = T \to 1$ **do**
8:   Simulate price $p_t^{(k+1)}$ according to Equation (15)
  Simulate the purchase decision according to the demand probability Equation (7)
10: **end for**
Update new system state $ss_t^{j,(k+1)}$ for all t and j
12: Iterate through step 2-11 until $|EV_t^{k+1} - EV_t^k| < \epsilon$

If the price adjustment cost is zero, price in last period does not enter into current period value function. Therefore, price in last period is no longer a state variable. The solution method in this case is similar to the baseline model except that the inventory a_t and time t are the only state variables in the problem. Algorithm 2 summarized the detail of the solution method in this case.

Algorithm 2 Solution Method (without Friction)

Initialize $ss_t^{j,(0)}$ for all t and j, iteration indicator k=0
2: **for** $t = 1 \to T$ **do**
  Calculate the expected value function $EV_t^{(k+1)}(a_{t+1})$ assuming $ss_t^{j,(k)}$ is known
4: **end for**
for $t = T \to 1$ **do**
6:   Simulate price $p_t^{(k+1)}$ according to Equation (15) but replace EV_t with Equation (18)
  Simulate the purchase decision according to the demand probability Equation (7)
8: **end for**
Update new system state $ss_t^{j,(k+1)}$ for all t and j
10: Iterate through step 2-11 until $|EV_t^{k+1} - EV_t^k| < \epsilon$

F Detailed Estimation Results of the Demand

In this section, I show the detailed estimation results of the demand model. In Table 13, one can see that listings with higher rating are more likely to be rented. Number of reviews and the length of those reviews positively affect the rental probability. Moreover, renters prefer listings that have higher property value and longer operation history in the market. Renters with shorter stay prefer listings in newer building, but renter with longer stay prefer listings in older building.

Table 13: Demand Estimation (other coefficients)

	one-day	two-day	three-day	four-day	five-day	six-day	seven-day
rating	0.0044	0.1136***	-0.0071*	0.0599***	0.0377***	0.0601***	0.0501***
	(0.0039)	(0.0042)	(0.0041)	(0.0033)	(0.0039)	(0.0036)	(0.0036)
superhost	0.3364***	-0.0999	0.2325*	0.3336***	0.2965***	0.2153**	0.2297*
	(0.1215)	(0.0812)	(0.1291)	(0.0919)	(0.0944)	(0.0978)	(0.1276)
bedroom	0.0921	-0.1281	0.1811**	-0.1511	0.1239*	0.2854***	0.3571***
	(0.1033)	(0.0921)	(0.0862)	(0.1121)	(0.0728)	(0.1057)	(0.0888)
cancellation	-0.1456***	0.0371	-0.0506*	-0.1518**	-0.1430**	-0.1344***	-0.1670***
	(0.0443)	(0.0289)	(0.0297)	(0.0632)	(0.0596)	(0.0411)	(0.0399)
photo	0.0411**	-0.0176	0.0280	0.0357*	-0.0374	0.0391*	0.0420
	(0.0171)	(0.0164)	(0.0200)	(0.0191)	(0.0301)	(0.0211)	(0.0281)
basic (air+wifi)	0.1336*	0.1152*	0.1761***	0.1589*	0.2011**	0.2982***	0.2677***
	(0.0742)	(0.0662)	(0.0566)	(0.0836)	(0.0874)	(0.0886)	(0.0801)
comfortable (TV or washer)	0.0605*	-0.0299	-0.0717*	0.0741*	0.1035**	0.0985**	0.1134***
	(0.0334)	(0.0335)	(0.0377)	(0.0418)	(0.0451)	(0.0489)	(0.0312)
instant booking	0.2113***	0.1996*	0.2569***	0.1938	0.2965***	0.3122***	0.2749***
	(0.0704)	(0.1115)	(0.0917)	(0.1490)	(0.0612)	(0.1011)	(0.0812)
number of review	0.0134**	0.0089*	0.0212***	0.0233***	0.0110	0.0099**	0.0103***
	(0.0056)	(0.0048)	(0.0069)	(0.0052)	(0.0071)	(0.0043)	(0.0034)
length of review	0.0152***	0.0189***	-0.0042	0.0081**	0.0133*	0.0191**	0.0144*
	(0.0043)	(0.0051)	(0.0037)	(0.0032)	(0.0070)	(0.0078)	(0.0076)
months on the platform	0.0651***	0.0443***	0.0566**	0.1129***	-0.0511*	0.0664***	0.0569***
	(0.0112)	(0.0120)	(0.0231)	(0.0333)	(0.0309)	(0.0212)	(0.0131)
property value (in thousand)	0.0035***	0.0031***	0.0044***	0.0059***	0.0053***	0.0058***	0.0057***
	(0.0002)	(0.0002)	(0.0004)	(0.0001)	(0.0006)	(0.0006)	(0.0040)
property age	0.3913**	0.4415***	-0.2145***	-0.5331***	-0.6621*	-0.5233***	-0.7137***
	(0.1610)	(0.1002)	(0.0899)	(0.1876)	(0.3503)	(0.2010)	(0.2209)
const.	-0.7056***	1.3928***	2.2234***	1.1133*	0.5115***	0.8416***	0.3321*
	(0.0884)	(0.1643)	(0.5113)	(0.6445)	(0.0509)	(0.0622)	(0.1684)
location dummies							
obs.				1,302,672			

*90% **95% ***99%.

G Inventory Structure

In this section I show the detail of the inventory inventory vector which is at the core of the multi-day rental analysis. Let m denote the number of days the listing owner groups together. For instance, if a listing owner groups 7 days (a week) together, then $m = 7$. Let n be the number of all possible combinations of stays within this m days period. Since at most one rental type is accepted in each period and each rental consists of consecutive days, a rental cannot include Monday and Wednesday but exclude Tuesday. Let A be an $m \times n$ incident matrix with 0\1 entry. The example below is the incident matrix with $m = 3$.

$$
\begin{array}{r}
 \\
Mon. \\
Tue. \\
Wed.
\end{array}
\begin{array}{c}
\begin{array}{ccccccc} case1 & case2 & case3 & case4 & case5 & case6 & case7 \end{array} \\
\begin{pmatrix}
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 1
\end{pmatrix}
\end{array}
$$

Case 1 shows there is no rental, and Case 2 to Case 4 are one-day rentals. Case 5 and Case 6 are two different types of two-day rentals. The last column, Case 7, is a three-day rental. Let a_t denote an $m \times 1$ availability vector with 0 in the ith entry representing the ith day is occupied. Let A^j be the j^{th} column of the incident matrix A. If a type j rental A^j is realized, the next period inventory vector a_{t-1} can be expressed as

$$a_{t-1} = a_t - (A^j)'$$

If $m = 7$, which is case in the model, there are 29 different cases (28 renal cases + 1 no rental)

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13
Sun.	1	0	0	0	0	0	0	1	0	0	0	0	0
Mon.	0	1	0	0	0	0	0	1	1	0	0	0	0
Tue.	0	0	1	0	0	0	0	0	1	1	0	0	0
Wed.	0	0	0	1	0	0	0	0	0	1	1	0	0
Thu.	0	0	0	0	1	0	0	0	0	0	1	1	0
Fri.	0	0	0	0	0	1	0	0	0	0	0	1	1
Sat.	0	0	0	0	0	0	1	0	0	0	0	0	1

	c14	c15	c16	c17	c18	c19	c20	c21	c22
Sun.	1	0	0	0	0	1	0	0	0
Mon.	1	1	0	0	0	1	1	0	0
Tue.	1	1	1	0	0	1	1	1	0
Wed.	0	1	1	1	0	1	1	1	1
Thu.	0	0	1	1	1	0	1	1	1
Fri.	0	0	0	1	1	0	0	1	1
Sat.	0	0	0	0	1	0	0	0	1

	c23	c24	c25	c26	c27	c28	c29
Sun.	**1**	0	0	**1**	0	**1**	0
Mon.	**1**	**1**	0	**1**	**1**	**1**	0
Tue.	**1**	**1**	**1**	**1**	**1**	**1**	0
Wed.	**1**	**1**	**1**	**1**	**1**	**1**	0
Thu.	**1**	**1**	**1**	**1**	**1**	**1**	0
Fri.	0	**1**	**1**	**1**	**1**	**1**	0
Sat.	0	0	**1**	0	**1**	**1**	0