

# Pacing Equilibria in Second-Price Auctions with Few Buyers

Yonglei Yan<sup>1</sup>, Zihe Wang<sup>2</sup>, Zhengyang Liu<sup>1\*</sup>

<sup>1</sup>Beijing Institute of Technology

<sup>2</sup>Renmin University of China

yanyonglei@bit.edu.cn, wang.zihe@ruc.edu.cn, zhengyang@bit.edu.cn

## Abstract

We present a polynomial-time algorithm for exactly computing second-price pacing equilibria (SPPE) in auction markets with a constant number of buyers. SPPE plays a central role in modern advertising auctions; however, computing or even approximating it is PPAD-hard in general. To overcome this computational barrier in the restricted setting, we adopt the *cell-decomposition* method. Specifically, we partition the solution space into polynomially many cells, each defined by hyperplanes corresponding to a fixed ordering of buyers’ scaled valuations across goods. Within each cell, the equilibrium computation reduces to solving a constant number of linear programs. Notably, our algorithm can also efficiently identify equilibria that optimize key objectives such as revenue or social welfare. To the best of our knowledge, this is the first algorithm that efficiently computes an *exact* SPPE for a simple and natural class of second-price pacing games.

## Introduction

Online advertising markets face a foundational challenge in auction design under budget constraints. As demonstrated by Aggarwal et al. (2024), advertisers must participate in sequences of second-price auctions while adhering to fixed overall budgets. Traditional second-price auction theory, reliant on quasi-linear utilities and unlimited budgets, fails to capture these real-world constraints. To bridge this gap, pacing mechanisms have become central to modern ad platforms (e.g., ad exchanges), where multiplicative pacing multipliers  $\alpha_i \in [0, 1]$  scale bids to respect budget limits. This adjustment (with the yield of effective offers  $\alpha_i v_{ij}$ ) ensures that long-term expenditure remains within budgets  $B_i$ . It gives rise to the concept of Second-Price Pacing Equilibria (SPPE), a stable state where buyers’ multiplier choices, platform allocations, and pricing rules converge.

The seminal work of Conitzer et al. (2022) formalized SPPE, proving existence under general conditions and characterizing key properties, including potential non-uniqueness and efficiency losses relative to social optima. This framework has since underpinned diverse learning algorithms for revenue and utility optimization (Balseiro and Gur 2019; Wang et al. 2023; Balseiro et al. 2024; Lucier

et al. 2024). Despite these advances, the computational complexity of finding SPPE has remained unresolved. Recent breakthroughs establish its intrinsic hardness: Chen, Kroer, and Kumar (2023) proved PPAD-completeness for computing approximate pacing equilibria with inverse-polynomial precision. Very recently, Chen and Li (2025) established strong inapproximability by showing that computing  $\gamma$ -approximate pacing equilibria is PPAD-hard for any  $\gamma < 1/3$ . This impossibility result holds even in highly sparse settings where each bidder values only four goods, mirroring a similar phenomenon known for Nash equilibria (Chen, Deng, and Teng 2006; Liu and Sheng 2018; Liu, Li, and Deng 2021).

These negative results imply that efficient exact computation — or even high-precision approximation — of SPPE is likely intractable in large markets with many buyers and goods, at least in the worst case. This leads to an interesting open question: *Under what practical constraints can SPPE be computed efficiently?*

We give an affirmative answer to this question by focusing on markets with a constant number of buyers but an arbitrarily large number of goods.

**Theorem 1.** *Given any instance of second-price pacing game with a constant number  $c$  of buyers, Algorithm 1 computes a second-price pacing equilibrium in polynomial time.*

Such settings capture real-world scenarios, such as auctions involving a few major advertising agencies or niche markets dominated by a small number of key players.

## Technical Overview

A pacing equilibrium consists of both allocations and pacing multipliers. Solving for it via programming-based methods requires introducing variables to capture both components. However, a single, universal formulation is unlikely to suffice due to the discrete structure imposed by the “no unnecessary pacing” condition (condition (d) in Definition 1). To address this challenge, we adopt the cell-decomposition method. This approach has been successfully applied in related domains, such as computing market equilibria (Devanur and Kannan 2008) and solving optimal multidimensional pricing problems (Chen et al. 2018).

Concretely, given a market with  $m$  goods and  $c$  buyers, we partition the pacing multiplier space  $(\alpha_1, \dots, \alpha_c) \in (0, 1]^c$

\*Zhengyang Liu is the corresponding author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

via hyperplane arrangements. We establish that  $O(mc^2)$  hyperplanes, induced by comparing scaled valuations  $\alpha_i v_{ij}$  across buyers and goods, divide the space into  $O(m^{c-1})$  cells where the weak ordering of buyers' scaled valuations remains invariant for all goods. Within each cell, we enumerate all  $2^c$  budget-activity cases that distinguish between the constraints  $0 < \alpha_i < 1$  and  $\alpha_i = 1$  for each buyer  $i \in [c]$ . Given each case, we reformulate equilibrium conditions as a linear programming through two critical steps. First, we introduce auxiliary variables  $y_{ij} = x_{ij} \alpha_i$  to linearize the payment terms  $x_{ij} p_j(\alpha)$  that arises when multiple buyers tie. Second, for each set  $I \subseteq [c]$  of buyers who share some good  $j$ , we derive linear constraints of the form  $\sum_{i \in I} y_{ij} v_{ij} = \alpha_{i'} v_{i'j}$ , where  $i'$  is any buyer in the set  $I$ , ensuring consistency of valuations among tied buyers. Budget exhaustion constraints are simultaneously preserved via transformed equalities. The resulting system of linear inequalities can then be efficiently solved using linear programming.

The algorithm runs in polynomial time due to two key properties: there are  $O(m^{c-1})$  cells, and each cell admits  $O(1)$  budget-activity cases. For each case, the corresponding linear system can be solved in  $\text{poly}(m)$  time. As a result, the overall complexity is  $O(\text{poly}(m))$ . We thus obtain a polynomial-time algorithm for computing a pacing equilibrium in budget-constrained second-price auctions with a constant number of buyers.

## Related Works

Recent research by Aggarwal et al. (2024) highlighted that advertisers in online advertising markets must participate in a sequence of second-price auctions under a fixed total budget. The seminal work of Conitzer et al. (2022) formally introduced the concept of Second-Price Pacing Equilibria and established its existence under general conditions. Subsequent studies, including those by Balseiro and Gur (2019); Wang et al. (2023); Balseiro et al. (2024); Lucier et al. (2024), have shown that SPPE has been extensively utilized in the development of various learning algorithms.

For the complexity results, Chen, Kroer, and Kumar (2023) further proved that computing approximate pacing equilibria to inverse-polynomial precision is PPAD-complete. Very recently, Chen and Li (2025) established a strong inapproximability result for pacing equilibria in second-price auctions, showing that for any constant  $\gamma < 1/3$ , computing a  $\gamma$ -approximate pacing equilibrium is PPAD-hard. The above results suggest that efficient algorithms for general instances are unlikely to exist. To the best of our knowledge, this work is the first positive result for pacing equilibrium computation in second-price auctions. In contrast, Borgs et al. (2007); Chen, Kroer, and Kumar (2021) proposed efficient computation methods for pacing and throttling equilibria in the regime of *first-price* auctions.

## Preliminaries

In this section, we will introduce the necessary definitions and notation.

## Second-Price Pacing Games

A *second-price pacing game*  $G = (n, m, (v_{ij}), (B_i))$  involves  $n$  buyers and  $m$  indivisible goods. Each buyer has a specific value to each good, with  $v_{ij}$  denoting the value that buyer  $i$  places on good  $j$ , and  $B_i$  representing the budget of buyer  $i$ . In the game, each buyer  $i$  chooses a *pacing multiplier*  $\alpha_i$ , and submits a bid of  $\alpha_i v_{ij}$  for each good  $j$ .

Each good is sold via a second-price auction. We denote by  $h_j(\alpha) = \max_i \alpha_i v_{ij}$  the highest bid for good  $j$ , and  $p_j(\alpha)$  the second-highest bid. The good is given to the highest-bidding buyers, who pay the second-highest bid  $p_j(\alpha)$ . In case of a tie, i.e., when multiple buyers submit the highest bid, the good is allocated fractionally: each highest bidder receives a non-zero share of good  $j$  at the unit price  $p_j(\alpha)$ . For indivisible goods, such fractional allocation can be interpreted as the probability that a buyer wins the good. We also denote by  $x_{ij}$  the probability that buyer  $i$  gets good  $j$ .

We characterize the stable state of this game with the following solution concept.

**Definition 1 (Pacing Equilibrium).** We say  $(\alpha, x)$  with  $\alpha = (\alpha_i) \in [0, 1]^n$ ,  $x = (x_{ij}) \in [0, 1]^{nm}$ , and  $\sum_{i \in [n]} x_{ij} \leq 1, \forall j \in [m]$  is a *pacing equilibrium* of a second-price pacing game  $G = (n, m, (v_{ij}), (B_i))$  if

- (a)  $x_{ij} > 0$  implies  $\alpha_i v_{ij} = h_j(\alpha)$ ;
- (b)  $h_j(\alpha) > 0$  implies  $\sum_{i \in [n]} x_{ij} = 1$ ;
- (c)  $\sum_{j \in [m]} x_{ij} p_j(\alpha) \leq B_i$ ;
- (d)  $\sum_{j \in [m]} x_{ij} p_j(\alpha) < B_i$  implies  $\alpha_i = 1$ .

For the conditions, (a) indicates that only the highest-bidding buyers can own the good; (b) ensures that any good with a strictly positive bid needs to be entirely allocated; (c) enforces the feasibility of budget constraints; and (d) imposes that no buyer is subject to excessive or unnecessary pacing. Buyers with surplus budget will bid truthfully, as this is a dominant strategy in the second-price auction.

In their work, Conitzer et al. (2022) demonstrated that a pacing equilibrium always exists in the context of second-price pacing games.

**Theorem 2** (Conitzer et al. (2022)). *Any second-price pacing game admits a pacing equilibrium.*

## Notation

For each good  $j \in [m]$ , we need to compare the values  $(\alpha_1 v_{1j}, \alpha_2 v_{2j}, \dots, \alpha_n v_{nj})$ . We define a partition of the good set according to the induced weak orderings of these values.

Let  $\Sigma_n$  denote the set of all weak orderings over the set  $[n]$ , and for each  $\sigma \in \Sigma_n$ , define:

$$S_\sigma := \{j \in [m] : (\alpha_1 v_{1j}, \dots, \alpha_n v_{nj}) \text{ induces order } \sigma\}.$$

Then the ordered tuple of sets  $(S_\sigma)_{\sigma \in \Sigma_n}$  forms a partition of the good set. For example, when  $n = 3$ ,  $\Sigma_3$  contains 13 distinct weak orderings such as  $(1 > 2 > 3), (1 = 2 > 3), \dots, (1 = 2 = 3)$ , corresponding to the following sets, respectively:  $S_{1>2>3} = \{j \in [m] : \alpha_1 v_{1j} > \alpha_2 v_{2j} >$

$\alpha_3 v_{3j}\}, \dots, S_{1=2>3} = \{j \in [m] : \alpha_1 v_{1j} = \alpha_2 v_{2j} > \alpha_3 v_{3j}\}$ .

The ordered tuple  $(S_\sigma)$  consists of all subsets  $S_\sigma$  corresponding to preference orderings over the buyers. For example, when  $n = 3$ , we have  $(S_\sigma) = (S_{1>2>3}, S_{1=2>3}, \dots, S_{1=2=3})$ .

The number of such orderings depends only on the number of buyers. When  $n = c$ , this number remains constant. One can obtain all possible orderings by partitioning the buyers into groups—using  $=$  within each group and  $>$  between groups

We further group the good partitions based on which buyers achieve the maximum scaled valuation for each good. For every non-empty subset  $I \subseteq [n]$ , define:

$$T_I := \{j \in [m] : \operatorname{argmax}_i (\alpha_i v_{ij}) = I\},$$

that is

$$T_I = \bigcup_{\sigma \in \Sigma_n, \operatorname{argmax}(\sigma) = I} S_\sigma.$$

Given an instance  $G = (n, m, (v_{ij}), (B_i))$ , the ordered tuple of sets  $(T_I)_{\emptyset \neq I \subseteq [n]}$  forms a partitioning of the good set  $[m]$  into  $2^n - 1$  parts, each corresponding to a distinct (and disjoint) subset of buyers simultaneously achieving the maximum bid on good  $j$ . For example, when  $n = 3$ , by definition we have that  $T_{\{1\}} = S_{1>2>3} \cup S_{1>2=3} \cup S_{1>3>2}$  and  $T_{\{1,2\}} = S_{1=2>3}$ .

Note that when  $\alpha_i = 0$ , buyer  $i$  effectively places a bid of zero on all goods, which is equivalent to not participating in the auction. In this case, the setting reduces to an auction with  $n - 1$  rather than  $n$  buyers. Without loss of generality, we assume that all  $\alpha_i$ 's are strictly positive.

For ease of exposition in the subsequent analysis, we introduce the following notation: We denote by  $r_{ij} := \alpha_i / \alpha_j$  the ratio between  $\alpha_i$  and  $\alpha_j$ , and  $y_{ij} := x_{ij} \alpha_i$  the product of  $x_{ij}$  and  $\alpha_i$ .

It is also worth noting that, in a pacing equilibrium, assigning a good with a highest bid of zero still satisfies the definition of equilibrium. This is because the highest bid for such a good is zero, implying that its associated price  $p_j(\alpha)$  is also zero. As a result, allocating this good does not affect any budget constraints. Therefore, for ease of exposition, we assume that  $h_j(\alpha)$  is strictly positive for every good.

### Warm-up: when $n = 2$

To facilitate a clearer exposition, we first restrict our attention to the case with only two buyers. We partition the set of goods into the following three parts according to  $\alpha \in (0, 1]^2$ :

$$\begin{aligned} T_{\{1\}} &= S_{1>2} = \{j : \alpha_1 v_{1j} > \alpha_2 v_{2j}\}, \\ T_{\{2\}} &= S_{1<2} = \{j : \alpha_1 v_{1j} < \alpha_2 v_{2j}\}, \\ T_{\{1,2\}} &= S_{1=2} = \{j : \alpha_1 v_{1j} = \alpha_2 v_{2j}\}. \end{aligned}$$

While the total number of possible ordered tuples  $(S_\sigma)$  may appear to be exponential in the number of goods  $m$ , it can in fact be bounded by a linear function of  $m$ .

**Lemma 3.** *When  $n = 2$  and all the values  $v_{ij}$ 's are given, the number of ordered tuples  $(S_\sigma)$ , due to  $\alpha \in (0, 1]^2$ , is bounded by  $O(m)$ .*

*Proof.* To determine the ordered tuple  $(S_\sigma)$ , it is necessary to compare  $\alpha_1 v_{1j}$  and  $\alpha_2 v_{2j}$  for each good  $j \in [m]$ . This can be reduced to compare the ratio  $r_{12} = \alpha_1 / \alpha_2$  with  $v_{2j} / v_{1j}$ . Since all values of  $v_{2j} / v_{1j}$  are given as part of the input, the problem becomes identifying, for a given  $r_{12} \in (0, \infty)$ , which goods satisfy  $v_{2j} / v_{1j} > r_{12}$  (thus belonging to  $S_{1<2}$ ), which satisfy  $v_{2j} / v_{1j} = r_{12}$  (belonging to  $S_{1=2}$ ), and which satisfy  $v_{2j} / v_{1j} < r_{12}$  (belonging to  $S_{1>2}$ ).

It is important to note that if  $v_{1j} = 0$  while  $v_{2j} > 0$ , the ratio  $\frac{v_{2j}}{v_{1j}}$  is considered to be  $\infty$ . Moreover, the case where  $v_{1j} = 0$  and  $v_{2j} = 0$  cannot occur, since we require that the highest bid  $h_j(\alpha)$  for each good be strictly positive.

Without loss of generality, we can assume that the goods are sorted in ascending order based on their value ratios  $v_{2j} / v_{1j}$ . Geometrically, these goods can be interpreted as  $m$  distinct points on the real line, where each point corresponds to a ratio  $v_{2j} / v_{1j}$ . It follows that the number of distinct cells of  $(S_\sigma)$  is at most  $2m + 1$ :  $m$  cells arise when  $r_{12}$  is exactly equal to some  $v_{2j} / v_{1j}$  for  $j \in [m]$ , and the other  $m + 1$  ones correspond to the cases where  $r_{12}$  lies strictly between adjacent ratios (or in the unbounded intervals  $(0, \min_j v_{2j} / v_{1j})$  and  $(\max_j v_{2j} / v_{1j}, \infty)$ ).

Therefore, the total number of possible cells of  $(S_\sigma)$  is  $O(m)$ .  $\square$

So we can enumerate all the possible cells with the above result.

Given a specific ordered tuple  $(S_\sigma)$ , the expression of  $p_j(\alpha)$  corresponding to each good can be explicitly determined.

In this case where  $n = 2$ ,  $p_j(\alpha) = \min\{\alpha_1 v_{1j}, \alpha_2 v_{2j}\}$ , and we can get that  $\forall j \in T_{\{1\}}, p_j(\alpha) = \alpha_2 v_{2j}$  and  $\forall j \in T_{\{2\}}, p_j(\alpha) = \alpha_1 v_{1j}$ , then we get that buyer 1 needs to pay  $\sum_{j \in T_{\{1\}}} p_j(\alpha) + \sum_{j \in T_{\{1,2\}}} x_{1j} \alpha_1 v_{1j}$ , and buyer 2 needs to pay  $\sum_{j \in T_{\{2\}}} p_j(\alpha) + \sum_{j \in T_{\{1,2\}}} x_{2j} \alpha_2 v_{2j}$ . Note that for each good  $j \in T_{\{1,2\}}$ , we have  $\alpha_1 v_{1j} = \alpha_2 v_{2j}$  and  $x_{1j} + x_{2j} = 1$ .

Subsequently, for each ordered tuple  $(S_\sigma)$ , we further categorize the problem into four cases based on the relative magnitudes of  $\alpha_1$ ,  $\alpha_2$  and 1. For each case, we construct a corresponding system of inequalities to facilitate the solution.

**Case 1:**  $\alpha_1 = 1, \alpha_2 = 1$ . In this case, we have the following:

$$\begin{cases} \sum_{j \in T_{\{1\}}} v_{2j} + \sum_{j \in T_{\{1,2\}}} x_{1j} v_{1j} \leq B_1 \\ \sum_{j \in T_{\{2\}}} v_{1j} + \sum_{j \in T_{\{1,2\}}} x_{2j} v_{2j} \leq B_2 \\ v_{1j} = v_{2j}, \forall j \in T_{\{1,2\}} \\ x_{1j} + x_{2j} = 1, \forall j \in T_{\{1,2\}} \\ 0 < x_{ij} < 1, \forall i \in \{1, 2\}, \forall j \in T_{\{1,2\}} \\ v_{1j} > v_{2j}, \forall j \in S_{1>2} \\ v_{1j} = v_{2j}, \forall j \in S_{1=2} \\ v_{1j} < v_{2j}, \forall j \in S_{1<2} \end{cases}$$

Standard linear programming techniques, such as the ellipsoid method proposed by Khachiyan (1979) or that of Karmarkar (1984), can be directly applied to solve this system. If a feasible solution exists, it corresponds to a pacing equilibrium.

**Case 2:**  $\alpha_1 < 1, \alpha_2 = 1$ . Buyer 1 exhausts her budget, which gives us the equation

$$\sum_{j \in T_{\{1\}}} v_{2j} + \sum_{j \in T_{\{1,2\}}} x_{1j} \alpha_1 v_{1j} = B_1.$$

Note that although the term  $x_{1j} \alpha_1 v_{1j}$  is quadratic in the variables  $x_{1j}$  and  $\alpha_1$ , for goods in the set  $T_{\{1,2\}}$ , we have the relation  $\alpha_2 v_{2j} = \alpha_1 v_{1j}$ . This allows us to rewrite the term as  $x_{1j} \alpha_2 v_{2j}$ , thereby reducing it to a linear expression.

Then we get the system below:

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{1\}}} v_{2j} + \sum_{j \in T_{\{1,2\}}} x_{1j} v_{2j} = B_1 \\ \sum_{j \in T_{\{2\}}} \alpha_1 v_{1j} + \sum_{j \in T_{\{1,2\}}} x_{2j} v_{2j} \leq B_2 \\ \alpha_1 v_{1j} = v_{2j}, \forall j \in T_{\{1,2\}} \\ x_{1j} + x_{2j} = 1, \forall j \in T_{\{1,2\}} \\ 0 < x_{ij} < 1, \forall i \in \{1, 2\}, \forall j \in T_{\{1,2\}} \\ 0 < \alpha_1 < 1 \\ \alpha_1 v_{1j} > v_{2j}, \forall j \in S_{1>2} \\ \alpha_1 v_{1j} = v_{2j}, \forall j \in S_{1=2} \\ \alpha_1 v_{1j} < v_{2j}, \forall j \in S_{1<2} \end{array} \right.$$

The existence of a solution to this linear system implies the existence of a pacing equilibrium in the second-price pacing game.

**Case 3:**  $\alpha_1 = 1, \alpha_2 < 1$ . Analogously, this case is almost identical to Case 2.

**Case 4:**  $\alpha_1 < 1, \alpha_2 < 1$ . It is similar to the previous situations that the following system holds:

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{1\}}} \alpha_2 v_{2j} + \sum_{j \in T_{\{1,2\}}} x_{1j} \alpha_1 v_{1j} = B_1 \\ \sum_{j \in T_{\{2\}}} \alpha_1 v_{1j} + \sum_{j \in T_{\{1,2\}}} x_{2j} \alpha_2 v_{2j} = B_2 \\ \alpha_1 v_{1j} = \alpha_2 v_{2j}, \forall j \in T_{\{1,2\}} \\ x_{1j} + x_{2j} = 1, \forall j \in T_{\{1,2\}} \\ 0 < x_{ij} < 1, \forall i \in \{1, 2\}, \forall j \in T_{\{1,2\}} \\ 0 < \alpha_1 < 1 \\ 0 < \alpha_2 < 1 \\ \alpha_1 v_{1j} > \alpha_2 v_{2j}, \forall j \in S_{1>2} \\ \alpha_1 v_{1j} = \alpha_2 v_{2j}, \forall j \in S_{1=2} \\ \alpha_1 v_{1j} < \alpha_2 v_{2j}, \forall j \in S_{1<2} \end{array} \right.$$

Since both  $x_{1j} \alpha_1 v_{1j}$  and  $x_{2j} \alpha_2 v_{2j}$  are quadratic terms. The observations that  $x_{1j} + x_{2j} = 1$  and  $\alpha_1 v_{1j} = \alpha_2 v_{2j}$

can help us simplify the equation system. Recall that  $y_{1j} = x_{1j} \alpha_1$  and  $y_{2j} = x_{2j} \alpha_2$ , we have

$$\begin{aligned} \sum_{j \in T_{\{1\}}} \alpha_2 v_{2j} + \sum_{j \in T_{\{1,2\}}} y_{1j} v_{1j} &= B_1, \\ \sum_{j \in T_{\{2\}}} \alpha_1 v_{1j} + \sum_{j \in T_{\{1,2\}}} y_{2j} v_{2j} &= B_2 \end{aligned}$$

and we use  $\alpha_1 v_{1j} = \alpha_2 v_{2j}$ , then we get:  $\alpha_1 / \alpha_2 = v_{2j} / v_{1j}, \forall j \in T_{\{1,2\}}$ .

Note that for any good  $j \in T_{\{1,2\}}$ , we have  $x_{1j} + x_{2j} = 1$ , which is equivalent to  $y_{1j} / \alpha_1 + y_{2j} / \alpha_2 = 1$ . Using the equation above, this yields the identity:

$$y_{1j} + \frac{v_{2j}}{v_{1j}} y_{2j} = \alpha_1,$$

that is,  $y_{1j} v_{1j} + y_{2j} v_{2j} = \alpha_1 v_{1j}$ , which means that when all buyers in  $I$  submit identical scaled bids for good  $j$ , under the second-price auction mechanism, we have that the total payment for good  $j$  equals any individual buyer's bid.

Hence we have:

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{1\}}} \alpha_2 v_{2j} + \sum_{j \in T_{\{1,2\}}} y_{1j} v_{1j} = B_1 \\ \sum_{j \in T_{\{2\}}} \alpha_1 v_{1j} + \sum_{j \in T_{\{1,2\}}} y_{2j} v_{2j} = B_2 \\ \alpha_1 v_{1j} = \alpha_2 v_{2j}, \forall j \in T_{\{1,2\}} \\ y_{1j} v_{1j} + y_{2j} v_{2j} = \alpha_1 v_{1j}, \forall j \in T_{\{1,2\}} \\ 0 < y_{ij} < \alpha_i, \forall i \in \{1, 2\}, \forall j \in T_{\{1,2\}} \\ 0 < \alpha_1 < 1 \\ 0 < \alpha_2 < 1 \\ \alpha_1 v_{1j} > \alpha_2 v_{2j}, \forall j \in S_{1>2} \\ \alpha_1 v_{1j} = \alpha_2 v_{2j}, \forall j \in S_{1=2} \\ \alpha_1 v_{1j} < \alpha_2 v_{2j}, \forall j \in S_{1<2} \end{array} \right.$$

This is a system of linear inequalities involving  $\alpha_1, \alpha_2$ , and  $y_{1j}, y_{2j}, \forall j \in [m]$ . If this system has a solution, then we can obtain a pacing equilibrium.

**Time Complexity.** The number of ordered tuples  $(S_\sigma)$  is  $O(m)$ , and the number of cases is 4, which is a constant. Then we get that the number of linear systems is  $O(m)$  and we can solve any linear system in polynomial time, so the algorithm can run in polynomial time.

## General Result

Things become complicate when we think about the general constant  $n = c$ .

**Pseudocode for the Algorithm.** For general instances where  $n = c$ , we propose the following algorithm:

---

Algorithm 1: Finding a pacing equilibrium in second-price auctions with constant buyers.

---

**Input:** Set of buyers  $B = \{1, \dots, c\}$ , goods  $G = \{1, \dots, m\}$ , budget  $B_i$  for each  $i \in B$  and valuation matrix  $(v_{ij})$ .

**Output:** Pacing multipliers  $(\alpha_i)$  and allocations  $(x_{ij})$ .

- 1: Enumerate all ordered tuples  $(S_\sigma)$
- 2: **for** each ordered tuple  $(S_\sigma)$  **do**
- 3:   Construct  $(T_I)$  corresponding to the  $(S_\sigma)$
- 4:   // Enumerate binary vectors of dimension  $c$
- 5:   **for** each binary vector  $b \in \{0, 1\}^c$  **do**
- 6:     **for** each  $b_i$  in  $b$  **do**
- 7:      **if**  $b_i = 1$  **then**
- 8:       Add constraint  $\alpha_i = 1$
- 9:      **else**
- 10:       Add constraint  $0 < \alpha_i < 1$
- 11:      **end if**
- 12:    **end for**
- 13:    Construct linear systems and solve the system to find  $(\alpha_i)$  and  $(x_{ij})$ .
- 14:    **if** there is a feasible solution satisfying all constraints **then**
- 15:     **return** the corresponding  $(\alpha_i)$  and  $(x_{ij})$ .
- 16:    **end if**
- 17: **end for**
- 18: **end for**

---

The construction of the linear system involved in the algorithm will be detailed in the proof of Theorem 4. We can show that this algorithm computes a pacing equilibrium in polynomial time.

**Theorem 4 (Main Result).** *Given any instance of second-price pacing game with a constant number  $c$  of buyers, Algorithm 1 computes a second-price pacing equilibrium in polynomial time.*

Before proving our main result, we can also prove that the number of distinct ordered tuples  $(S_\sigma)$  is bounded by a polynomial of  $m$ .

**Lemma 5.** *Given all  $v_{ij}$  and  $n = c$ , which is a constant, the number of distinct ordered tuple  $(S_\sigma)$ , over  $\alpha \in (0, 1]^c$  is bounded by a polynomial in  $m$ .*

*Proof.* For the case where  $n = c$ , we must compare  $\alpha_i v_{ij}$  and  $\alpha_{i'} v_{i'j}$  for all  $i, i'$  and all  $j$ . To facilitate this, consider the tuple  $(\alpha_1/\alpha_c, \alpha_2/\alpha_c, \dots, \alpha_{c-1}/\alpha_c)$ , denoted as  $(r_{1c}, r_{2c}, \dots, r_{(c-1)c})$ .

When  $i \neq c$  and  $i' = c$ , comparing  $\alpha_i v_{ij}$  with  $\alpha_{i'} v_{i'j}$  is equivalent to comparing  $r_{ic} \cdot v_{ij}$  with  $v_{cj}$ . When  $i \neq c$  and  $i' \neq c$ , the comparison between  $\alpha_i v_{ij}$  and  $\alpha_{i'} v_{i'j}$  becomes equivalent to comparing  $r_{ic} \cdot v_{ij}$  with  $r_{i'c} \cdot v_{i'j}$ .

Thus, for each good  $j$ , we consider hyperplanes in the  $(c-1)$ -dimensional space of the form  $r_{ic} v_{ij} = v_{cj}$  and  $r_{i'c} v_{i'j} = r_{i'c} v_{i'j}$ . Each good  $j$  induces  $(c-1)$  axis-aligned hyperplanes and  $(c-1)(c-2)/2$  origin-passing hyperplanes, resulting in a total of

$$m \cdot \left( (c-1) + \frac{(c-1)(c-2)}{2} \right) = \frac{mc(c-1)}{2}$$

hyperplanes overall.

By Theorem 6, we obtain that these hyperplanes partition the  $(c-1)$ -dimensional space into at most  $O(m^{c-1})$  distinct cells. Here, we confine our attention to the upper bound of  $O(m^{c-1})$ , which remains valid even when the hyperplanes are not in general position.  $\square$

**Theorem 6 (Zaslavsky (1975); Orlik and Terao (1992)).** *Let  $\mathcal{A}$  be a collection of  $k$  affine hyperplanes in general position in  $\mathbb{R}^d$ . Then the number of  $j$ -dimensional faces ( $0 \leq j \leq d$ ) induced by the arrangement is given by*

$$f_j = \binom{k}{d-j} \sum_{i=0}^j \binom{k-d+j}{i}.$$

*In particular, the number of  $d$ -dimensional cells (i.e., the connected components of  $\mathbb{R}^d \setminus \bigcup \mathcal{A}$ ) is*

$$f_d = \sum_{i=0}^d \binom{k}{i}.$$

Now we are ready to prove our main result.

*Proof of Theorem 4.* By Lemma 5, we have that the ordered tuple  $(S_\sigma)$  is polynomial in  $m$ . Thus, it is sufficient to show that for each specific  $(S_\sigma)$ , a pacing equilibrium can be computed in polynomial time.

Given a specific ordered tuple  $(S_\sigma)$ , the expression of  $p_j(\alpha)$  can be explicitly determined for each good  $j$ . In particular, if good  $j$  is included in set  $T_{\{i\}}$ , then  $x_{ij} = 1$ , which implies that the corresponding function  $p_j(\alpha)$  is linear in  $\alpha$ .

For each  $(S_\sigma)$ , we can decompose it into  $2^c$  cases and conduct a case-by-case analysis to facilitate the solution.

The  $2^c$  possible cases can be broadly classified into three categories: (i) all  $\alpha_i = 1$ , (ii) all  $\alpha_i < 1$ , and (iii) the remaining  $2^c - 2$  mixed cases, where some but not all  $\alpha_i$ 's equal 1. We analyze each of these categories separately.

**Case 1:**  $\alpha_i = 1$  for each  $i \in [c]$ .

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\{i\} \subset I \subseteq [c]} \sum_{j \in T_I} x_{ij} \alpha_i v_{ij} \leq B_i, \forall i \in [c] \\ \alpha_i v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq [c], |I| \geq 2, \forall i, i' \in I, \forall j \in T_I \\ \sum_{i \in I} x_{ij} = 1, \forall I \subseteq [c], |I| \geq 2, \forall j \in T_I \\ 0 < x_{ij} < 1, \forall I \subseteq [c], |I| \geq 2, \forall i \in I, \forall j \in T_I \\ \alpha_i = 1, \forall i \in [c] \\ \alpha_1 v_{1j} > \dots > \alpha_{c-1} v_{(c-1)j} > \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c-1>c} \\ \alpha_1 v_{1j} > \dots > \alpha_{c-1} v_{(c-1)j} = \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c-1=c} \\ \alpha_1 v_{1j} > \dots > \alpha_c v_{cj} > \alpha_{c-1} v_{(c-1)j}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c>c-1} \\ \dots \\ \alpha_c v_{cj} > \dots > \alpha_2 v_{2j} > \alpha_1 v_{1j}, \text{ where} \\ \quad \forall j \in S_{c>c-1>\dots>1} \end{array} \right.$$

In this setting, the above constraints constitute a linear system over the variables  $(x_{ij})$ . Solving this linear system. If a solution exists, then we get a pacing equilibrium.

**Case 2: some but not all of  $\alpha_i$ 's equal 1.** Note that there are actually  $2^c - 2$  possible cases. Given each case, we denote by  $E$  the set of buyers  $i$  satisfying  $\alpha_i = 1$ , and by  $L$  the set of buyers  $i'$  satisfying  $\alpha_{i'} < 1$ .

$$E := \{i \in [c] : \alpha_i = 1\}, L := \{i' \in [c] : \alpha_{i'} < 1\}.$$

All buyers in  $L$  exhaust their budgets, and consequently, we impose the following constraints.

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\{i\} \subset I \subseteq [c]} \sum_{j \in T_I} x_{ij} \alpha_i v_{ij} \leq B_i, \forall i \in E \\ \sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\{i\} \subset I \subseteq [c]} \sum_{j \in T_I} x_{ij} \alpha_i v_{ij} = B_i, \forall i \in L \\ \alpha_i v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq [c], |I| \geq 2, \forall i, i' \in I, \forall j \in T_I \\ \sum_{i \in I} x_{ij} = 1, \forall I \subseteq [c], |I| \geq 2, \forall j \in T_I \\ 0 < x_{ij} < 1, \forall I \subseteq [c], |I| \geq 2, \forall i \in I, \forall j \in T_I \\ \alpha_i = 1, \forall i \in E \\ 0 < \alpha_i < 1, \forall i \in L \\ \alpha_1 v_{1j} > \dots > \alpha_{c-1} v_{(c-1)j} > \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c-1>c} \\ \alpha_1 v_{1j} > \dots > \alpha_{c-1} v_{(c-1)j} = \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c-1=c} \\ \alpha_1 v_{1j} > \dots > \alpha_c v_{cj} > \alpha_{c-1} v_{(c-1)j}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c>c-1} \\ \dots \\ \alpha_c v_{cj} > \dots > \alpha_2 v_{2j} > \alpha_1 v_{1j}, \text{ where} \\ \quad \forall j \in S_{c>c-1>\dots>1} \end{array} \right.$$

In this case, the variables include  $\alpha_{i'}$  for each  $i' \in L$ , as well as the allocation variables  $x_{ij}$  corresponding to goods that are assigned to multiple buyers.

Under these conditions, for all  $i' \in L$ , the term  $x_{i'j} \alpha_{i'} v_{i'j}$  is quadratic term. Notably, these quadratic terms arise only when a good is allocated to multiple buyers whose bids are equal. Consider one such quadratic term,  $x_{i'_0j} \alpha_{i'_0} v_{i'_0j}$ . The treatment of these quadratic terms requires a case-by-case analysis, which we divide into the following two scenarios.

If there exists a buyer  $i \in E$  such that their bid for good  $j$  equals that of buyer  $i'_0$ , i.e.,  $\alpha_{i'_0} v_{i'_0j} = \alpha_i v_{ij}$ , then the quadratic term becomes  $x_{i'_0j} \alpha_i v_{ij}$ , which is linear in  $x_{i'_0j}$ . Thus, the quadratic term can be eliminated in this case.

If no such buyer  $i \in E$  exists, and instead there are  $k \geq 1$  buyers such that  $i'_1, \dots, i'_k \in L$  whose bids for good  $j$  are equal to that of  $i'_0$ , then we proceed as follows. Let us define new variables  $y_{i'_0j} = x_{i'_0j} \alpha_{i'_0}$ ,  $\dots$ ,  $y_{i'_kj} = x_{i'_kj} \alpha_{i'_k}$ . Our goal is to eliminate the original allocation variables  $x_{i'_0j}, \dots, x_{i'_kj}$ . First, all associated quadratic terms  $x_{i'_lj} \alpha_{i'_l} v_{i'_lj}$  for  $0 \leq l \leq k$  become linear in the form

$y_{i'_lj} v_{i'_lj}$ . Second, to handle the original constraints involving  $x_{i'_lj}$ , we make the following transformations: The bounds  $0 < x_{i'_lj} < 1$  become  $0 < y_{i'_lj} < \alpha_{i'_l}$ , for all  $0 \leq l \leq k$ .

To reformulate the constraint  $\sum_{0 \leq l \leq k} x_{i'_lj} = 1$ , we observe that this is equivalent to  $\sum_{0 \leq l \leq k} \frac{y_{i'_lj}}{\alpha_{i'_l}} = 1$ , which can be rewritten as

$$y_{i'_0j} + \sum_{1 \leq l \leq k} y_{i'_lj} \cdot \frac{\alpha_{i'_0}}{\alpha_{i'_l}} = \alpha_{i'_0}.$$

Next, we employ the identities:

$$\alpha_i v_{ij} = \alpha_{i'} v_{i'j}, \quad \forall j \in T_I, \{i\} \subset I \subseteq [c], \forall i, i' \in I,$$

which means that all buyers in  $I$  submit identical scaled bids for good  $j$ .

Under the second-price auction mechanism, this leads to the conclusion that the total payment for good  $j$  equals any individual buyer's bid. Consequently, we have:

$$\sum_{0 \leq l \leq k} y_{i'_lj} v_{i'_lj} = \alpha_{i'_0} v_{i'_0j}.$$

Through these transformations, the original constraint is thus converted into the above equivalent form.

Consequently, the original system can be reduced to the following linear system.

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\{i\} \subset I \subseteq [c]} \sum_{j \in T_I} x_{ij} \alpha_i v_{ij} \leq B_i, \forall i \in E \\ \sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\substack{\{i\} \subset I \subseteq [c] \\ \exists i' \in I, i' \in E}} \sum_{j \in T_I} x_{ij} \alpha_{i'} v_{i'j} \\ \quad + \sum_{\substack{\{i\} \subset I \subseteq [c] \\ \forall i' \in I, i' \in L}} \sum_{j \in T_I} y_{ij} v_{ij} = B_i, \forall i \in L \\ \alpha_i v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq [c], |I| \geq 2, \forall i, i' \in I, \forall j \in T_I \\ \sum_{i \in I} x_{ij} = 1, \forall I \subseteq [c], I \not\subseteq L, |I| \geq 2, \forall j \in T_I \\ 0 < x_{ij} < 1, \forall I \subseteq [c], I \not\subseteq L, |I| \geq 2, \forall j \in T_I \\ \sum_{i \in I} y_{ij} v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq L, |I| \geq 2, \forall j \in T_I \\ 0 < y_{ij} < \alpha_i, \forall I \subseteq L, |I| \geq 2, \forall j \in T_I \\ \alpha_i = 1, \forall i \in E \\ 0 < \alpha_i < 1, \forall i \in L \\ \alpha_1 v_{1j} > \dots > \alpha_{c-1} v_{(c-1)j} > \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c-1>c} \\ \alpha_1 v_{1j} > \dots > \alpha_{c-1} v_{(c-1)j} = \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c-1=c} \\ \alpha_1 v_{1j} > \dots > \alpha_c v_{cj} > \alpha_{c-1} v_{(c-1)j}, \text{ where} \\ \quad \forall j \in S_{1>2>\dots>c>c-1} \\ \dots \\ \alpha_c v_{cj} > \dots > \alpha_2 v_{2j} > \alpha_1 v_{1j}, \text{ where} \\ \quad \forall j \in S_{c>c-1>\dots>1} \end{array} \right.$$

It can be observed that each quadratic term in this constraint has been transformed into a linear term. Therefore, we can directly apply linear programming techniques to solve this system. If a feasible solution exists, it corresponds to a pacing equilibrium.

**Case 3:**  $\alpha_i < 1$  for each  $i \in [c]$ . In this case, all buyers exhaust their budgets, and we also use  $y_{ij} = x_{ij}, \alpha_i$ , we can get:

$$\sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\{i\} \subset I \subseteq [c]} \sum_{j \in T_I} y_{ij} v_{ij} = B_i, \forall i \in [c],$$

which means that all buyers have exhausted their budgets.

Additionally, as in Case 2, the following relation also holds:

$$0 < y_{ij} < \alpha_i, \forall I \subseteq [c], |I| \geq 2, \forall i' \in I, \forall j \in T_I,$$

and

$$\sum_{i \in I} y_{ij} v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq [c], |I| \geq 2, \forall i' \in I, \forall j \in T_I.$$

Upon applying the same idea, we result in the following set of constraints.

$$\left\{ \begin{array}{l} \sum_{j \in T_{\{i\}}} p_j(\alpha) + \sum_{\{i\} \subset I \subseteq [c]} \sum_{j \in T_I} y_{ij} v_{ij} = B_i, \forall i \in [c] \\ \alpha_i v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq [c], |I| \geq 2, \forall i, i' \in I, \forall j \in T_I \\ \sum_{i \in I} y_{ij} v_{ij} = \alpha_{i'} v_{i'j}, \forall I \subseteq [c], |I| \geq 2, \forall i' \in I, \forall j \in T_I \\ 0 < y_{ij} < \alpha_i, \forall I \subseteq [c], |I| \geq 2, \forall i \in I, \forall j \in T_I \\ 0 < \alpha_i < 1, \forall i \in [c] \\ \alpha_1 v_{1j} > \cdots > \alpha_{c-1} v_{(c-1)j} > \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\cdots>c-1>c} \\ \alpha_1 v_{1j} > \cdots > \alpha_{c-1} v_{(c-1)j} = \alpha_c v_{cj}, \text{ where} \\ \quad \forall j \in S_{1>2>\cdots>c-1=c} \\ \alpha_1 v_{1j} > \cdots > \alpha_c v_{cj} > \alpha_{c-1} v_{(c-1)j}, \text{ where} \\ \quad \forall j \in S_{1>2>\cdots>c>c-1} \\ \quad \dots \\ \alpha_c v_{cj} > \cdots > \alpha_2 v_{2j} > \alpha_1 v_{1j}, \text{ where} \\ \quad \forall j \in S_{c>c-1>\cdots>1} \end{array} \right.$$

The variables involved are  $y_{ij}$  and  $\alpha_i$ . Solving this system of linear inequalities, if a solution exists, yields a pacing equilibrium.

**Time Complexity.** By Lemma 5, we can deduce that the number of ordered tuples ( $S_\sigma$ ) is  $O(m^{c-1})$ . Then, according to the algorithm, for each ordered tuple ( $S_\sigma$ ), there are  $2^c$  possible cases, where  $2^c$  is a constant. Each case corresponds to a linear system, and each linear system can be solved in polynomial time. Therefore, the algorithm can run in polynomial time.

As for the computation of the LP with  $n$  variables, Cohen, Lee, and Song (2019) established the time complexity

of  $O(n^\omega \log(\frac{n}{\delta}))$ , where  $\omega \approx 2.37$  denotes the matrix multiplication exponent and  $\delta$  represents the relative accuracy. Integrating their result with our algorithm yields a total time complexity of  $O(m^{c-1+\omega} \log(\frac{m}{\delta}))$ .  $\square$

## Conclusion

We focus on the setting where the number of buyers in second-price auctions is constant. In this restricted regime, we present the first polynomial-time algorithm for computing an exact second-price pacing equilibrium (SPPE). Our approach leverages the cell-decomposition method. To make each cell tractable, we transform the formulation into a linear program by introducing auxiliary variables, which allows us to solve each case where the pacing multipliers are controlled efficiently.

A natural direction for future work is to explore the dual setting where the number of goods is constant. More importantly, obtaining a constant additive approximation of an SPPE remains an open challenge. We conjecture that the  $1/3$ -approximation bound recently established by Chen and Li (2025) is tight.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grants 62472029 and 62572476) and the Key Laboratory of Interdisciplinary Research of Computation and Economics (Shanghai University of Finance and Economics), Ministry of Education.

## References

- Aggarwal, G.; Badanidiyuru, A.; Balseiro, S. R.; Bhawalkar, K.; Deng, Y.; Feng, Z.; Goel, G.; Liaw, C.; Lu, H.; Mahdian, M.; Mao, J.; Mehta, A.; Mirrokni, V.; Leme, R. P.; Perloff, A.; Piliouras, G.; Schneider, J.; Schwartzman, A.; Sivan, B.; Spendlove, K.; Teng, Y.; Wang, D.; Zhang, H.; Zhao, M.; Zhu, W.; and Zuo, S. 2024. Auto-Bidding and Auctions in Online Advertising: A Survey. *SIGecom Exch.*, 22(1): 159–183.
- Balseiro, S. R.; Bhawalkar, K.; Feng, Z.; Lu, H.; Mirrokni, V.; Sivan, B.; and Wang, D. 2024. A Field Guide for Pacing Budget and ROS Constraints. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*. OpenReview.net.
- Balseiro, S. R.; and Gur, Y. 2019. Learning in Repeated Auctions with Budgets: Regret Minimization and Equilibrium. *Management Science*, 65(9): 3952–3968.
- Borgs, C.; Chayes, J. T.; Immorlica, N.; Jain, K.; Etesami, O.; and Mahdian, M. 2007. Dynamics of Bid Optimization in Online Advertisement Auctions. In *Proceedings of the 16th International Conference on World Wide Web (WWW)*, 531–540. ACM.
- Chen, X.; Deng, X.; and Teng, S.-H. 2006. Sparse games are hard. In *International Workshop on Internet and Network Economics*, 262–273. Springer.

- Chen, X.; Diakonikolas, I.; Paparas, D.; Sun, X.; and Yannakakis, M. 2018. The complexity of optimal multidimensional pricing for a unit-demand buyer. *Games and Economic Behavior*, 110: 139–164.
- Chen, X.; Kroer, C.; and Kumar, R. 2021. Throttling Equilibria in Auction Markets. In *Proceedings of the 17th Conference on Web and Internet Economics (WINE)*, volume 13112 of *Lecture Notes in Computer Science*, 551. Springer.
- Chen, X.; Kroer, C.; and Kumar, R. 2023. The Complexity of Pacing for Second-Price Auctions. *Mathematics of Operations Research*, 49(4): 2109–2135.
- Chen, X.; and Li, Y. 2025. Constant Inapproximability of Pacing Equilibria in Second-Price Auctions. *arXiv preprint arXiv:2501.15295*.
- Cohen, M. B.; Lee, Y. T.; and Song, Z. 2019. Solving linear programs in the current matrix multiplication time. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019*, 938–942. New York, NY, USA: Association for Computing Machinery. ISBN 9781450367059.
- Conitzer, V.; Kroer, C.; Sodomka, E.; and Moses, N. E. S. 2022. Multiplicative Pacing Equilibria in Auction Markets. *Operations Research*, 70(2): 963–989.
- Devanur, N. R.; and Kannan, R. 2008. Market equilibria in polynomial time for fixed number of goods or agents. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, 45–53. IEEE.
- Karmarkar, N. 1984. A new polynomial-time algorithm for linear programming. In *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing, STOC '84*, 302–311. New York, NY, USA: Association for Computing Machinery. ISBN 0897911334.
- Khachiyan, L. G. 1979. A polynomial algorithm in linear programming. *Dokl. Akad. Nauk SSSR*, 244(5): 1093–1096.
- Liu, Z.; Li, J.; and Deng, X. 2021. On the approximation of Nash equilibria in sparse win-lose multi-player games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 5557–5565.
- Liu, Z.; and Sheng, Y. 2018. On the approximation of Nash equilibria in sparse win-lose games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
- Lucier, B.; Pattathil, S.; Slivkins, A.; and Zhang, M. 2024. Autobidders with Budget and ROI Constraints: Efficiency, Regret, and Pacing Dynamics. In *Proceedings of the 37th Annual Conference on Learning Theory (COLT)*, volume 247 of *Proceedings of Machine Learning Research*, 3642–3643. PMLR.
- Orlik, P.; and Terao, H. 1992. *Arrangements of Hyperplanes*, volume 300 of *Grundlehren der mathematischen Wissenschaften*. Berlin: Springer-Verlag. ISBN 978-3-540-55259-3.
- Wang, Q.; Yang, Z.; Deng, X.; and Kong, Y. 2023. Learning to Bid in Repeated First-Price Auctions with Budgets. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, 36494–36513. PMLR.
- Zaslavsky, T. 1975. Facing up to arrangements: face-count formulas for partitions of space by hyperplanes. *Memoirs of the American Mathematical Society*, 1(154): 1–102.