Coupon Design in Advertising Systems

Weiran Shen,[1] Pingzhong Tang, [2] Xun Wang, [2] Yadong Xu, [2] Xiwang Yang [3]
[1] Renmin University of China
[2] Tsinghua University
[3] ByteDance
shenweiran@ruc.edu.cn, kenshinping@gmail.com, wxhelloworld@outlook.com,
xuyd17@mails.tsinghua.edu.cn,yangxiwang@bytedance.com

Abstract

Online platforms sell advertisements via auctions (e.g., VCG and GSP auction) and revenue maximization is one of the most important tasks for them. Many revenue increment methods are proposed, like reserve pricing, boosting, coupons and so on. The novelty of coupons rests on the fact that coupons are optional for advertisers while the others are compulsory. Recent studies on coupons have limited applications in advertising systems because they only focus on second price auctions and do not consider the combination with other methods. In this work, we study the coupon design problem for revenue maximization in the widely used VCG auction. Firstly, we examine the bidder strategies in the VCG auction with coupons. Secondly, we cast the coupon design problem into a learning framework and propose corresponding algorithms using the properties of VCG auction. Then we further study how to combine coupons with reserve pricing in our framework. Finally, extensive experiments are conducted to demonstrate the effectiveness of our algorithms based on both synthetic data and industrial data.

Introduction

Online advertising has been one of the most important industry on the Internet. In the United States, its revenue has grown by 15.9 percent in 2019 compared to 2018, surpassing 124$ billion. [1] Advertising has become one of the key revenue sources for many Internet companies, such as Google, Facebook, ByteDance. For example, more than 83.8 percent of Google's revenue comes from online advertising. And the market still shows no signs of slowing down. There are various types of online advertising, for example, social media advertising (e.g., Instagram and Facebook), paid search advertising (e.g., Google and Baidu), and native advertising (e.g., BuzzFeed and Tiktok), and so on. These advertisements are usually sold via auction mechanisms. The mechanisms determine which advertisements will be presented and how much the corresponding advertisers need to pay. There are two widely used auctions, i.e., the GSP auction (Edelman, Ostrovsky, and Schwarz 2007) (generalized second price) which is used by Google and Baidu, and the VCG auction (Vickrey 1961; Clarke 1971; Groves et al. 1973) which is used by Facebook and ByteDance. The difference between them is that the VCG auction is truthful, i.e., advertisers submit bids that truthfully reveal their valuations, while the GSP auction is not. However, it has been proved that the GSP auction has a Nash equilibrium whose outcome is equivalent to that of the VCG auction. Therefore, in this paper, we study how to maximize the revenue in the VCG auction.

In marketing, a coupon is a ticket or document that can be redeemed for a financial discount or rebate when purchasing a product. It is effective in promoting sales and attracting new customers in numerous realistic scenarios, and consumers can find coupons everywhere, ranging from the wrappings of food, to webpage banners. Coupons have been widely studied in economics (Bester and Petrakis 1996; Moraga-González and Petrakis 1999; Kang et al. 2006; Board and Skrzypacz 2016). The most common arguments developed by economists to explain the use of coupons are twofold: 1) Coupons allow for price discrimination. 2) Coupons allow for peak-load pricing (Mckenzie 2008). In this work, coupons play the role of price discrimination. The intuition of revenue increment is that low-valued advertisers would spontaneously bid higher if they are provided with coupons, which in turn forces the high-valued advertisers to pay more when they win based on the payment rule of the VCG auction. In fact, many methods can be used to increase revenue, such as reserve price (Hartline and Roughgarden 2009), boosting (Golrezaei et al. 2017), squashing (Lahaie and Pennock 2007a), anchoring (Lahaie and Pennock 2007b). Coupons are significantly different from them. As for coupons, advertisers hold the right to decide whether to use them. However, they cannot refuse to use the other methods, e.g., advertisers always want to remove the reserve price but they can not. In fact, sometimes reserve prices may have a negative effect on revenue (Ostrovsky and Schwarz 2011). Higher reserve prices make the auction less attractive which results in fewer participated advertisers.

In this work, we generalize (Shen et al. 2020b) to the VCG auction, including the combination with reserve pricing. The intuition is that the VCG auction (with heterogeneous slots) can be decomposed into some tractable sub-VCG auctions (with homogeneous slots). Our contributions can be summarized as follows.

[1]https://www.statista.com/statistics/183816/us-online-advertising-revenue-since-2000/

- We generalize the application of coupons for revenue maximization in advertising systems to more realistic scenarios (i.e., VCG auctions).
- We propose algorithms for coupon optimization using the properties of the VCG auction, along with the combination with reserve pricing.
- Based on both synthetic data and industrial data, extensive experiments are conducted to demonstrate the effectiveness of our algorithms.

Related Works

Our work is related to the revenue-maximizing mechanism design. Myerson (1981) studies the optimal auction where advertisers should be ranked in descending order of their virtual values. If nobody has positive virtual values, the item would not be sold to anyone. However, Roughgarden and Schrijvers (2016) argues that this setting relies heavily on the exact estimation of the advertisers' value distribution and can be sensitive to estimation errors. Thus, simpler mechanisms are applied in industry, e.g., VCG and GSP auction. Many methods have been proposed to increase revenue in these auctions. Reserve pricing is one of the most widely studied methods: Hartline and Roughgarden (2009) presents how to use a reserve-price-based VCG auction to approximate the optimal one. Jin et al. (2019) proves a tight approximation ratio for anonymous reserve prices. In (Zeithammer 2019), there is a bid level below which a winning bidder has to pay according to the first-price auction instead of the second-price auction with reserve. Another method is boosting (Golrezaei et al. 2017) where the seller assigns a boost value to each advertiser which can transform his regular bid into a boosted one. Besides, squashing (Lahaie and Pennock 2007a) and anchoring (Lahaie and Pennock 2007b) are also widely used methods in industry (Chawla, Fu, and Karlin 2014; Huang, Mansour, and Roughgarden 2015; Tang and Wang 2016; Tang and Sandholm 2011).

Our work is also related to automated mechanism design. Conitzer and Sandholm (2002) introduces the automated mechanism design approach, where the mechanism is computationally created for the specific problem instance at hand. Learning theory tools such as pseudo-dimension have been used to prove strong guarantees in auction settings (Morgenstern and Roughgarden 2015, 2016). In a similar direction, bounds on the sample complexity have also been developed (Hsu et al. 2016). Mohri and Medina (2014) uses DC programming to learn anonymous reserve price for the second-price auction. Derakhshan, Golrezaei, and Leme (2019); Derakhshan, Pennock, and Slivkins (2020) use linear programming to compute personalized reserve prices in auction mechanism. Duetting et al. (2019) uses deep learning to design revenue-maximizing mechanisms and deals with the incentive constraint via regret network. Shen, Tang, and Zuo (2019) also handles the same problem but can always return exactly incentive compatible mechanisms. Recently, Tang (2017) uses reinforcement learning to design and optimize mechanisms in dynamic industrial environments and has achieved success in industry (Cai et al. 2018a,b; Shen et al. 2020a).

Preliminaries

In this section, the setting of VCG auction with coupons in advertising systems will be introduced first. Then the formal problem of learning to design coupons will be defined.

VCG Auction with Coupons in Advertising Systems

There are n bidders $[n] = \{1, 2, \ldots, n\}$ and m slots (usually, $n \gg m$). We will refer to 'ad i' as the advertisement submitted by bidder i. Each bidder can occupy at most 1 slot and each slot can be allocated to at most 1 bidder. Let v_i be the private value which quantifies the value of a click for bidder i. And v_i is drawn from a publicly known distribution F_i. For convenience, let $\boldsymbol{v} = (v_1, \ldots, v_n)$ denote the value profile, and let $\boldsymbol{v}_{-i} = (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ denote the value profile of all bidders except bidder i. Similarly, we can define the bid profile $\boldsymbol{b}$ and $\boldsymbol{b}_{-i}$. We only use $\alpha_j \in [0, 1]$ to denote the click-through-rate of ad i when it is allocated to slot j as (Edelman, Ostrovsky, and Schwarz 2007) do, which implies that the number of times a particular slot is clicked does not depend on the ad in this slot. The reason is that the click-through-rate can be decomposed into position effect and ad effect while the ad effect can be taken into account in the private valuation. Besides, $\alpha_j \geq \alpha_{j'}$ holds when $j < j'$. In response to the bid profile, VCG auction determines how to allocate each bidder and how much each bidder has to pay according to some allocation rule and payment rule. To be specific, the allocation rule is a function π that maps the bid profile $\boldsymbol{b}$ to an n-dimensional vector indicating the quantity of clicks allocated to each bidder, i.e., $\pi : \mathbb{R}^n \mapsto [0, 1]^n$; The payment rule is a function $p : \mathbb{R}^n \mapsto \mathbb{R}^n_+$ that maps the bid profile $\boldsymbol{b}$ to an n-dimensional non-negative vector specifying the payment for each bidder. For simplicity, we use $\pi(\boldsymbol{b})$ and π, $p(\boldsymbol{b})$ and p interchangeably. If bidder i is allocated to slot j, then $\pi_i = \alpha_j$, and if bidder i does not win the auction, then $\pi_i = 0$. These rules are specified as:

- Allocation rule: $\pi(\boldsymbol{b}) \in \arg\max_{\pi'} \sum_{i=1}^n \pi'_i b_i$.
- Payment rule: $p_i(\boldsymbol{b}) = \max_{\pi'} \sum_{i' \neq i} \pi'_{i'} b_{i'} - \sum_{i' \neq i} \pi_{i'} b_{i'}$.

Coupons work as follows. Before submitting bids, each bidder i will be informed that he will be provided with a constant coupon $c_i \geq 0$. This coupon can get c_i off the payment when he wins. Hence bidders would change their regular bids depending on their values and coupons, i.e., $b_i = b_i(v_i; c_i)$. Therefore, the allocation rule and payment rule of VCG auction with coupons would be:

$$\begin{aligned} \pi(\boldsymbol{b}) &\in \arg\max_{\pi'} \sum_{i=1}^n \pi'_i b_i, \\ p_i(\boldsymbol{b}) &= \max_{\pi'} \sum_{i' \neq i} \pi'_{i'} b_{i'} - \sum_{i' \neq i} \pi_{i'} b_{i'} - \pi_i c_i. \end{aligned} \tag{1}$$

The utility of bidder i is $u_i(\boldsymbol{b}) = \pi_i v_i - p_i$, and the revenue of platform would be:

$$Rev(\boldsymbol{v}, \boldsymbol{c}) = \sum_{i=1}^n p_i. \tag{2}$$

In what follows, let (i) denote the order of bidders satisfying if $i < i'$, then $[b_{(i)} > b_{(i')}] \vee [(b_{(i)} = b_{(i')}) \wedge (c_{(i)} \leq c_{(i')})]$, that is, (i) denotes the bidder with the i-th highest bid (if there are more than one, then choose the bidder with the least coupon).

Learning Formulation

There are two sets of data – the training data and the testing data, containing features and bids. Both sets of data are generated by VCG auction without coupons. And our goal is to train a model which can provide coupons for bidders in each auction and maximize the revenue for platform. Since VCG auction is a truthful mechanism, in these sets of data, we can regard bids as bidders' values.

Let us define the problem formally. We consider a generic feature space $\mathcal{X}$ with the label space $\mathcal{B} = \mathbb{R}_+^n$ consisting of the value profile $\boldsymbol{v}$. We regard $(\boldsymbol{x}^t, \boldsymbol{v}^t)$ as a data instance where $\boldsymbol{x}^t, \boldsymbol{v}^t$ denote the feature vector and value profile in auction t, respectively. The hypothesis function $h : \mathcal{X} \to \mathbb{R}_+^n$ is used to set coupons $\boldsymbol{c}^t = h(\boldsymbol{x}^t)$. Thus, the objective is to select a hypothesis function h out of some hypothesis set H to minimize the corresponding empirical loss:

$$\mathcal{L}_S(h) = -\frac{1}{T}\sum_{t=1}^{T} Rev(\boldsymbol{v}^t; h(\boldsymbol{x}^t)). \tag{3}$$

where $S = ((\boldsymbol{x}^1, \boldsymbol{v}^1), \ldots, (\boldsymbol{x}^T, \boldsymbol{v}^T))$. Let α_j^t, p_i^t and π_i^t denote the corresponding notations in auction t. The goal is to learn the predictor which can work for any advertiser if corresponding features can be provided. For the sake of description, we use a new hypothesis function h_i to set the coupon for bidder i as $c_i^t = h_i(\boldsymbol{x}_i^t)$, where $\boldsymbol{x}_i^t$ denotes the feature for bidder i in auction t. For convenience, in what follows we will omit the superscript t when there is no confusion.

It is worth noting that all proofs can be found in supplementary materials due to limited space.

Problem Analysis

In this section, we will introduce the property of coupons and show the difficulty of coupon optimization first. Then, the comparison with related work would be presented.

Proposition 1. *In VCG auction with coupons, the dominant strategy of bidder i is using coupon c_i and bidding $v_i + c_i$.*

Proposition 1 denotes the dominant strategy of bidders in VCG auction with coupons. Therefore, we can use $v_i + c_i$ to denote the bid of bidder i when he is provided with coupon c_i in experiments. Thus we can define the revenue of VCG auction with coupons as Equation (4).

$$\begin{aligned} Rev(\boldsymbol{v};\boldsymbol{c}) &= \sum_{j=1}^{m}\Big(\sum_{i=j}^{m}\alpha_i b_{(i+1)} - \sum_{i=j+1}^{m}\alpha_i b_{(i)} - \alpha_j c_{(j)}\Big) \\ &= \sum_{j=1}^{m}\Big[j\cdot(\alpha_j - \alpha_{j+1})(v_{(j+1)} + c_{(j+1)}) - \alpha_j c_{(j)}\Big] \end{aligned} \tag{4}$$

Definition 1 (No-feature case). *In this case, features are not considered, and the coupon optimization problem is to minimize the following empirical loss:*

$$\mathcal{L}_S(\boldsymbol{c}) = -\frac{1}{T}\sum_{t=1}^{T} Rev(\boldsymbol{v}^t; \boldsymbol{c}), \tag{5}$$

where $\boldsymbol{c}$ is an $n-$dimensional vector specifying coupons which need to be optimized.

Proposition 2. *In the no-feature case, the optimization of coupons in VCG auction (i.e., Equation* (5)*) is NP-hard.*

Proposition 2 points out coupon optimization is hard, so we use the coordination descent method to optimize coupons. In this method, instead of optimizing coupons simultaneously, we optimize one of them while fixing the others at each time step.

Comparison with Related Work

As proposition 1 demonstrated, although coupons are optional for advertisers, they still accept and use all coupons in equilibrium. We will compare the difference between coupons and other involuntary discount mechanism, like boosting (Golrezaei et al. 2017) and bidding credits in squashing (Lahaie and Pennock 2007a). Boosting and bidding credits both multiply the submitted bids with a factor, i.e., $a_i * b_i$. As for boosting, there is no restriction on a_i. While for bidding credits, a_i is between 0 and 1. Then in the VCG auction, there are two differences.

- When $0 \leq a_i < 1$, advertiser i prefer not to accept boosting or bidding credits. However, as for coupons, advertisers' dominant strategy would always be accepting them.
- Coupons can increase the revenue in some cases where boosting can not. For example, consider one slot and two bidders whose values are 0 and 1. Then, boosting can not increase the revenue since a_i multiplies 0 still equals 0. While for coupons, this would not be a problem.

Comparing with the second-price setting, several non-trivial technical issues are raised in the VCG setting. For example, Shen et al. (2020b) relies heavily on the fact that given value profile $\boldsymbol{v}$ and coupons profile c_{-i}, $Rev(\boldsymbol{v};\boldsymbol{c})$ (as a function of c_i) has at most one discontinuous point. Hence the surrogate loss function can be decomposed as a difference of two convex functions and DC programming can be applied. However, in the VCG setting, the revenue function is more complicated as Proposition 3 denotes.

Proposition 3. *In VCG auction with coupons, given value profile $\boldsymbol{v}$ and the coupon profile c_{-i}, $Rev(\boldsymbol{v};\boldsymbol{c})$ (as a function of c_i) is not continuous, and is neither convex nor concave. Actually, there can be m discontinuous points.*

We use Example 1 to demonstrate Proposition 3. As illustrated in Figure 1, $Rev(\boldsymbol{v};\boldsymbol{c})$ (as a function of c_i) is a piecewise function. There can be multiple discontinuous points. They divide the function into multiple line segments, which can have different slopes.

Example 1. *Let $n = 6$, $m = 4$, $\boldsymbol{v} = (5,4,3,2,1,0)$ and $\boldsymbol{\alpha} = (1.0, 0.4, 0.3, 0.1)$. When $c_{-6} = \boldsymbol{0}$, the value of revenue with respect to c_6 is demonstrated in Figure 1.*

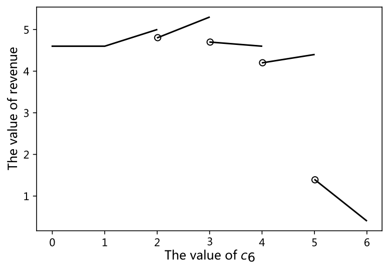


Figure 1: The value of revenue w.r.t. c_6

Furthermore, when reserve price is taken into consideration, the revenue function can be much more complicated, making the coupon optimization harder.

Warm up

In this section, we will introduce a special case of VCG auction with coupons. As Proposition 4 denotes, we can extend the technique in (Mohri and Medina 2014; Shen et al. 2020b) to tackle the special case.

Definition 2 (**Special case**). *In this case of VCG auction with coupons, the slots are homogeneous, i.e.,* $\alpha_1 = \alpha_2 = \cdots = \alpha_m = \alpha$.

Let subscript (j) only represent the order of bids. We use $b_{-i,(j)} = v_{-i,(j)} + c_{-i,(j)}$ to denote the j-th highest bid in bid profile b_{-i}, that is, if $j < l$, then the order $[b_{-i,(j)} > b_{-i,(l)}] \vee [(b_{-i,(j)} = b_{-i,(l)}) \wedge (c_{-i,(j)} \leq c_{-i,(l)})]$ is ensured. Let $L(y; \boldsymbol{v}, c_{-i}) = -Rev(\boldsymbol{v}; (y, c_{-i}))$ be the negative number of the revenue as a function of $c_i = y$, which can be rewritten as:

$$
\begin{aligned}
&L(y; \boldsymbol{v}, c_{-i}) \\
&= \begin{cases}
-m\alpha b_{-i,(m+1)} + \alpha \sum_{j=1}^{m} c_{-i,(j)}, & y \leq b_{-i,(m+1)} - v_i \\
[-m\alpha b_{-i,(m)} + \alpha \sum_{j=1}^{m} c_{-i,(j)} & \\
\ -\alpha(c_{-i,(m)} - y)^+], & y = b_{-i,(m)} - v_i \\
\alpha y - m\alpha b_{-i,(m)} + \alpha \sum_{j=1}^{m-1} c_{-i,(j)}, & y > b_{-i,(m)} - v_i \\
-m\alpha y - m\alpha v_i + \alpha \sum_{j=1}^{m} c_{-i,(j)}, & otherwise
\end{cases}
\end{aligned} \tag{6}
$$

Here x^+ means $\max\{x, 0\}$. As Proposition 4 states, $L(y; \boldsymbol{v}, c_{-i})$ has at most one discontinuous point, which is a reduction of Proposition 3.

Proposition 4. *In the special case, given the value profile* $\boldsymbol{v}$ *and the coupon profile* c_{-i}, $L(y; \boldsymbol{v}, c_{-i})$ *has at most one discontinuous point at* $y = b_{-i,(m)} - v_i$. *When* $y < b_{-i,(m)} - v_i$, $L(y; \boldsymbol{v}, c_{-i})$ *is non-increasing and concave. When* $y > b_{-i,(m)} - v_i$, $L(y; \boldsymbol{v}, c_{-i})$ *is an increasing linear function. What's more, for all the cases,* $L(y; \boldsymbol{v}, c_{-i})$ *achieves its minimum at* $y = (b_{-i,(m)} - v_i)^+$.

However, $L(y; \boldsymbol{v}, c_{-i})$ is still not a good choice for optimization since $L(y; \boldsymbol{v}, c_{-i})$ is neither convex nor concave, but quasi-convex in fact. And a sum of quasi-convex functions does not maintain the quasi-convex property and can have many local minima (Mohri and Medina 2014). We extend the technique in (Mohri and Medina 2014; Shen et al. 2020b) to tackle the special case:

1. We smooth Equation (6) to derive a **surrogate loss** function $L^\gamma(y; \boldsymbol{v}, c_{-i})$, which is training-friendly and has a good approximation of Equation (6) theoretically.
2. We decompose L^γ as the difference of two convex functions, i.e., $L^\gamma = g_1 - g_2$. Then we apply DC programming and coordination descent method to optimize the coupons.

Due to limited space, detailed formulas of L^γ and algorithms are presented in the supplementary materials.

Solution for the General Case

In this section, we will discuss how to extend the aforementioned special case into the general case, i.e., the click-through-rates of different slots are also different. We decompose each VCG auction into m sub-VCG auctions which belong to the special case, and complete the coupon optimization based on the decomposition. Then we study how to combine coupons with reserve prices in our framework.

Constructing m sub-VCG Auctions

Algorithm 1 Constructing m sub-VCG auctions

Initialize $Q = \emptyset$.
for $j = 1$ to m **do**
  Let $d_j = \alpha_j - \alpha_{j+1}$, and construct a sub-VCG auction A_j as: there are n bidders, j slots, and each slot has the same click-through-rates d_j.
  $Q = Q \cup \{A_j\}$.
end for

The decomposition process is demonstrated in Algorithm 1. There are m steps. At the j-th step, we construct the j-th sub-VCG auction, where there are j slots and n bidders, and all slots share the same click-through-rates d_j. Thus, the j-th sub-VCG auction belongs to the special case. Furthermore, Theorem 1 denotes that given value profile $\boldsymbol{v}$ and coupon profile $\boldsymbol{c}$, the results of these m sub-VCG auctions are equivalent to those results of the original VCG auction for both platform (in terms of revenue) and each bidder (in terms of allocation and payment).

Theorem 1. *Given value profile* $\boldsymbol{v}$ *and coupon profile* $\boldsymbol{c}$*, the original VCG auction and these sub-VCG auctions would receive the same bid profile* $\boldsymbol{b}$*. Besides, for bidder* i*, his total payment and allocation (the sum of obtained click-through-rates) within these sub-VCG auctions are equivalent to those in the original VCG auction.*

As a result, we present Theorem 2 to denote that by computing the optimal coupon for bidder i in sub-VCG auctions, the optimal coupon for bidder i in the original VCG auction can also be obtained.

Theorem 2. *Given* $\boldsymbol{v}$ *and* c_{-i}*, the optimal coupon* c_i^* *for bidder* i *in the original VCG auction belongs to the following sets*

$$\{0\} \cup \{(b_{-i,(j)} - v_i)^+\}_{j \in [m]},$$

where $(b_{-i,(j)} - v_i)^+$ is the optimal coupon for bidder i in the j-th sub-VCG auction A_j.

Combine with the coordinate descent method, we propose Algorithm 2 to optimize coupon for the no-feature case. Here λ in line 7 denotes the learning rate, ϵ and K_{out} in line 10 are used to determine when the algorithm terminates.

Algorithm 2 Algorithm for the no-feature case

1: Run Algorithm 1 to construct mT sub-VCG auctions.
2: Initialize $\boldsymbol{c} \leftarrow 0$ and $k_{out} = 0$, and generate a random permutation of 1 to n as $\bar{\mathcal{N}}$.
3: **repeat**
4: Set $\boldsymbol{c}_p \leftarrow \boldsymbol{c}$ and $k_{out} \leftarrow k_{out} + 1$.
5: **for** $i = \bar{\mathcal{N}}_1$ to $\bar{\mathcal{N}}_n$ **do**
6: Fix c_{-i}, calculate the optimal coupon via evaluating Equation (5) at the following $O(mT)$ points and returning the best y^*.

$$\{0\} \cup \{(b^t_{-i,(j)} - v^t_i)^+\}^{t\in[T]}_{j\in[m]}$$

7: Use $\lambda y^* + (1-\lambda)c_i$ to update c_i.
8: **end for**
9: Generate a new random permutation $\bar{\mathcal{N}}$ if $\mathcal{L}_S(\boldsymbol{c}) \geq \mathcal{L}_S(\boldsymbol{c}_p)$.
10: **until** $\|\boldsymbol{c} - \boldsymbol{c}_p\| \leq \epsilon$ or $k_{out} = K_{out}$

Algorithm Design for the General Case

The hypothesis set H consists of linear functions whose unbiased term is bounded, i.e., $H = \{h : \boldsymbol{x}_i \mapsto \boldsymbol{\omega} \cdot \boldsymbol{x}_i + c_0 | \|\boldsymbol{\omega}\| \leq \Delta\}$ and c_0 is a positive constant. Besides, Δ is chosen to satisfy $\Delta \leq \frac{c_0}{\max \|\boldsymbol{x}_i\|}$, which guarantees that coupons are non-negative because

$$\begin{aligned}\boldsymbol{\omega} \cdot \boldsymbol{x}_i + c_0 &\geq c_0 - |\boldsymbol{\omega} \cdot \boldsymbol{x}_i| \geq c_0 - \|\boldsymbol{\omega}\|\|\boldsymbol{x}_i\| \\ &\geq c_0 - \Delta \cdot \max \|\boldsymbol{x}_i\| \geq 0.\end{aligned}$$

It is worth noting that $\max \|\boldsymbol{x}_i\|$ can be bounded once we normalize the feature space.

Note that Theorem 1 states that the results of the m sub-VCG auctions are equivalent to those of the original VCG auction, which holds over the T VCG auctions, thus the empirical surrogate loss can be written as

$$\sum_{t=1}^{T}\sum_{j=1}^{m} L^{\gamma}(\boldsymbol{\omega}_i \cdot \boldsymbol{x}^t_i + c_0; \boldsymbol{v}^t, c^t_{-i}, A^t_j), \tag{7}$$

where A^t_j denotes the j-th constructed sub-VCG auction of the t-th VCG auction by executing Algorithm 1, and the corresponding click-through-rate is $d^t_j = \alpha^t_j - \alpha^t_{j+1}$. It is worth noting that A^t_j belongs to the special case. Follow the work in (Mohri and Medina 2014; Shen et al. 2020b), we also use the technique DC programming for optimization. Therefore, we complete the algorithm design for the general case as Algorithm 3. Here $\text{DCA}(\boldsymbol{\omega}^{k-1}_i)$ (line 8) means to solve the following optimization problem (Equation (8)),

Algorithm 3 Algorithm for the general case

1: Run Algorithm 1 to construct mT sub-VCG auctions.
2: Initialize $\boldsymbol{\omega} \leftarrow \mathbf{0}$ and $k_{out} = 0$, and generate a random permutation of 1 to n as $\bar{\mathcal{N}}$.
3: **repeat**
4: Set $\boldsymbol{\omega}_p \leftarrow \boldsymbol{\omega}$ and $k_{out} \leftarrow k_{out} + 1$.
5: **for** $i = \bar{\mathcal{N}}_1$ to $\bar{\mathcal{N}}_n$ **do**
6: Use $\boldsymbol{\omega}$ to initialize $\boldsymbol{\omega}^0_i$, and calculate coupons for bidders except i via $c^t_j = \boldsymbol{\omega} \cdot \boldsymbol{x}^t_j + c_0$.
7: **for** $k = 1$ to K **do**
8: $\boldsymbol{\omega}^k_i \leftarrow \text{DCA}(\boldsymbol{\omega}^{k-1}_i)$.
9: **end for**
10: Use $\lambda\boldsymbol{\omega}^K_i + (1-\lambda)\boldsymbol{\omega}$ to update $\boldsymbol{\omega}$.
11: **end for**
12: Set $\Psi = \{0.1 \cdot (10\Delta/\|\boldsymbol{\omega}\|)^{0.1j} | j = 0, 1, \ldots, 10\}$, compute $\eta^* \in \arg\min_{\eta\in\Psi} \mathcal{L}_S(\eta \cdot \boldsymbol{\omega})$ and use $\eta^* \cdot \boldsymbol{\omega}$ to update $\boldsymbol{\omega}$.
13: Generate a new random permutation $\bar{\mathcal{N}}$ if $\mathcal{L}_S(\boldsymbol{\omega}) \geq \mathcal{L}_S(\boldsymbol{\omega}_p)$.
14: **until** $\|\boldsymbol{\omega} - \boldsymbol{\omega}_p\| \leq \epsilon$ or $k_{out} = K_{out}$

$$\min_{\|\boldsymbol{\omega}_i\| \leq \Lambda, \boldsymbol{s}} \quad \sum_{t=1}^{T}\sum_{j=1}^{m} s^t_j - \delta G_2(\boldsymbol{\omega}^{k-1}_i) \cdot \boldsymbol{\omega}_i \quad \text{s.t. } \forall j \in [m]$$

$$\begin{cases} c^t_i = \boldsymbol{\omega}_i \cdot \boldsymbol{x}^t_i + c_0, & t \in [T] \\ s^t_j/d^t_j \geq c^t_i - jb^t_{-i,(j)} + \sum_{l=1}^{j-1} c^t_{-i,(l)}, & t \in C^j_1 \cup C^j_3 \cup C^j_4 \\ s^t_j/d^t_j \geq -jc^t_i - jv^t_i + f^{t,j}_{i,1}, & t \in C^j_2 \\ s^t_j/d^t_j \geq (1 + \frac{e^{t,j}_i}{\gamma d^{t,j}_{i,1}})(c^t_i - d^{t,j}_{i,1}) - jb^t_{-i,(j)} + f^{t,j}_{i,1}, & t \in C^j_2 \\ s^t_j/d^t_j \geq (\frac{e^{t,j}_i}{\gamma d^{t,j}_{i,1}} - j)(c^t_i - d^{t,j}_{i,1}) - jb^t_{-i,(j)} + f^{t,j}_{i,2}, & t \in C^j_3 \\ s^t_j/d^t_j \geq \frac{\phi^{t,j}_i}{\alpha^t}(c^t_i - d^{t,j}_{i,1}) - jb^t_{-i,(j)} + f^{t,j}_{i,2}, & t \in C^j_4 \end{cases} \tag{8}$$

where $G_2(\boldsymbol{\omega}^{k-1}_i) = \sum_{t,j} g_2(\boldsymbol{\omega}^{k-1}_i \cdot \boldsymbol{x}^t_i + c_0; \boldsymbol{v}^t, c^t_{-i}, A^t_j)$ and $\delta G_2(\boldsymbol{\omega}^{k-1}_i)$ denotes an arbitrary element of the subgradient $\partial G_2(\boldsymbol{\omega}^{k-1}_i)$. Besides, C^j_k, $d^{t,j}_{i,1}$, $d^{t,j}_{i,2}$, $e^{t,j}_i$, $f^{t,j}_{i,1}$, $f^{t,j}_{i,2}$ and $\phi^{t,j}_i$ are similar to the definitions in the special case with just a substitution from m to j as Equation (9).

$$\begin{aligned} & d^{t,j}_{i,1} = b^t_{-i,(j)} - v^t_i, \ d^{t,j}_{i,2} = b^t_{-i,(j+1)} - v^t_i, \ e^{t,j}_i = v^t_{-i,(j)} - v^t_i, \\ & f^{t,j}_{i,1} = \sum\nolimits_{l=1}^{j} c^t_{-i,(l)}, \ f^{t,j}_{i,2} = \sum\nolimits_{l=1}^{j-1} c^t_{-i,(l)} + d^{t,j}_{i,1}, \\ & \phi^{t,j}_i = d^t_j/\gamma d^{t,j}_{i,1} \cdot (j(b^t_{-i,(j+1)} - b^t_{-i,(j)}) + e^{t,j}_i), \\ & C^j_1 = \{t | v^t_i \geq b^t_{-i,(j)}\}, \ C^j_2 = \{t | v^t_i < b^t_{-i,(j)}, v^t_i \leq v^t_{-i,(j)}\}, \\ & C^j_3 = \{t | v^t_i < b^t_{-i,(j)}, v^t_i > v^t_{-i,(j)}, (1-\gamma)d^{t,j}_{i,1} \geq d^{t,j}_{i,2}\}, \\ & C^j_4 = \{t | v^t_i < b^t_{-i,(j)}, v^t_i > v^t_{-i,(j)}, (1-\gamma)d^{t,j}_{i,1} < d^{t,j}_{i,2}\}. \end{aligned} \tag{9}$$

Combination with Reserve Price

We combine reserve prices with coupons to further improve the revenue by eliminating the negative payments caused by large coupons. Let $r_i + c_i$ denote the eager reserve price for bidder i where r_i is independent of c_i, the eager reserve price works as below.

1. The auction first eliminates the bidder who does not clear his reserve price, then the remaining bidders form the candidate set $S = \{i|b_i \geq r_i + c_i\}$.
2. Bidders in the candidate set S are allocated according to the allocation rule of VCG auction.
3. The payment rule can be modified as follows. If bidder i gets the j-th slot, then he need to pay

$$\sum_{k=j}^{m} (\alpha_k - \alpha_{k+1}) \max\{b_{(k+1)}, r_i + c_i\} - \alpha_j c_i, \quad (10)$$

   where $b_{(k+1)} = 0$ holds for $k = |S|, \dots, m$ if $|S| \leq m$.

It is worth noting that the payment (Equation (10)) must be **non-negative**. Furthermore, we use Proposition 5 to show the dominant strategy of bidders in VCG auction with coupons when we combine reserve prices. Thus we can still use $v_i + c_i$ to denote the bid of bidder i when he is provided with coupon c_i in experiments.

Proposition 5. *In VCG auction with coupons $\{c_i\}$ and reserve prices $\{r_i + c_i\}$, the dominant strategy of bidder i is using coupon c_i and bidding $v_i + c_i$.*

Note that r_i is independent of c_i, we can naturally integrate the reserve prices into the framework of VCG auction with coupons once we optimize the coupon for each bidder. For example, if we simply select $r_i = 0$ for bidder i, then the reserve price for bidder i is c_i. Bidder i will always clear his reserve price, but his payment can never be negative.

Experiment

In this section, we first verify the validity of the design of surrogate loss function by comparing it with the real loss function. Then we examine the impact of hyper-parameters of our algorithms and verify the properties of coupons. Finally, we conduct some experiments to show the effectiveness of our algorithms against state-of-the-art algorithms.

Data Description

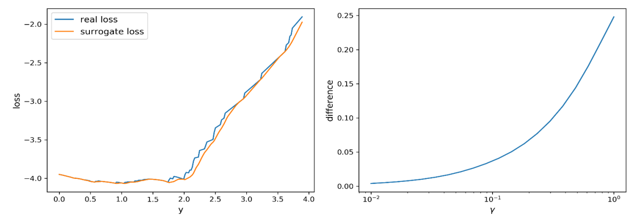
We use both synthetic data and industrial data to demonstrate the results of our experiments.

As for synthetic data, we choose three different types of distribution to sample the value data, i.e., the uniform distribution, the Pareto distribution and the lognormal distribution. To be specific, we sample T numbers from a standard distribution with fixed parameters for each bidder (i.e., $X \sim U(0, 1)$ or $\ln X \sim N(0, 1)$), and then each bidder is associated with a random scale size to multiply these T numbers as his values in T auctions.

As for industrial data, it comes from one of the biggest short-form mobile video community in the world. We randomly extract 177 million auctions from the log. Each advertiser instance contains many features, including labels, industry category, pricing type (i.e., cost per click, cost per mille or cost per action), budget and so on, and the number of features for each bidder in each auction is 178. Besides, all the bids are transformed into the interval $(0, 10)$, and features are normalized to satisfy $\|\boldsymbol{x}_i\| \leq 10$. We choose $c_0 = 10$ so that $\Delta = 1$.

Surrogate Loss vs. Real Loss

We use synthetic data to show the difference between L^γ and L. For a target bidder, we randomly select related $T = 50$ value profiles and generate others' coupons. Without loss of generality, the click-through-rates are all 1. As Figure 2(a) show, $\mathcal{L}_S^\gamma$ is a lower bound for $\mathcal{L}_S$, and the difference between two functions is relatively small. Besides, Figure 2(b) denotes that we can approach the real loss function as we select sufficiently small γ. Similar results are also shown on the industrial data (see in supplementary materials). Although small γ is necessary to guarantee the precision accuracy, too small γ could make the surrogate loss function tend to be discontinuous. γ is chosen to be 0.1 in the experiments.

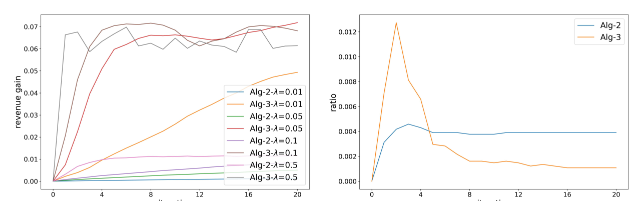


(a) (b)

Figure 2: Plot of the difference between $\mathcal{L}_S$ and $\mathcal{L}_S^\gamma$ on the synthetic data. (a) $\mathcal{L}_S$ and $\mathcal{L}_S^\gamma$ as the function of coupon y when $\gamma = 0.09$, where x-axis denotes the coupon y while y-axis denotes the value of loss. (b) average difference between $\mathcal{L}_S$ and $\mathcal{L}_S^\gamma$ as the function of γ, where x-axis denotes the choice of γ while y-axis denotes the difference.

Demonstration of Training Phase

We implement the algorithms for coupon optimization, where Algorithm 3 is implemented using Gurobi 9.0 (Gurobi Optimization 2020). We first examine the impact of learning rate λ in terms of revenue. The results are shown as Figure 3(a), where Alg-2 and Alg-3 represent Algorithm 2 and 3 respectively and λ is selected from $\{0.01, 0.05, 0.1, 0.5\}$. We can see that Alg-3 always yields better performance than Alg-2 since it can utilize features.



(a) (b)

Figure 3: Plot of training process. (a) revenue curve, where x-axis denotes the iteration while y-axis denotes the corresponding revenue gain. (b) ratio curve, where x-axis denotes the iteration while y-axis denotes the corresponding ratio of negative payments.

Algorithms may not converge within 20 iterations when λ is small, i.e., $\lambda \le 0.1$ in Alg-2 and $\lambda = 0.01$ in Alg-3. As for Alg-2, larger λ can achieve better performance, thus we choose $\lambda = 0.5$ in Alg-2 to guarantee convergence and achieve better performance. While for Alg-3, although larger λ can have higher revenue in some iterations, it is unstable during the training phase. Hence λ is set to 0.05 in Alg-3 to maintain robustness and obtain comparable performance. Besides, in the remaining experiments, we use $\epsilon = 0.001$ and $K_{out} = 20$ in both algorithms.

We further verify the property that the payment can be negative for some bidder when the coupon provided for him is too large. We calculate the ratio of negative payments after each iterations, i.e., $\frac{\#\{(i,t)|p_i^t \le 0\}}{mT}$. As shown in Figure 3(b), the payment of some bidder can be negative, but the ratio of negative payments is small ($< 0.5\%$). Besides, the ratio increases to improve the revenue in the beginning, but then tends to decrease. What's more, the final ratio of negative payments in Alg-3 is smaller than that in Alg-2.

Revenue Comparison

We conduct some experiments to show the effectiveness of coupons and our algorithms. The performance metric is $\rho_a = \frac{Rev_a - Rev_0}{Rev_0} \times 100\%$, where Rev_a represents the revenue achieved through the corresponding algorithm a and Rev_0 denotes the revenue obtained without these methods in VCG auction. The following methods are compared:

- (**ARP**) This method sets anonymous reserve price in VCG auction (Mohri and Medina 2014).
- (**SRP**) This method sets bidder-specific reserve price in VCG auction via maximizing $v_i(1-F_i(v_i))$ (Hartline and Roughgarden 2009).
- (**BVCG**) This method assigns a boost value for each bidder in VCG auction (Golrezaei et al. 2017).
- (**Alg-2a/Alg-3a**) This method combines Algorithm 2 or 3 with reserve prices $r_i + c_i$ where $r_i = 0$.
- (**Alg-2b/Alg-3b**) This method combines Algorithm 2 or 3 with reserve prices $r_i + c_i$ where r_i belongs to $\arg\max_{v_i} v_i(1 - F_i(v_i))$.

Note that (Mohri and Medina 2014; Golrezaei et al. 2017) only study the second price setting, we extend their methods to VCG auction as ARP and BVCG (details are provided in supplementary materials).

As for synthetic data, given the type of distribution and the number of slots m, we randomly sample the value data and divide the whole dataset into training data and testing data, with training data makes up about 70%. Then we run different algorithms on the training data and calculate ρ_a on the testing data. This procedure is repeated for 20 times and the results when $m = 4$ are summarized as Table 1 shows.

We can draw the following conclusions from Table 1. Firstly, coupons work well on these synthetic datasets as Alg-2 can increase the revenue by a large margin. Secondly, Alg-2 always outperforms ARP, SRP and BVCG on the datasets sampled from uniform distribution. This is because coupons can approximate the optimal auction in (Myerson 1981) via producing similar allocation (see in supplementary materials). For other distributions, Alg-2 always has better performance than BVCG. Thirdly, results show the effectiveness of combining reserve prices with coupons. Alg-2(a) outperforms Alg-2 by eliminating the negative payments and Alg-2(b) performs the best among above methods in all datasets by setting bidder-specific reserve prices. Similar conclusions can also be seen in the results when $m \ne 4$, we demonstrate them in the supplementary materials.

Table 1: Performance (i.e. ρ_a) on synthetic data when $m = 4$. Numbers in the brackets denote standard deviations.

Method	Uniform	Pareto	Lognormal
ARP	2.58(1.05)	8.88(2.38)	2.92(3.24)
SRP	12.16(5.98)	9.80(3.36)	21.86(9.44)
BVCG	7.51(3.32)	1.83(1.27)	6.94(2.80)
Alg-2	13.91(5.77)	3.18(2.57)	10.36(4.77)
Alg-2a	13.94(5.75)	3.28(2.57)	10.41(4.75)
Alg-2b	**15.75**(6.07)	**9.94**(3.36)	**22.54**(9.15)

As for industrial data, We first divide 177 million into 59 groups. Then in each group, auctions are divided into training data and testing data. After training, we summarize the average results on these 59 groups as Table 2. On the industrial data, Alg-3(a) outperforms others by a large margin, demonstrating the effectiveness of our mechanisms. More details can be found in supplementary materials.

Table 2: Performance (i.e. ρ_a) on the industrial data.

Method	training data	testing data
ARP	0.00(0.00)	0.00(0.00)
SRP	−1.05(1.13)	−2.86(4.01)
BVCG	1.19(0.36)	0.13(0.75)
Alg-2	1.33(0.57)	0.32(0.29)
Alg-2a		0.36(0.28)
Alg-3	**4.74**(3.95)	4.86(3.81)
Alg-3a		**4.87**(3.81)

Conclusion

In this paper, we study how to design coupons in VCG auction via a learning framework. Firstly, we derive the dominant strategy of each bidder and characterize the coupons. Then we start from a special case where slots are homogeneous and extend this special case to the general one via auction decomposition and complete the algorithm design. We also combine coupons with reserve prices in our framework. Finally, extensive experiments are conducted to demonstrate the effectiveness of the algorithms.

Acknowledgments

This work is is supported by Beijing Academy of Artificial Intelligence (BAAI), Science and Technology Innovation 2030—"New Generation Artificial Intelligence" Major Project No. 2018AAA0100904, 2018AAA0101103.

References

Bester, H.; and Petrakis, E. 1996. Coupons and oligopolistic price discrimination. *International Journal of Industrial Organization* 14(2): 227–242.

Board, S.; and Skrzypacz, A. 2016. Revenue management with forward-looking buyers. *Journal of Political Economy* 124(4): 1046–1087.

Cai, Q.; Filos-Ratsikas, A.; Tang, P.; and Zhang, Y. 2018a. Reinforcement Mechanism Design for e-commerce. In Champin, P.; Gandon, F. L.; Lalmas, M.; and Ipeirotis, P. G., eds., *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, 1339–1348. ACM. doi:10.1145/3178876.3186039. URL https://doi.org/10.1145/3178876.3186039.

Cai, Q.; Filos-Ratsikas, A.; Tang, P.; and Zhang, Y. 2018b. Reinforcement Mechanism Design for Fraudulent Behaviour in e-Commerce. In McIlraith, S. A.; and Weinberger, K. Q., eds., *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, 957–964. AAAI Press. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16650.

Chawla, S.; Fu, H.; and Karlin, A. R. 2014. Approximate revenue maximization in interdependent value settings. In Babaioff, M.; Conitzer, V.; and Easley, D. A., eds., *ACM Conference on Economics and Computation, EC '14, Stanford , CA, USA, June 8-12, 2014*, 277–294. ACM. doi:10.1145/2600057.2602858. URL https://doi.org/10.1145/2600057.2602858.

Clarke, E. H. 1971. Multipart pricing of public goods. *Public choice* 17–33.

Conitzer, V.; and Sandholm, T. 2002. Complexity of Mechanism Design. In Darwiche, A.; and Friedman, N., eds., *UAI '02, Proceedings of the 18th Conference in Uncertainty in Artificial Intelligence, University of Alberta, Edmonton, Alberta, Canada, August 1-4, 2002*, 103–110. Morgan Kaufmann. URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1\&smnu=2\&article_id=850\&proceeding_id=18.

Derakhshan, M.; Golrezaei, N.; and Leme, R. P. 2019. LP-based Approximation for Personalized Reserve Prices. In Karlin, A.; Immorlica, N.; and Johari, R., eds., *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019*, 589. ACM. doi:10.1145/3328526.3329594. URL https://doi.org/10.1145/3328526.3329594.

Derakhshan, M.; Pennock, D. M.; and Slivkins, A. 2020. Beating Greedy For Approximating Reserve Prices in Multi-Unit VCG Auctions. *CoRR* abs/2007.12653. URL https://arxiv.org/abs/2007.12653.

Duetting, P.; Feng, Z.; Narasimhan, H.; Parkes, D. C.; and Ravindranath, S. S. 2019. Optimal Auctions through Deep Learning. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, 1706–1715. PMLR. URL http://proceedings.mlr.press/v97/duetting19a.html.

Edelman, B.; Ostrovsky, M.; and Schwarz, M. 2007. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American economic review* 97(1): 242–259.

Golrezaei, N.; Lin, M.; Mirrokni, V.; and Nazerzadeh, H. 2017. Boosted Second Price Auctions: Revenue Optimization for Heterogeneous Bidders. *Available at SSRN 3016465* .

Groves, T.; et al. 1973. Incentives in teams. *Econometrica* 41(4): 617–631.

Gurobi Optimization, L. 2020. Gurobi Optimizer Reference Manual. URL http://www.gurobi.com.

Hartline, J. D.; and Roughgarden, T. 2009. Simple versus optimal mechanisms. In Chuang, J.; Fortnow, L.; and Pu, P., eds., *Proceedings 10th ACM Conference on Electronic Commerce (EC-2009), Stanford, California, USA, July 6–10, 2009*, 225–234. ACM. doi:10.1145/1566374.1566407. URL https://doi.org/10.1145/1566374.1566407.

Hsu, J.; Morgenstern, J.; Rogers, R. M.; Roth, A.; and Vohra, R. 2016. Do prices coordinate markets? In Wichs, D.; and Mansour, Y., eds., *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, 440–453. ACM. doi:10.1145/2897518.2897559. URL https://doi.org/10.1145/2897518.2897559.

Huang, Z.; Mansour, Y.; and Roughgarden, T. 2015. Making the Most of Your Samples. In Roughgarden, T.; Feldman, M.; and Schwarz, M., eds., *Proceedings of the Sixteenth ACM Conference on Economics and Computation, EC '15, Portland, OR, USA, June 15-19, 2015*, 45–60. ACM. doi:10.1145/2764468.2764475. URL https://doi.org/10.1145/2764468.2764475.

Jin, Y.; Lu, P.; Qi, Q.; Tang, Z. G.; and Xiao, T. 2019. Tight approximation ratio of anonymous pricing. In Charikar, M.; and Cohen, E., eds., *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, 674–685. ACM. doi:10.1145/3313276.3316331. URL https://doi.org/10.1145/3313276.3316331.

Kang, H.; Hahn, M.; Fortin, D. R.; Hyun, Y. J.; and Eom, Y. 2006. Effects of perceived behavioral control on the consumer usage intention of e-coupons. *Psychology & Marketing* 23(10): 841–864.

Lahaie, S.; and Pennock, D. M. 2007a. Revenue analysis of a family of ranking rules for keyword auctions. In MacKie-Mason, J. K.; Parkes, D. C.; and Resnick, P., eds., *Proceedings 8th ACM Conference on Electronic Commerce (EC-2007), San Diego, California, USA, June 11-15, 2007*, 50–56. ACM. doi:10.1145/1250910.1250918. URL https://doi.org/10.1145/1250910.1250918.

Lahaie, S.; and Pennock, D. M. 2007b. Revenue analysis of a family of ranking rules for keyword auctions. In MacKie-Mason, J. K.; Parkes, D. C.; and Resnick, P., eds., *Proceedings 8th ACM Conference on Electronic Commerce (EC-2007), San Diego, California, USA, June 11-15, 2007*, 50–56. ACM. doi:10.1145/1250910.1250918. URL https://doi.org/10.1145/1250910.1250918.

Mckenzie, R. D. 2008. *Why Popcorn Costs So Much at the Movies: And Other Pricing Puzzles*. Springer New York.

Mohri, M.; and Medina, A. M. 2014. Learning Theory and Algorithms for revenue optimization in second price auctions with reserve. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, 262–270. JMLR.org. URL http://proceedings.mlr.press/v32/mohri14.html.

Moraga-González, J. L.; and Petrakis, E. 1999. Coupon advertising under imperfect price information. *Journal of Economics & Management Strategy* 8(4): 523–544.

Morgenstern, J.; and Roughgarden, T. 2015. On the Pseudo-Dimension of Nearly Optimal Auctions. In Cortes, C.; Lawrence, N. D.; Lee, D. D.; Sugiyama, M.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, 136–144. URL http://papers.nips.cc/paper/5766-on-the-pseudo-dimension-of-nearly-optimal-auctions.

Morgenstern, J.; and Roughgarden, T. 2016. Learning Simple Auctions. In Feldman, V.; Rakhlin, A.; and Shamir, O., eds., *Proceedings of the 29th Conference on Learning Theory, COLT 2016, New York, USA, June 23-26, 2016*, volume 49 of *JMLR Workshop and Conference Proceedings*, 1298–1318. JMLR.org. URL http://proceedings.mlr.press/v49/morgenstern16.html.

Myerson, R. B. 1981. Optimal Auction Design. *Math. Oper. Res.* 6(1): 58–73. doi:10.1287/moor.6.1.58. URL https://doi.org/10.1287/moor.6.1.58.

Ostrovsky, M.; and Schwarz, M. 2011. Reserve prices in internet advertising auctions: a field experiment. In Shoham, Y.; Chen, Y.; and Roughgarden, T., eds., *Proceedings 12th ACM Conference on Electronic Commerce (EC-2011), San Jose, CA, USA, June 5-9, 2011*, 59–60. ACM. doi:10.1145/1993574.1993585. URL https://doi.org/10.1145/1993574.1993585.

Roughgarden, T.; and Schrijvers, O. 2016. Ironing in the Dark. In Conitzer, V.; Bergemann, D.; and Chen, Y., eds., *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016*, 1–18. ACM. doi:10.1145/2940716.2940723. URL https://doi.org/10.1145/2940716.2940723.

Shen, W.; Peng, B.; Liu, H.; Zhang, M.; Qian, R.; Hong, Y.; Guo, Z.; Ding, Z.; Lu, P.; and Tang, P. 2020a. Reinforcement Mechanism Design: With Applications to Dynamic Pricing in Sponsored Search Auctions. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 2236–2243. AAAI Press. URL https://aaai.org/ojs/index.php/AAAI/article/view/5600.

Shen, W.; Tang, P.; Wang, X.; Xu, Y.; and Yang, X. 2020b. Learning to Design Coupons in Online Advertising Markets. In Seghrouchni, A. E. F.; Sukthankar, G.; An, B.; and Yorke-Smith, N., eds., *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020*, 1242–1250. International Foundation for Autonomous Agents and Multiagent Systems. URL https://dl.acm.org/doi/abs/10.5555/3398761.3398905.

Shen, W.; Tang, P.; and Zuo, S. 2019. Automated Mechanism Design via Neural Networks. In Elkind, E.; Veloso, M.; Agmon, N.; and Taylor, M. E., eds., *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, QC, Canada, May 13-17, 2019*, 215–223. International Foundation for Autonomous Agents and Multiagent Systems. URL http://dl.acm.org/citation.cfm?id=3331696.

Tang, P. 2017. Reinforcement mechanism design. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 5146–5150. ijcai.org. doi:10.24963/ijcai.2017/739. URL https://doi.org/10.24963/ijcai.2017/739.

Tang, P.; and Sandholm, T. 2011. Approximating Optimal Combinatorial Auctions for Complements Using Restricted Welfare Maximization. In Walsh, T., ed., *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, 379–385. IJCAI/AAAI. doi:10.5591/978-1-57735-516-8/IJCAI11-072. URL https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-072.

Tang, P.; and Wang, Z. 2016. Optimal Auctions for Negatively Correlated Items. In Conitzer, V.; Bergemann, D.; and Chen, Y., eds., *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016*, 103–120. ACM. doi:10.1145/2940716.2940766. URL https://doi.org/10.1145/2940716.2940766.

Vickrey, W. 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance* 16(1): 8–37.

Zeithammer, R. 2019. Soft Floors in Auctions. *Manag. Sci.* 65(9): 4204–4221. doi:10.1287/mnsc.2018.3164. URL https://doi.org/10.1287/mnsc.2018.3164.