Exploring Leximin Principle for Fair Core-Selecting Combinatorial Auctions: Payment Rule Design and Implementation

Hao Cheng[1] *, **Shufeng Kong**[2] †, **Yanchen Deng**[3], **Caihua Liu**[4], **Xiaohu Wu**[5], **Bo An**[3], **Chongjun Wang**[1]

[1]State Key Laboratory for Novel Software Technology, Nanjing University, China
[2]School of Software Engineering, Sun Yat-Sen University, China
[3]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[4]School of Artificial Intelligence, Guilin University of Electronic Technology, China
[5]National Engineering Research Center of Mobile Network Technologies, Beijing University of Posts and Telecommunications, China
chengh@smail.nju.edu.cn, kongshf@mail.sysu.edu.cn, ycdeng@ntu.edu.sg, caihua.liu@guet.edu.cn, xiaohu.wu@bupt.edu.cn, boan@ntu.edu.sg, chjwang@nju.edu.cn

Abstract

Core-selecting combinatorial auctions (CAs) restrict the auction result in the core such that no coalitions could improve their utilities by engaging in collusion. The minimum-revenue-core (MRC) rule is a widely used core-selecting payment rule to maximize the total utilities of all bidders. However, the MRC rule can suffer from severe unfairness since it ignores individuals' utilities. To address this limitation, we propose to explore the leximin principle to achieve fairness in core-selecting CAs since the leximin principle prefers to maximize the utility of the worst-off; the resulting bidder-leximin-optimal (BLO) payment rule is then theoretically analyzed and an effective algorithm is further provided to compute the BLO outcome. Moreover, we conduct extensive experiments to show that our algorithm returns fairer utility distributions and is faster than existing algorithms of core-selecting payment rules.

1 Introduction

Combinatorial auctions (CAs) have found applications in many real-world auction problems, including procurement auctions [Sandholm, 2007] and government spectrum auctions [Cramton, 2013; Ausubel and Baranov, 2017; Leyton-Brown *et al.*, 2017]. CAs are attractive as they can produce efficient outcomes even when bidders have complex preferences on bundles of heterogeneous goods. A combinatorial auction mechanism often includes two components: 1) an allocation rule to specify the bundle of items assigned to each bidder; 2) a payment rule to specify the price a bidder should pay for his bundle. Note that an allocation rule is normally obtained by solving the NP-hard winner determination problem [Rothkopf *et al.*, 1998]. In this paper, we will focus on the payment rules' designs and implementations.

To address the truthfulness issue of bidding, where bidders are unwilling to give their real valuations of items but report false information to improve their utilities, the well-known Vickrey-Clarke-Groves (VCG) payment rule [Vickrey, 1961; Clarke, 1971; Groves, 1973] ensures that the optimal strategy for each bidder is to bid their true valuations of the items. However, the VCG payment rule still suffers from various other issues that prohibit it from real-world adoption, e.g., arbitrarily low revenue for the seller, which may cause collusion between the seller and some bidders [Ausubel *et al.*, 2006].

The drawbacks of the VCG payment rule have motivated the development of core-selecting payment rules [Day and Raghavan, 2007], which provide a principled way to ensure that there are no opportunities for collusion between the seller and bidders. The core is a set of all possible payment outcomes where the utility of any coalition of bidders in the auction cannot be improved by engaging in collusion [Gillies, 1959]. However, the core can be exponentially large, and thus, selecting a suitable payment outcome from the core is a critical problem in core-selecting CAs. Many efforts have been devoted to solving this problem by designing some principles of selection [Day and Raghavan, 2007; Erdil and Klemperer, 2010; Day and Cramton, 2012; Lubin *et al.*, 2015; Bünz *et al.*, 2018]. Among them, the minimum-revenue-core (MRC) rule is particularly popular, where it achieves the largest total incentive by selecting the core outcome that maximizes the total utilities of all bidders.

However, it is observed that the MRC rule may cause severe unfairness, since some winners may be forced to sacrifice their utilities in the MRC rule. A detailed example to illustrate this unfair phenomenon caused by the MRC rule is provided in Example 1 of this paper, where, in the optimal MRC payment outcome, a winner's utility is 0 while he could have obtained a positive utility. More importantly, this zero-utility phenomenon is not rare since nearly 30% of winners are forced to get a utility of zero on average in our data gener-

*The work was done during a visiting scholar program at Nanyang Technological University.

†Corresponding author

ated by CATS [Leyton-Brown *et al.*, 2000], which simulates bidding behavior in many realistic economic environments.

To address the limitation of the MRC rule, we propose to explore the leximin principle to achieve fairness in core-selecting CAs and propose a novel bidder-leximin-optimal (BLO) payment rule that selects a core outcome based on the leximin principle which prefers to maximize the utility of the worst-off [Rawls, 2004]. Informally, the BLO rule maximizes the minimum utility, the second minimum utility and so forth among all the bidders, which achieves egalitarian fairness. Furthermore, we analyze the BLO outcome theoretically and provide an effective algorithm to compute it. We summarize our contributions[1] as follows:

1. We investigate the unfair phenomenon of the MRC rule and propose the BLO rule to achieve fairness in the core-selecting CAs.
2. We give a theoretical analysis of the BLO rule and show that the BLO outcome is unique and Pareto optimal. Moreover, we offer that the worst-case relative ratio between a bidder's utility (resp. the total utility of all bidders) in the BLO outcome and the maximum utility among all the core outcomes is $\frac{1}{|W|}$ (resp. $\frac{4}{|W|+2+\frac{|W| \mod 2}{|W|}}$), where $|W|$ is the winner size.
3. Finally, we propose an exact algorithm named WF-CGS-CR to compute the BLO outcome. Our WF-CGS-CR algorithm consists of the water-filling framework, a constraint generation search subroutine, and a constraint-reuse strategy. Given the oracle access for winner determination, it is effective in time polynomial in the number of winners. Extensive experiments show that our algorithm returns fairer utility distributions and is also faster than existing algorithms of core-selecting payment rules.

2 Related Work

As a practical mechanism, core-selecting CAs and the MRC rule were proposed in [Day and Raghavan, 2007] and used in several scenarios, including selling wireless spectra [Cramton, 2013] and online advertising [Goel *et al.*, 2015]. Then the quadratic rule [Day and Cramton, 2012] was proposed to select the MRC point nearest to the VCG point in l_2-distance. They also study some other reference points like all-zero points. Recently, some works design the core-selecting payment rule by approximating Bayesian equilibrium of different core outcomes [Lubin *et al.*, 2015; Bünz *et al.*, 2018].

Except for the incentives, fairness is an essential metric in the core-selecting mechanisms. [Lubin *et al.*, 2015] studied the fairness measured by the Gini coefficient and proposed a fractional rule that selects a point fraction to the VCG point performs best. However, their experiments show that relaxing the MRC constraint does not change the equilibrium properties; and there are no systematic shifts in efficiency, fairness, and aggregate incentives. There are many principles to define fairness. In this paper, we adopt the leximin principle in the core-selecting CAs. The method is similar to the egalitarian core proposed by Arin [Arin and Inarra, 2001], which selects the utility distribution in the cooperative game via the leximin principle [Rawls, 2004]. The difference is that the total utility is not constant in core-selecting CAs.

As for the pricing algorithms, given access to the separation oracle of winner determination, one can use the Ellipsoid algorithm [Grötschel *et al.*, 1981] to optimize convex objectives (e.g., bidders' total utility) exactly over the core polytope in polynomial time. However, this approach is rather slow in practice, and Core Constraint Generation (CCG) algorithm [Day and Raghavan, 2007] was proposed as an effective heuristic algorithm to solve such an optimization problem, which is commonly used in core-selecting CAs [Day and Cramton, 2012; Cramton, 2013; Bünz *et al.*, 2015].

However, it is hard to formulate a global objective for the BLO rule, making it difficult to use the above algorithms directly. Then, Fast Core algorithm [Niazadeh *et al.*, 2022] was proposed to compute an approximate Pareto optimal outcome in the rich advertisement, which can be used to compute an approximate BLO core outcome. Instead of the approximate algorithm, we focus on the exact algorithm. Moreover, the experiments show that our exact algorithm is much faster than Fast Core.

3 Preliminaries

Combinatorial auctions are specifically designed for domains where bidders can have complex preferences over bundles of heterogeneous items. Formally, let $N = \{1, \ldots, n\}$ be a set of bidders and s be the seller. Let $M = \{a, b, c, \ldots\}$ be the set of items with $|M| = m$. Each bidder i has a valuation function $v_i : 2^M \to \mathbb{R}$ which specifies the bidder's valuation for every possible bundle of items $S \in 2^M$. Unless stated otherwise, bidders will only need to give their values for bundles in which they are interested, and all other bundles will be given values 0. Also, we assume that $v_i(\emptyset) = 0$ $(\forall i \in N)$.

A CA consists of two steps: step 1 computes an item allocation outcome, and step 2 computes an item payment outcome. An allocation rule specifies the bundle of items a_i assigned for each bidder i, where $a_i \in 2^M$ and $a_i \cap a_j = \emptyset$ $(\forall i \neq j)$, and a payment rule specifies the price p_i that the bidder i should pay for his bundle a_i. Obviously, p_i must be 0 if a_i is empty.

Let π_i denote the utility of an auction participant $i \in N \cup \{s\}$. If i is the bidder, we assume that his utility is quasi-linear, i.e., $\pi_i = v_i(a_i) - p_i$. If i is the seller s, his utility is $\pi_s = \sum_{i \in N} p_i$, which is also known as the revenue. The *social welfare* is the summation of all participants' utility:

$$\pi_s + \sum_{i \in N} \pi_i = \sum_{i \in N} p_i + \sum_{i \in N} (v_i(a_i) - p_i) = \sum_{i \in N} v_i(a_i) \quad (1)$$

One can observe that the social welfare does not depend on the payment, and an allocation rule can be computed by maximizing the social welfare as follows:

$$\max_{\{a_i | a_i \subseteq M, i \in N, a_i \cap a_j = \emptyset, \forall i \neq j\}} \sum_{i \in N} v_i(a_i) \quad (2)$$

Note that the above optimization problem is also known as the *winner determination problem*. Let $w(\{v_i(\cdot)\}_{i \in N})$ be the

[1]Note that we will refer all proofs to the appendix. Our appendix and code are available at https://github.com/ha0cheng/Fair-BLO.

maximum social welfare given the valuation function profile $\{v_i(\cdot)\}_{i\in N}$. Once we have a social welfare maximizing allocation $\{a_i|i \in N, a_i \cap a_j = \emptyset, \forall i \neq j\}$ by solving the winner determination problem, in step 2, we need to decide the price p_i for each allocated item bundle $a_i (i \in N)$.

The simplest payment rule is the so-called *pay-as-bid* payment rule, where a winner $i(a_i \neq \emptyset)$ pays the price $p_i = v_i(a_i)$ for buying the item bundle a_i. However, the pay-as-bid payment rule suffers from the truthfulness issue, where bidders are unwilling to report the true valuation functions but strategically give false information instead. Therefore, in step 2, the famous VCG mechanism is proposed to offer bidders incentives to ensure that bidding truthfully is the dominant strategy. Unfortunately, VCG can lead to very low or even zero revenue, which is unstable and may cause collusion between the seller and some bidders.

3.1 Core-selecting Payment Rules

The weaknesses of VCG payment rule have motivated the development of *core-selecting payment rules*, which provide a principled way to ensure that the revenue of the auction is high enough such that there are no opportunities for collusion.

Formally, let $S \subseteq N \cup \{s\}$ denote a coalition of participants.Then, if $s \notin S$, the social welfare of S must be 0 since the auction cannot happen without the seller's participation; If $s \in S$, the social welfare of S can be computed as follows:

$$\pi_s + \sum_{i\in S\setminus\{s\}} \pi_i = \sum_{i\in S\setminus\{s\}} p_i + \sum_{i\in S\setminus\{s\}} (v_i(a_i) - p_i) = \sum_{i\in S\setminus\{s\}} v_i(a_i) \tag{3}$$

The coalition characteristic function, $\mathcal{F} : 2^{N\cup\{s\}} \to \mathbb{R}$, which indicates the optimal social welfare of each coalition S is then defined as follows:

$$\mathcal{F}(S) = \begin{cases} w(\{v_i(\cdot)\}_{i\in S\setminus\{s\}}), \text{if } s \in S \\ 0, \text{if } s \notin S \end{cases} \tag{4}$$

The *core* is the set of all possible outcomes where the total utility of any coalition cannot be improved by engaging in collusion [Gillies, 1959]. The definition is given as follows:

Definition 1 (Core)**.** *The core of a CA is a set of outcomes satisfying the following two conditions:*

- *Efficiency:*

$$\sum_{i\in N\cup\{s\}} \pi_i = \mathcal{F}(N \cup \{s\}) \tag{5}$$

- *Coalitional rationality:*

$$\sum_{i\in S} \pi_i \geq \mathcal{F}(S),\ \forall S \subseteq N \cup \{s\} \tag{6}$$

Note that the efficiency condition requires that the allocation outcome must be optimal, namely, the allocations are obtained by solving the winner determination problem. It is possible that there is more than one optimal allocation, and we break ties by selecting a random optimal allocation $\{a_i^*\}_{i\in N}$. Denote by W the winner set, i.e., $W = \{i|a_i^* \neq \emptyset\}$. On the other hand, the coalitional rationality condition guarantees that the maximum social welfare that any coalition S could get (i.e., RHS of the inequality) is not larger than what the coalition can already get under the current outcome (i.e., LHS of the inequality). *Finally, a core-selecting CA selects a social welfare maximizing allocation and picks a payment outcome in the core.*

If coalition S does not include the seller, then $\mathcal{F}(S) = 0$. Thus the corresponding constraints of coalitional rationality can be summarized as follows:

$$\pi_i \geq 0,\ \forall i \in N \tag{7}$$

If $s \in S$, substitute Eq.5 into Eqs.6 and we can arrange the constraints as follows:

$$\sum_{i\in N\setminus S} \pi_i \leq w(N) - w(S),\ \forall S \subseteq N \tag{8}$$

where $w(S)$ is the abbreviation of $w(\{v_i(\cdot)\}_{i\in S})$. Eqs.7 and Eqs.8 form the complete core constraint set, which includes $2^N + N$ constraints in total. We use the utility to analyze in the following discussion and each bidder's payment price is mapped by $p_i = v_i(a_i^*) - \pi_i$.

3.2 Minimum-Revenue-Core Payment Rule

Core-selecting CAs are not guaranteed to be truthful [Goeree and Lien, 2016]. To maximize the incentives of truthful bidding, the *Minimum-Revenue-Core rule* (MRC) is proposed as a fundamental principle to select the core outcomes [Day and Raghavan, 2007; Day and Milgrom, 2008; Day and Cramton, 2012]. The MRC rule maximizes the total utility of all bidders, which can be computed as follows:

$$\max_{\{\pi_i\}_{i\in N}\in U} \sum_{i\in N} \pi_i \tag{9}$$

where U is the core of the related CA.

Unfortunately, we observe that the MRC rule could suffer from severe unfairness of utility distribution, where some winners' utilities are forced to be zero. This unfair phenomenon largely dues to the fact that the MRC rule emphasizes the total utility and ignores individuals' utilities. We will illustrate this unfair utility distribution phenomenon in the following example.

Example 1. *Consider a CA with* 5 *bidders and* 3 *items. Let* $N = \{1, 2, 3, 4, 5\}$ *be the set of bidders and* $M = \{a, b, c\}$ *be the set of items. Table 1 gives the valuation functions of all bidders. By solving the winner determination problem, we have that the winners are bidders 1, 2, and 3, thus the maximum social welfare is that* $w(N) = v_1(\{a\}) + v_2(\{b\}) + v_3(\{c\}) = 6$ *according to Eq.1. Similarly,* $w(\{3, 4, 5\}) = 4$ *and the corresponding constraint is* $\pi_1 + \pi_2 \leq 6 - 4 = 2$. *The other constraints are obtained similarly. After removing the redundant constraints, the core of this example is shown as follows:*

$$\textit{Core} = \begin{cases} \pi_1 \geq 0, \pi_2 \geq 0, \pi_3 \geq 0 \\ \pi_1 + \pi_2 \leq 2 \\ \pi_2 + \pi_3 \leq 2 \end{cases} \tag{10}$$

According to the MRC rule, the unique utility distribution is $\{2, 0, 2\}$. *Bidder 2 must sacrifice all of his utility even* *though he could have obtained a positive utility in other core outcomes (e.g., $\{1,1,1\}$). In this case, bidder 2 would prefer bidding untruthfully to get more utility. For example, he can bid the value 1 for the item b, which leads to a utility of 1 in the MRC rule.*

Table 1: A CA with 5 bidders and 3 items. Winning bids are marked with "*". The last three columns are utilities computed from the VCG, MRC, and our new BLO rule, respectively.

	bids					utilities		
	{a}	{b}	{c}	{a,b}	{b,c}	VCG	MRC	BLO
Bidder 1	2*					2	2	1
Bidder 2		2*				2	0	1
Bidder 3			2*			2	2	1
Bidder 4				2		0	0	0
Bidder 5					2	0	0	0

Extending this example to the general case, we have the following lemma:

Lemma 1. *Assuming that bidders bid truthfully, in the worst case, $|W|-2$ winners may get zero utility in the MRC rule, even though there exists some core outcome that assigns them with positive utilities, where $|W|$ is the winner size.*

Moreover, we observe that the unfair phenomenon often happens in a wide range of domains and datasets when using the MRC rule. To address this issue, we will explore and propose a novel payment rule to achieve fairness.

4 Bidder-Leximin-Optimal Payment Rule

We propose to adopt the *leximin principle* to achieve fairness in core-selecting CAs, where the leximin principle prefers the utility distribution with a larger utility for the worst-off.

Definition 2 (Leximin Dominance)**.** *Let $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ be two real vectors. $\boldsymbol{x}$ leximin dominates $\boldsymbol{y}$ if the following holds:*

- *There exists some integer $0 \leq k \leq n-1$ such that the k-smallest elements of both vectors are equal, and the $(k+1)$-smallest element of $\boldsymbol{x}$ is larger than the $(k+1)$-smallest element of $\boldsymbol{y}$.*

We denote the leximin-dominant relation by "$\succ_{Lex}$", and then a *Bidder-Leximin-Optimal* (BLO) core outcome is defined as follows:

Definition 3 (Bidder-Leximin-Optimal)**.** *Given a core U of a CA, a utility outcome $\boldsymbol{\pi} \in U$ is bidder-leximin-optimal if and only if it is not leximin dominated by any utility outcome $\boldsymbol{\pi}'$ in the core, i.e.,*

$$\nexists \boldsymbol{\pi}' \in U : \boldsymbol{\pi}' \succ_{Lex} \boldsymbol{\pi} \tag{11}$$

The BLO payment rule follows the leximin principle to give priority to those who are worst-off. It selects a core outcome that is BLO rather than a core outcome that maximizes the total utility of bidders. In order to illustrate the advantage of the BLO payment rule over the MRC payment rule, we consider the following example.

Example 2. *Let us recall the CA problem in example 1. The unique MRC utility outcome is $\pi = \{2,0,2\}$ for the winners. On the other hand, the BLO utility outcome is $\boldsymbol{\pi}' = \{1,1,1\}$ under the core constraints in Eqs.10. One can observe that $\boldsymbol{\pi}'$ is a fairer utility distribution than $\boldsymbol{\pi}$.*

4.1 Theoretical Analysis of the BLO Payment Rule

This section provides three theoretical results for the BLO payment rule. First, the following theorem establishes that the BLO outcome is unique and Pareto optimal.

Theorem 1. *The BLO outcome always exists, and it is unique and Pareto optimal, where no bidders can improve their utilities without harming other bidders' utilities.*

We then consider a lower bound of a bidder's utility in the BLO core outcome.

Theorem 2. *Assuming that bidders bid truthfully, given the BLO outcome $\{\pi_i^{BLO}\}_{i \in N}$, we have the following tight lower bound:*

$$\pi_i^{BLO} \geq \frac{1}{|W|}\pi_i^*$$

where π_i^* is the maximum utility for bidder i in the core.

The above lower bound shows that a bidder's utility is always larger than zero in the BLO outcome, unless it is zero in every core outcome's utility distribution. Regarding the total utility of the BLO outcome, it is clear that it cannot be better than that of an MRC outcome, because the total utility of an MRC outcome is maximum. The following result shows the ratio between the BLO outcome and the MRC outcome in terms of their total utility.

Theorem 3. *Assuming that bidders bid truthfully, given the BLO utility outcome $\{\pi_i^{BLO}\}_{i \in N}$, we have the following tight lower bound:*

$$\sum_{i \in N} \pi_i^{BLO} \geq \frac{4}{|W| + 2 + \frac{|W| \mod 2}{|W|}} \sum_{i \in N} \pi_i^{MRC}$$

where $\{\pi_i^{MRC}\}_{i \in N}$ is an MRC outcome.

One can observe that the relative ratio is 1 if the size of the winner set is 1 or 2, but the relative gap grows linearly with the winner set size $|W|$. Besides, the BLO outcome has a better guarantee than the general Pareto optimal outcome, which has a tight relative ratio guarantee of $\frac{1}{|W|-1}$ [2].

5 An Exact Algorithm for the BLO Payment Rule

In this section, we propose an exact algorithm, named WF-CGS-CR, to compute the BLO payment outcome. The WF-CGS-CR algorithm is a **water-filling (WF) algorithm** to increase all active winners' utilities uniformly in an interactive manner until reaching the BLO outcome, and a **constraint generation search (CGS) subroutine** is adopted to compute the maximum utility increment at each water-filling iteration. Moreover, we offer a **constraint-reuse (CR) strategy** to accelerate the CGS by setting a better initialization with the history of constraints. We will elaborate on each component of our WF-CGS-CR below.

Since core-selecting CAs involve solving the winner determination problem, existing works [Day and Raghavan, 2007; Day and Cramton, 2012] often assume that there is a so-called winner determination oracle to solve this problem automatically. We will follow such an assumption in this paper and introduce the concept as follows:

[2]More detail in the appendix.

Definition 4 ($\mathcal{WD}$)**.** *Let $\mathcal{WD}$ be an oracle with this input-output relation:*
Input: *submitted bid profile $\{b_i(\cdot)\}_{i\in N}$, where $b_i : 2^M \to \mathbb{R}$ is bidder i's reported valuation function*[3].
Output: *a winner set W and maximum social welfare $w(\{b_i(\cdot)\}_{i\in N})$ under the submitted bids.*

In other words, we do not dig into the winner determination solving but only regard it as a black-box interface. Any state-of-the-art winner determination method can be used in this way. *Oracle complexity* is defined as the query time to the oracle in the worst case, which is the primary analysis aspect for the algorithms in this paper.

5.1 Water-filling Algorithm

Water-filling algorithm was first proposed in [Niazadeh *et al.*, 2022] to compute an approximate Pareto optimal outcome. Different from the original version, we develop the algorithm for the exact BLO outcome.

Given a utility distribution in the core, we have two kinds of winners: the active winner and the frozen winner. The active winner has the potential to increase his utility, while the frozen winner can not increase the utility further. Our water-filling algorithm starts from the all-zero utility distribution and the initial active winner set W (i.e., the original winner set). Then, at the t-th iteration, it does the following:

- Run the search subroutine to return the maximum utility increment $\Delta\pi^t$ and the frozen winner set W_f^t.
- Generate the next utility distribution by increasing each active winner's utility by $\Delta\pi^t$ and generate the next active winner set by removing W_f^t from the current active winner set W_a^t.

The iteration stops when the active winner set becomes empty, i.e., no one can increase his utility anymore. The pseudo-code is shown in Alg.1.

A search subroutine is adopted to find the maximum utility increment and the frozen winner set at each iteration. Formally, given that the current utility distribution $\{\pi_i^t\}_{i\in N}$ is in the core, the optimization problem for the search subroutine can be formulated as follows:

$$\max_{\{\Delta\pi \mid \{\pi_i^t+\Delta\pi\}_{i\in W_a^t}\in U\}} \Delta\pi \tag{P0}$$

where $\{\pi_i^t+\Delta\pi\}_{i\in W_a^t}$ means increasing the utilities of winners in W_a^t by $\Delta\pi$ while the other winners' utilities remain unchanged. After the utility increment, some winners' utilities can not increase anymore, i.e., the frozen winner set at this iteration. Then we have the following lemma:

Lemma 2. *Given that the results computed by the search subroutine are correct, the water-filling algorithm returns the BLO outcome.*

5.2 Constraint Generation Search Subroutine

A naive method to solve the problem (P0) is obtaining all the core constraints, but it needs to query the winner determination oracle exponential times according to Eqs.8. To avoid this, in this section, we first formulate the problem (P0) to an equivalent mathematical format and then solve it through the constraint generation search.

[3]We regard the bidders' reported valuation functions as their true valuation functions in the computation part.

Algorithm 1 Water-filling algorithm for finding the exact BLO outcome

Input: bid profile $\{b_i(\cdot)\}_{i\in N}$.
Output: the BLO outcome.

Step $t \leftarrow 1$
$W, w(N) \leftarrow \mathcal{WD}(\{b_i(\cdot)\}_{i\in N})$
Initialize the utility distribution $\{\pi_i^t\}_{i\in N}$ as all-zero
Initialize the active winner set W_a^t as W
while W_a^t is not empty **do**
  Run the search subroutine to return the maximum utility increment $\Delta\pi^t$ and the frozen winner set W_f^t
  Generate the next utility distribution:
$$\pi_i^{t+1} = \begin{cases} \pi_i^t + \Delta\pi^t, \text{ if } i \in W_a^t \\ \pi_i^t, \text{ otherwise} \end{cases}$$
  Generate the next active winner set:
$$W_a^{t+1} = W_a^t \backslash W_f^t$$
  Step $t \leftarrow t+1$
return $\{\pi_i^t\}_{i\in N}$

Optimization Problem Formulation

In the original format (P0), the restriction condition is to ensure the utility distribution is still in the core, in other words, satisfying all the core constraints. According to Eqs.8, the constraints can be written as:

$$\sum_{i\in N\backslash S} \pi_i^t + |W_a^t \setminus S|\Delta\pi \le w(N) - w(S),\ \forall S \subseteq N \tag{12}$$

If $W_a^t \subseteq S$, then $|W_a^t \setminus S| = 0$, and the corresponding constraints can be ignored. Thus, we can arrange the remaining constraints as follows:

$$\Delta\pi \le \frac{w(N) - w(S) - \sum_{i\in N\backslash S}\pi_i^t}{|W_a^t\backslash S|} \tag{13}$$

where $S \subseteq N$ *and* $W_a^t \not\subseteq S$. Every such constraint provides an upper bound for $\Delta\pi$. Denote by $\Delta\pi^S$ the upper bound corresponding to S, i.e., RHS in Eq.13. Thus, finding the maximum utility increment satisfying all the core constraints is equivalent to finding the minimum upper bound above. Problem (P0) is equivalent to the following optimization problem:

$$\min_{\{S \mid S\subseteq N, W_a^t \not\subseteq S\}} \frac{w(N) - w(S) - \sum_{i\in N\backslash S}\pi_i^t}{|W_a^t\backslash S|} \tag{P1}$$

Let S^* be one of the optimal coalitions for problem (P1), thus $\Delta\pi^t = \Delta\pi^{S^*}$. After the increment, the corresponding core constraint of S^* would be tight for the new utility distribution $\{\pi_i^{t+1}\}_{i\in N}$, thus we have

$$\sum_{i\in N\backslash S^*} \pi_i^{t+1} = w(N) - w(S^*) \tag{14}$$

According to Eq.14, bidders belonging to $N\backslash S^*$ cannot increase their utilities without violating this constraint, i.e., they are frozen. Therefore, the frozen winner set W_f^t is $W_a^t\backslash S^*$ at this iteration. Problem (P1) is actually a nesting optimization problem with a fractional format since $w(S)$ is corresponding to the winner determination problem. Thus it is intractable to solve directly, and we propose to use the constraint generation method that only considers the most useful constraints.

Constraint Generation Search

Generally speaking, CGS starts from an upper bound of the utility increment, then reduces the upper bound iteratively through generated constraints until it is the minimum.

Formally, CGS initializes the upper bound $\Delta\bar{\pi}$ as ∞ and the frozen winner set as W_a^t. It checks whether this upper bound is the minimum by the following equation:

$$w(N) - \sum_{i\in N} \tilde{\pi}_i = w_B \tag{15}$$

where $\{\tilde{\pi}_i\}_{i\in N}$ is the utility distribution by setting the utility increment as $\Delta\bar{\pi}$ and w_B is the maximum social welfare under the truncated bid profile $\{\max(b_i(\cdot) - \tilde{\pi}_i, 0)\}_{i\in N}$. If Eq.15 is satisfied, the current upper bound $\Delta\bar{\pi}$ is the minimum, and CGS will return the result. Otherwise, a smaller upper bound is computed by

$$\Delta\bar{\pi} = \Delta\bar{\pi} - \frac{w_B - (w(N) - \sum_{i\in N}\tilde{\pi}_i)}{|\bar{W}_f|} \tag{16}$$

The corresponding frozen winner set is updated by

$$\bar{W}_f = W_a^t\backslash B \tag{17}$$

where B is the winner set under the truncated bid profile $\{\max(b_i(\cdot) - \tilde{\pi}_i, 0)\}_{i\in N}$. If the new frozen winner set satisfies $|\bar{W}_f| = 1$, then this upper bound is the minimum. Otherwise, CGS will start a new iteration with the smaller upper bound. The oracle complexity of CGS is given as follows:

Lemma 3. *Constraint generation search requires at most $|W_a^t|$ queries to the oracle $\mathcal{WD}$ to solve the optimization problem (P1).*

Assume that the iteration number of the water-filling algorithm is T. Naturally, since $W = W_a^1 \supset ... \supset W_a^T = \emptyset$, the query time of the CGS subroutine in the water-filling algorithm is at most $|W| + (|W|-1) + ... + 1 = \frac{|W|(|W|+1)}{2}$.

5.3 Constraint-reuse Strategy

To avoid CGS starting from the naive upper bound ∞, we propose the constraint-reuse strategy to set a better initialization through the history of constraints.

Formally, we store the binary tuple $(W_f^S, \Delta\pi^S)$ in the stored constraint set $\mathbf{C}$, where W_f^S is the frozen winner set if S is an optimal coalition, i.e., $W_f^S = W_a^t \setminus S$. Then in each CGS subroutine, we first update each binary tuple $\mathbf{C}$ based on the new utility distribution and active winner set. The recurrence formula is given as follows:

$$\begin{cases} \acute{W}_f^S = W_f^S \cap W_a^t \\ \Delta\acute{\pi}^S = \dfrac{|W_f^S|}{|\acute{W}_f^S|}(\Delta\pi^S - \Delta\pi^{t-1}) \end{cases} \tag{18}$$

Algorithm 2 Constraint generation search with the constraint-reuse strategy

Input: bid profile $\{b_i(\cdot)\}_{i\in N}$, maximum social welfare $w(N)$, current utility distribution $\{\pi_i^t\}_{i\in N}$, active winner set W_a^t, last utility increment $\Delta\pi^{t-1}$, constraint set $\mathbf{C}$.

Output: the maximum utility increment $\Delta\pi^t$ and the frozen winner set W_f^t.

/*Constraint-reuse Strategy*/

Initialize $\Delta\bar{\pi} \leftarrow \infty$; $\bar{W}_f \leftarrow W_a^t$

for $(W_f^S, \Delta\pi^S)$ in $\mathbf{C}$ **do**

  $\mathbf{C} \leftarrow \mathbf{C} \setminus \{(W_f^S, \Delta\pi^S)\}$

  if $W_f^S \cap W_a^t \neq \emptyset$ **then**

    $\acute{W}_f^S \leftarrow W_f^S \cap W_a^t$; $\Delta\acute{\pi}^S = \frac{|W_f^S|}{|\acute{W}_f^S|}(\Delta\pi^S - \Delta\pi^{t-1})$

    if $\Delta\acute{\pi}^S < \Delta\bar{\pi}$ **then** $\Delta\bar{\pi} \leftarrow \Delta\acute{\pi}^S$; $\bar{W}_f \leftarrow \acute{W}_f^S$

    $\mathbf{C} \leftarrow \mathbf{C} \cup \{(\acute{W}_f^S, \Delta\acute{\pi}^S)\}$

/*Constraint Generation Search*/

while True **do**

  Generate the utility distribution by upper bound $\Delta\bar{\pi}$:

$$\tilde{\pi}_i = \begin{cases} \pi_i^t + \Delta\bar{\pi}, \text{ if } i \in W_a^t \\ \pi_i^t, \text{ otherwise} \end{cases}$$

  $B, w_B \leftarrow \mathcal{WD}(\{\max(b_i(\cdot) - \tilde{\pi}_i, 0)\}_{i\in N})$

  if $w(N) - \sum_{i\in N}\tilde{\pi}_i = w_B$ **then** break

  $\Delta\bar{\pi} \leftarrow \Delta\bar{\pi} - \frac{w_B - (w(N) - \sum_{i\in N}\tilde{\pi}_i)}{|\bar{W}_f|}$; $\bar{W}_f \leftarrow W_a^t\backslash B$

  $\mathbf{C} \leftarrow \mathbf{C} \cup \{(\bar{W}_f, \Delta\bar{\pi})\}$

  if $|\bar{W}_f| = 1$ **then** break

return $\Delta\bar{\pi}, \bar{W}_f$

Above all, we have introduced the three components of the WF-CGS-CR algorithm and the runtime guarantee is shown as follows:

Theorem 4. *The WF-CGS-CR algorithm requires at most $\frac{|W|(|W|+1)}{2} + 1$ queries to oracle $\mathcal{WD}$ and an additional time complexity $O(|W|^5)$ to obtain the BLO outcome, where $|W|$ is the winner size.*

By adopting this formula, we transfer each binary tuple $(W_f^S, \Delta\pi^S)$ to a new binary tuple $(\acute{W}_f^S, \Delta\acute{\pi}^S)$ before CGS. Eventually, the minimum upper bound in the constraint set is computed and serves as the initial upper bound for CGS. Besides, each new binary tuple is stored in the constraint set if $\acute{W}_f^S$ is not empty. The complete pseudo-code for the CGS subroutine is shown in Alg.2. Note that this strategy would increase some extra computation complexity that is minor since the winner determination oracle is the main time-consuming part.

6 Experiments

In the experiments, we use the CATS (Combinatorial Auction Test Suite) [Leyton-Brown *et al.*, 2000] as the CA instance generator. Same as [Bünz *et al.*, 2015], we choose six representative CATS distributions: Arbitrary, Decay(L4),

Algorithm	Arbitrary	Decay(L4)	Matching	Paths	Regions	Scheduling
VCG	12.49 (4.52)	1.83 (0.87)	**0.75** (0.03)	**2.50** (0.18)	2.35 (0.45)	**0.83** (0.36)
MRC	22.09 (10.30)	4.84 (2.35)	2.12 (0.62)	5.70 (1.54)	4.42 (1.40)	2.37 (1.93)
MRC-VCG	23.32 (10.86)	4.84 (2.21)	2.23 (0.63)	5.86 (1.51)	4.76 (1.48)	2.90 (2.49)
MRC-Zero	23.48 (10.91)	5.17 (2.74)	2.23 (0.64)	5.90 (1.55)	4.87 (1.47)	2.99 (2.79)
Fast Core	33.26 (15.24)	6.80 (3.19)	7.67 (1.00)	21.46 (3.07)	7.23 (2.54)	11.48 (8.9)
WF-CGS	9.67 (4.85)	3.57 (1.83)	2.98 (0.42)	10.37 (1.56)	2.37 (1.13)	2.36 (0.97)
WF-CGS-CR	**7.15** (3.64)	**1.81** (0.93)	1.22 (0.18)	3.25 (0.36)	**1.55** (0.54)	1.58 (1.17)

Table 3: Results for the run time. Average run times (in seconds) over 50 CA instances are provided, with standard error in the parentheses. Each CA instance is generated by CATS, with 64 goods and 1000 bids. The lowest average run time in each column is marked in **bold**.

Matching, Paths, Regions, and Scheduling. For each distribution, we generate 50 CA instances with 64 goods and 1000 bids. All the experiments were run on a high-performance computer with a 3.10GHz Intel core and 16GB of RAM. Besides, the winner determination problem is transformed into a format of integer programming, solved by the solver CPLEX 20.1.0 [Manual, 1987]. The following payment rules are implemented for the comparison:

- *VCG*: the VCG payment rule, where each winner's utility is $\pi_i^{VCG} = w(N) - w(N \setminus \{i\})$.
- *MRC*: the MRC payment rule that selects one core outcome with the maximum total utility of bidders [Day and Raghavan, 2007].
- *MRC-VCG*: a quadratic payment rule that selects the MRC point nearest to the VCG point in l_2-distance [Day and Cramton, 2012].
- *MRC-Zero*: a quadratic payment rule that selects the MRC point nearest to the all-zero point in l_2-distance [Day and Cramton, 2012].
- *Fast Core*: the approximate BLO payment rule implemented through the water-filling algorithm with the binary search [Niazadeh *et al.*, 2022].

We use the state-of-the-art Core Constraint Generation (CCG) algorithm to compute all the MRC outcomes above [Day and Raghavan, 2007]. Unfortunately, it has no theoretical guarantee for the query time; in other words, its oracle complexity may be exponential. Instead, the oracle complexity of Fast Core is $O(|W| \log \frac{|W|}{\epsilon})$, where ϵ is the parameter representing the approximate gap. Same as [Niazadeh *et al.*, 2022], ϵ is set as 0.01 in our experiments.

6.1 Experimental Results

The auction results are shown in Table 2. We can see that core-selecting payment rules enhance the seller's revenue compared to the VCG payment rule. Among the core-selecting payment rules, the BLO rule achieves a fairer utility distribution with a tiny reduction of the total utility (i.e., from 173.65 to 147.17). Moreover, the MRC rules produce an unfair utility distribution, with nearly 30% winners obtaining zero utility, which impairs the incentives for bidders. Instead, the BLO rule has only 1.69% zero-utility winners on average, thus providing a guarantee for the worst-off. Fast Core achieves approximate fairness, but there are still 12.71% zero-utility winners.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	11661.54	513.04	5.03	13.63	1.69%
MRC-VCG	12000.93	173.65	0.01	10.36	31.63%
MRC-Zero	12000.93	173.65	0.05	9.54	28.57%
Fast Core	12046.95	127.63	0.83	5.94	12.71%
BLO	12027.41	147.17	1.18	6.77	1.69%

Table 2: Results for the auction. The measurements include the revenue, the total utility, the minimum utility (Min), the standard deviation (Std), and the percentage of the zero-utility winners among all winners (Zero ratio). All results are averages over 300 CA instances, with 50 instances for each distribution. Note that ↑ indicates larger values are better, and ↓ indicates smaller values are better in terms of fairness.

The average run time results are shown in Table 3, where WF-CGS represents our algorithm without the constraint-reuse strategy. Note that we include the run time for computing VCG prices for the MRC rules since it serves as the initial point in the CCG algorithm [Day and Raghavan, 2007]. As we can see, the WF-CGS-CR algorithm performs the best among all the core-selecting algorithms, which is even close to the VCG algorithm. Moreover, the constraint-reuse strategy is effective in our experiments, especially in the distribution of Paths.

7 Conclusion

In this paper, we propose the BLO payment rule to address the unfair issue with the MRC rule. The BLO payment rule adopts the leximin principle to select the core outcome, prioritizing the worst-off bidders. Furthermore, we theoretically analyze the BLO rule and propose the WF-CGS-CR algorithm to compute the exact BLO outcome. Our algorithm has the quadratic oracle complexity and achieves the best run time in the experiments, even close to VCG.

Compared with the MRC rule, the BLO rule has the following advantages: (1) the BLO outcome is unique, avoiding the problem of core outcome selection; (2) the BLO outcome achieves leximin fairness and thus avoids the unfair phenomenon caused by the MRC rule; (3) given the winner determination oracle access, the BLO outcome is computed effectively in time polynomial in the winner size. Therefore, the BLO rule could serve as an alternative payment rule for the core-selecting CAs.

Acknowledgements

This research was supported by the National Natural Science Foundation of China (Grant No. 62192783, U1811462), the Collaborative Innovation Center of Novel Software Technology and Industrialization at Nanjing University. The work of Shufeng Kong was partially supported by the "Hundred Talents Program" of Sun Yat-sen University. The work of Caihua Liu was partially supported and funded by the Humanities and Social Sciences Youth Foundation, Ministry of Education of the People's Republic of China (Grant No.21YJC870009). Bo An is supported by the National Research Foundation, Singapore under its Industry Alignment Fund – Pre-positioning (IAF-PP) Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

References

[Arin and Inarra, 2001] Javier Arin and Elena Inarra. Egalitarian solutions in the core. *International Journal of Game Theory*, 30(2):187–193, 2001.

[Ausubel and Baranov, 2017] Lawrence M. Ausubel and Oleg Baranov. A practical guide to the combinatorial clock auction. *The Economic Journal*, 127(605):334–350, 2017.

[Ausubel *et al.*, 2006] Lawrence M Ausubel, Paul Milgrom, et al. The lovely but lonely vickrey auction. *Combinatorial Auctions*, 17:22–26, 2006.

[Bünz *et al.*, 2015] Benedikt Bünz, Sven Seuken, and Benjamin Lubin. A faster core constraint generation algorithm for combinatorial auctions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 827–834, 2015.

[Bünz *et al.*, 2018] Benedikt Bünz, Benjamin Lubin, and Sven Seuken. Designing core-selecting payment rules: A computational search approach. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 109–109, 2018.

[Clarke, 1971] Edward H Clarke. Multipart pricing of public goods. *Public Choice*, pages 17–33, 1971.

[Cramton, 2013] Peter Cramton. Spectrum auction design. *Review of Industrial Organization*, 42(2):161–190, 2013.

[Day and Cramton, 2012] Robert W Day and Peter Cramton. Quadratic core-selecting payment rules for combinatorial auctions. *Operations Research*, 60(3):588–603, 2012.

[Day and Milgrom, 2008] Robert Day and Paul Milgrom. Core-selecting package auctions. *International Journal of Game Theory*, 36(3):393–407, 2008.

[Day and Raghavan, 2007] Robert W Day and Subramanian Raghavan. Fair payments for efficient allocations in public sector combinatorial auctions. *Management Science*, 53(9):1389–1406, 2007.

[Erdil and Klemperer, 2010] Aytek Erdil and Paul Klemperer. A new payment rule for core-selecting package auctions. *Journal of the European Economic Association*, 8(2-3):537–547, 2010.

[Gillies, 1959] Donald B Gillies. Solutions to general non-zero-sum games. *Contributions to the Theory of Games*, 4:47–85, 1959.

[Goel *et al.*, 2015] Gagan Goel, Mohammad Reza Khani, and Renato Paes Leme. Core-competitive auctions. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 149–166, 2015.

[Goeree and Lien, 2016] Jacob K Goeree and Yuanchuan Lien. On the impossibility of core-selecting auctions. *Theoretical Economics*, 11(1):41–52, 2016.

[Grötschel *et al.*, 1981] Martin Grötschel, László Lovász, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981.

[Groves, 1973] Theodore Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.

[Leyton-Brown *et al.*, 2000] Kevin Leyton-Brown, Mark Pearson, and Yoav Shoham. Towards a universal test suite for combinatorial auction algorithms. In *Proceedings of the 2nd ACM Conference on Electronic Commerce*, pages 66–76, 2000.

[Leyton-Brown *et al.*, 2017] Kevin Leyton-Brown, Paul Milgrom, and Ilya Segal. Economics and computer science of a radio spectrum reallocation. *Proceedings of the National Academy of Sciences*, 114(28):7202–7209, 2017.

[Lubin *et al.*, 2015] Benjamin Lubin, Benedikt Bünz, and Sven Seuken. New core-selecting payment rules with better fairness and incentive properties. In *The Third Conference on Auctions, Market Mechanisms and Their Applications*. ACM, 2015.

[Manual, 1987] CPLEX User's Manual. Ibm ilog cplex optimization studio. *Version*, 12(1987-2018):1, 1987.

[Niazadeh *et al.*, 2022] Rad Niazadeh, Jason Hartline, Nicole Immorlica, Mohammad Reza Khani, and Brendan Lucier. Fast core pricing for rich advertising auctions. *Operations Research*, 70(1):223–240, 2022.

[Rawls, 2004] John Rawls. A theory of justice. In *Ethics*, pages 229–234. Routledge, 2004.

[Rothkopf *et al.*, 1998] Michael H Rothkopf, Aleksandar Pekeč, and Ronald M Harstad. Computationally manageable combinational auctions. *Management science*, 44(8):1131–1147, 1998.

[Sandholm, 2007] Tuomas Sandholm. Expressive commerce and its application to sourcing: How we conducted $35 billion of generalized combinatorial auctions. *AI Magazine*, 28(3):45–45, 2007.

[Vickrey, 1961] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961.

Appendix

A Proof for Lemma 1

Proof for Lemma 1. We extend Example 1 to a general case, and the bid profile is given as follows:

$$\begin{cases} bidder\ i : \{I_i\} = 2 \quad \forall i \in [1, |W|] \quad \textbf{(winners)} \\ bidder\ |W|+1 : \{I_2, I_3, ..., I_{|W|}\} = |W|-1 \\ bidder\ |W|+2 : \{I_1, I_3, ..., I_{|W|}\} = |W|-1 \end{cases} \tag{19}$$

where $I_1, I_2, ..., I_{|W|}$ are the item bundles of the winners. There are $|W|+2$ bidders in this instance, and the winner size is $|W|$. The core in this instance is given as follows:

$$Core = \begin{cases} \pi_1 \geq 0, \pi_2 \geq 0, ..., \pi_{|W|} \geq 0 \\ \pi_1 + \pi_3 +, ..., +\pi_{|W|} \leq |W|+1 \\ \pi_2 + \pi_3 +, ..., +\pi_{|W|} \leq |W|+1 \end{cases} \tag{20}$$

In this example, the unique MRC outcome is $\pi_1 = \pi_2 = |W|+1$ and $\pi_i = 0, \forall i > 2$, where bidder $3, 4, ..., |W|$ get a zero utility though they are winners. However, they can get a positive utility in some core outcome (e.g., $\pi_i = 1, \forall i \in [1, |W|]$). In this case, $|W|-2$ winners are forced to get a zero utility even though they could have obtained a positive utility in the core. □

B Proof for Theorem 1

Proof for Theorem 1. **Existence**: The pay-as-bid point (i.e., all-zero point) is always in the core since it satisfies all the core constraints in Eqs.7 and Eqs.8. Therefore, the core is a nonempty compact convex set. Thus, the BLO core outcome always exists due to the following proposition in [Radunovic and Le Boudec, 2007]:

Proposition 1. *Any compact subset of $\mathbb{R}^n$ has a leximin maximal vector.*

Uniqueness: Assume that the BLO outcome is not unique, and let $\{\pi_i\}_{i\in N}$ and $\{\pi'_i\}_{i\in N}$ be two BLO outcomes. Thus, they are different vectors with the same leximin sequence. let $X = \{i | \pi_i \neq \pi'_i\}$ and $k = \arg\min_{i\in X} \min(\pi_i, \pi'_i)$. Thus, bidder k is the bidder with minimum utility among all the bidders with the different utilities in these two outcomes. Without loss of generality, assuming that $\pi_k < \pi'_k$, then we analyze the utilities of bidders from the three cases:

Case 1: bidder i satisfying $\pi_i < \pi_k$. According to the definition, we have $\pi_i = \pi'_i$ for these bidders.

Case 2: bidder i satisfying $\pi_i = \pi_k$. We have $\pi_i \leq \pi'_i$; otherwise, bidder k is not the bidder with the minimum utility in X.

Case 3: bidder i satisfying $\pi_i > \pi_k$. There is no relationship between π_i and π'_i.

Then we construct a new outcome $\{(1-\epsilon)\pi_i + \epsilon\pi'_i\}_{i\in N}$, where $0 < \epsilon < \min(\frac{1}{2}, \min_{\{i|\pi_i>\pi_k, \pi_i>\pi'_i\}} \frac{\pi_i - \pi_k}{\pi_i - \pi'_i})$. This outcome is in the core due to the convexity. Thus, for bidder i in case 1 and case 2, we have $(1-\epsilon)\pi_i + \epsilon\pi'_i \geq \pi_i$. For bidder i in case 3, we have:

$$(1-\epsilon)\pi_i + \epsilon\pi'_i = \pi_k + (\pi_i - \pi_k) - \epsilon(\pi_i - \pi'_i) > \pi_k \tag{21}$$

Assuming that the index of bidder k in the leximin order of $\{\pi_i\}_{i\in N}$ is $\hat{k}$ (note that we let $\hat{k}$ be the latest index for the bidder whose utility is π_k), then we have that $\hat{k}$-largest utility in this new outcome is larger than π_k since there are at least $N - \hat{k} + 1$ bidders whose utility is larger than π_k (this is because $(1-\epsilon)\pi_k + \epsilon\pi'_k > \pi_k$). Moreover, we have $(1-\epsilon)\pi_i + \epsilon\pi'_i \geq \pi_i$ for the bidder whose utility is not larger than π_k and $(1-\epsilon)\pi_i + \epsilon\pi'_i > \pi_k$ for the bidder whose utility is larger than π_k. Thus, the leximin utility vector before the index $\hat{k}$ in the new outcome is not leximin dominated by that $\{\pi_i\}_{i\in N}$. Therefore, according to the definition of leximin dominance, we have $\{(1-\epsilon)\pi_i + \epsilon\pi'_i\}_{i\in N} \succ_{Lex} \{\pi_i\}_{i\in N}$. This generates a contradiction; thus, the BLO outcome is unique.

Pareto optimality: According to the water-filling algorithm, each winner is bounded by a tight core constraint in the final outcome. Thus, each winner can not improve his utility individually without breaking this core constraint. Therefore, the BLO outcome is Pareto optimal. □

C Proof for Theorem 2

First, we have the following lemma:

Lemma 4. *For each bidder, the maximum utility in the core is his VCG utility.*

Proof. The VCG utility of bidder i is:

$$\pi_i^{VCG} = w(N) - w(N \backslash \{i\}) \tag{22}$$

Then we consider the outcome where bidder i obtains his VCG utility while the other bidder's utility is zero. For the core constraints in Eqs.8, we have the following cases:

Case 1: $i \notin S$. We have:

$$\sum_{i\in N\backslash S} \pi_i = w(N) - w(N\backslash\{i\}) \leq w(N) - w(S) \tag{23}$$

Case 2: $i \in S$. We have:

$$\sum_{i\in N\backslash S} \pi_i = 0 \leq w(N) - w(S) \tag{24}$$

Thus, this outcome is in the core. Let $S = N\backslash\{i\}$ in Eqs.8, then we have:

$$\pi_i \leq w(N) - w(N\backslash\{i\}) = \pi_i^{VCG} \tag{25}$$

Therefore, each bidder's maximum utility in the core is his VCG utility. □

Proof for Theorem 2 . Assume that the iteration number of the water-filling algorithm is T. We consider the final BLO outcome is $\{\pi_i^T\}_{i\in W}$ for the winners. For winner i, his utility is frozen due to a tight core constraint. Without loss of generation, the core constraint is:

$$\sum_{i\in N\backslash B^t} \pi_i^T = w(N) - w(B^t) \tag{26}$$

where $B \subseteq N$ is the coalition for the corresponding core constraint. We can see that the utilities of winners in $W \backslash B^t$ are frozen before or at the t-th iteration. Thus, the utility of such bidder is no more than π_i^T. Then we have:

$$|W \backslash B^t| \pi_i^T \geq w(N) - w(B^t) \tag{27}$$

Since $i \notin B^t$, thus we have:

$$\begin{aligned} \pi_i^T &\geq \frac{w(N) - w(B^t)}{|W \backslash B^t|} \\ &\geq \frac{w(N) - w(N \backslash \{i\})}{|W|} \\ &= \frac{\pi_i^{VCG}}{|W|} \end{aligned} \tag{28}$$

Therefore, theorem 2 is proved. □

D Proof for Theorem 3

First, we introduce the skeleton for the proof. Denote by $\{\pi_i^{MRC}\}_{i \in W}$ the MRC outcome and $\{\pi_i^{BLO}\}_{i \in w}$ the BLO outcome. Denote by $\mathcal{CA}$ the total combinatorial auction instance set, and for any CA instance $ca \in \mathcal{CA}$, let $r^{ca} = \frac{\sum_{i \in W} \pi^{MRC}}{\sum_{i \in W} \pi^{BLO}}$ be the ratio of total utility between the MRC outcome and the BLO outcome. Note that we only consider the utilities of the winners since the non-winner utilities are zero.

We introduce three operations OP_1, OP_2, OP_3 to operate on the instance ca in sequence and generate a final instance to derive the last ratio. The main process is as follows:

$$ca \xrightarrow{OP_1} ca^1 \xrightarrow{OP_2} ca^2 \xrightarrow{OP_3} ca^3$$

where ca^1, ca^2, ca^3 are the CA instances after the operation OP_1, OP_2, OP_3 and the winner size $|W|$ remains the same for these four instances. Then our goal is to prove the following inequalities:

$$r^{ca} \leq r^{ca^1} \leq r^{ca^2} \leq r^{ca^3} \leq \frac{|W| + 2 + \frac{|W| \mod 2}{|W|}}{4} \tag{29}$$

In the following sections, we will introduce these three operations and prove the inequalities in Eqs.29.

D.1 Operation OP_1

In this section, we introduce the operation OP_1 and prove that $r^{ca} \leq r^{ca^1}$.

For CA instance ca, we get the unique BLO outcome as $\{\pi_i^{BLO}\}_{i \in W}$ by the water-filling algorithm. We resort the winners in ascending order of utility as follows:

$$\alpha_1, ..., \beta_1, \alpha_2, ..., \beta_2, ..., \alpha_T, ..., \beta_T$$

where $W_f^t = \{\alpha_t, ..., \beta_t\}$ and $\alpha_1 = 1, \beta_T = |W|$. Denote by Π^t the amount of utility for winner $i \in [\alpha_t, \beta_t]$. According to the water-filling algorithm, Π^t corresponds to a boundary core constraint as follows:

$$C^t : \sum_{i \in W \backslash B^t} \pi_i \leq w(N) - w(B^t), \ \forall t \in [1, T] \tag{30}$$

where these inequalities take the equal sign in the BLO outcome. Let $L^t = W \backslash B^t$, and L^t does not include any winner that belongs to $W_f^j, \forall j > t$ and $W_f^t \subseteq L^t$. Thus, we can rewrite these constraints as follows:

$$\sum_{j=1}^{t-1} \sum_{i \in X_{tj}} \pi_i + \sum_{i \in W_f^t} \pi_i \leq |W_f^t| \Pi^t + \sum_{j=1}^{t-1} |X_{t,j}| \Pi^j, \forall t \in [1, T] \tag{31}$$

where $X_{t,j}$ represents the intersection of L^t and W_f^j, i.e., $X_{t,j} = L^t \cap W_f^j$. Let R^t represent the RHS in Eqs.31. Then we have:

$$\begin{cases} L^t = X_{t,1} \cup X_{t,2} ... \cup X_{t,t-1} \cup W_f^t \\ R^t = |W_f^t| \Pi^t + \sum\limits_{j \in \{1,...,t-1\}} |X_{t,j}| \Pi^j \end{cases} \tag{32}$$

Thus, for the instance ca, there are at least T core constraints as follows:

$$C^t : \sum_{i \in L^t} \pi_i \leq R^t, \ t \in [1, T] \tag{33}$$

Then we can define the operation OP_1 as follows:

Definition 5 (Operation OP_1)**.** *Given any instance $ca \in \mathcal{CA}$ and its core constraints for the BLO rule are $C^1, ..., C^T$ as above. Operation OP_1 constructs a new instance ca^1 with the submitted bid profile as follows:*

$$\begin{cases} \textit{bidder } i : \{I_i\} = \sum\limits_{t \in [1,T]} R^t, \ \forall i \in [1, |W|] \ \textbf{\textit{(winners)}} \\ \textit{bidder } i + t : \cup_{i \in L^t} I_i = \sum\limits_{i \in L^t} b_i(I_i) - R^t, \ \forall t \in [1, T] \end{cases} \tag{34}$$

where $I_1, I_2, ..., I_{|W|}$ are the item bundles of the winners and L^t, R^t are defined in Eqs.32.

In this instance, the winners are $1, 2, ..., |W|$. Since the non-winner utilities are zero in the core, thus we can rewrite the core constraints in Eqs.8 as follows:

$$\sum_{i \in W \backslash S} \pi_i \leq w(N) - w(S), \ \forall S \subseteq N \tag{35}$$

Let $S = (W \backslash L^t) \cup \{|W| + t\}$, and we have:

$$\sum_{i \in L^t} \pi_i \leq w(N) - (\sum_{i \in W \backslash L^t} b_i(I_i) + \sum_{i \in L^t} b_i(I_i) - R^t) = R^t \tag{36}$$

This inequality is the same as C^t. With the core constraints in Eqs.7, the core of instance ca^1 is given as follows:

$$Core = \begin{cases} \sum\limits_{i \in L^t} \pi_i \leq R^t, \ \forall t \in [1, T] \\ \pi_i \geq 0, \forall i \in [1, |W|] \end{cases} \tag{37}$$

To verify the correctness of the core, we assume that $\{\hat{\pi}_i\}_{i \in W}$ is a point in the above core. Then we consider the truncated bid profile $\{b_i(I_i) - \hat{\pi}_i\}_{i \in W}$.

First, for any winner $i \in L^t$, we have

$$\hat{\pi}_i \le R^t - \sum_{i \in L^t} \hat{\pi}_i \le R^t \le \sum_{t \in [1,T]} R^t = b_i(I_i) \tag{38}$$

This means that $b_i(I_i) - \hat{\pi}_i \ge 0$, thus the above bid profile is reasonable. Then we can rewrite the constraint C^t as follows:

$$\begin{aligned} \sum_{i \in L^t} (b_i(I_i) - \hat{\pi}_i) &\ge \sum_{i \in L^t} b_i(I_i) - R^t \\ &= b_{|W|+t}(\cup_{i \in L^t} I_i) \end{aligned} \tag{39}$$

According to this inequality, bidder $|W|+t$ cannot be a block coalition for this point since the truncated bid price of L^t is not less than $|W|+t$ for the same item bundle. This inequality is true for all $t = 1, 2, ..., T$. No potential block coalition exists, so the point is in the core. Therefore, Eqs.37 describes the core of ca^1 correctly. Due to the same constraints $C^1, C^2, ..., C^t$, the BLO outcome of ca^1 is the same as that of ca. Then, the total utility of the MRC outcome of ca^1 is computed by the following linear programming:

$$\begin{aligned} \sum_{i \in W} \pi_i^{MRC} = \max \sum_{i \in W} \pi_i \\ \text{s.t. } \; C^t \;\; \forall t = 1, 2, ..., T \\ \pi_i \ge 0 \;\; \forall i \in W \end{aligned} \tag{40}$$

We know that the constraint set of the instance ca includes all the above constraints. Thus, the core of ca is just a subset of the core of ca^1 so that the MRC outcome's total utility in ca is no more than that in ca^1. Then we can conclude that $r^{ca} \le r^{ca^1}$

D.2 Operation OP_2

In this section, we introduce the operation OP_2 and prove that $r^{ca^1} \le r^{ca^2}$. We begin with the definition of the operation OP_2.

Definition 6 (Operation OP_2)**.** *Given the instance ca^1, operation OP_2 is an operation of T steps. Let $X'_{i,j} = X_{i,j}, \; \forall i, j : 1 < j < i \le T$. For $t = T, T-1, ..., 1$, it does the followings:*

Check whether exists $k > t$ satisfying that $W_f^t = X'_{k,t}$:

Case 1: exist. Let $X'_{t,j} = \emptyset$ for all $j < t$. Add t into the set T^1.

Case 2: not existing. Let $X'_{j,t} = \{\alpha_t, ..., \alpha_t + |X_{j,t}| - 1\}$ for allj $> t$. Add t into the set T^2.

Then the bid profile of instance ca^2 is constructed as follows:

$$\begin{cases} \textit{bidder } i : \{I_i\} = \displaystyle\sum_{t \in [1,T]} R^t, \; \forall i \in [1, |W|] \; \textbf{\textit{(winners)}} \\ \textit{bidder } |W| + t : \cup_{i \in L^t} I_i = \displaystyle\sum_{i \in L^t} b_i(I_i) - R^t, \forall t \in [1, T] \end{cases} \tag{41}$$

where $I_1, ..., I_{|W|}$ are the bid item bundles of the winners and L^t, R^t are defined as follows:

$$\begin{cases} L^t = X'_{t,1} \cup X'_{t,2} ... \cup X'_{t,t-1} \cup W_f^t \\ R^t = \displaystyle\sum_{j=1}^{t-1} |X'_{t,j}| \Pi^j + |W_f^t| \Pi^t \end{cases} \tag{42}$$

For any $t \in T^1$, since $X'_{t,j} = \emptyset, \forall j < t$, we have:

$$\begin{cases} L^t = W_f^t, \forall t \in T^1 \\ R^t = |W_f^t| \Pi^t, \forall t \in T^1 \end{cases} \tag{43}$$

For any $t \in T^2$, we have:

$$\begin{cases} L^t = X'_{t,1} \cup X'_{t,2} ... \cup X'_{t,t-1} \cup W_f^t \\ R^t = |W_f^t| \Pi^t + \displaystyle\sum_{j \in \{1,...,t-1\}} |X'_{t,j}| \Pi^j \end{cases} \tag{44}$$

Thus, the core of ca^2 can be written as

$$\begin{cases} \displaystyle\sum_{i \in W_f^t} \pi_i \le |W_f^t| \Pi^t, \; \forall t \in T^1 \\ \displaystyle\sum_{j=1}^{t-1} \sum_{i \in X'_{t,j}} \pi_i + \sum_{i \in W_f^t} \pi_i \le \sum_{j=1}^{t-1} |X'_{t,j}| \Pi^j + |W_f^t| \Pi^t, \; \forall t \in T^2 \\ \pi_i \ge 0, \; \forall i \in W \end{cases} \tag{45}$$

According to the core, the water-filling process of ca^2 is the same as that of ca^1. Thus, the BLO outcome is the same in the instance ca^1 and ca^2. To prove that $r^{ca^1} \le r^{ca^2}$, we need to prove the following lemma:

Lemma 5. *After the operation OP_2, the total utility of the MRC outcome does not decrease.*

Proof. In this proof, we provide an MRC outcome $\{\pi'_i\}_{i \in W}$ for the instance ca^2 and then prove the total utility of this outcome is an upper bound of total utility in instance ca^1.

First we give the format of $\{\pi'_i\}_{i \in W}$ as follows:

$$\pi'_i = \begin{cases} \displaystyle\sum_{j=1}^{t-1} |X'_{t,j}| \Pi^j + |W_f^t| \Pi^t, \text{ if } i = \beta_t \textit{ such that } \beta_t \in T^2 \\ 0, \text{ otherwise} \end{cases} \tag{46}$$

We can see that this outcome satisfies the first line constraints in Eqs.45 since the related winners get zero utility. For the second line constraints, we have that $\beta_t \notin X'_{j,t}, \forall j > t$ if $t \in T^2$ due to that $\alpha_t + |X_{j,t}| - 1 < \beta_t$ when $W_f^t \ne X_{j,t}$. Thus, these constraints are satisfied. The final constraints are satisfied naturally. Therefore, $\{\pi'_i\}_{i \in W}$ is in the core. Next, we prove that this outcome is an MRC outcome.

For any $t \in T^1$, we have that the condition $(\exists k > t) W_f^t = X'_{k,t}$ is true. Then after step t, OP_2 processes the step $t-1, t-2, ..., 1$. For each step $t' \in \{t-1, t-2, ..., 1\}$, $X'_{k,t}$ will not change sine $t' < t < k$. Therefore, For any $t \in T^1$, there exists $k \in T^2$ satisfying $X'_{k,t} = W_f^t$. Thus we have:

$$\begin{aligned}
\sum_{i \in W} \pi_i &= \sum_{t \in T^1} \sum_{i \in W_f^t} \pi_i + \sum_{t \in T^2} \sum_{i \in W_f^t} \pi_i \\
&\le \sum_{t \in T^2} \sum_{j=1}^{t-1} \sum_{i \in X'_{t,j}} \pi_i + \sum_{t \in T^2} \sum_{i \in W_f^t} \pi_i \\
&\le \sum_{t \in T^2} (\sum_{j=1}^{t-1} |X'_{t,j}| \Pi^j + |W_f^t| \Pi^t) \\
&= \sum_{t \in T^2} \pi'_{\beta_t} = \sum_{i \in W} \pi'_i
\end{aligned} \tag{47}$$

The first inequality was established because the set W_f^t for $t \in T^1$ is included in the union of $X'_{t,j}$ for $t \in T^2$. The second inequality is because of the core constraints in Eqs.35. According to this inequality, $\sum_{i \in W} \pi'_i$ is an upper bound for the total utility in ca^2 so that $\{\pi'_i\}_{i \in W}$ is an MRC outcome.

Next, we prove that $\sum_{i \in W} \pi'_i$ is also an upper bound of total utility in the instance ca^1. First we prove that $|X'_{t,j}| = |X_{t,j}|, \forall t \in T^2$. In OP_2, two operations exist for the subset $X'_{t,j}$. The first operation is letting $X'_{t,j} = \emptyset$ when $t \in T^1$, which can not change the size fo $X'_{t,j}$ where $t \in T^2$. The second operation $X'_{j,t} = \{\alpha_t, ..., \alpha_t + |X_{j,t}| - 1\}$ does not change the size of $X'_{j,t}$, either. Since the $X'_{t,j}$ is initialized to be $X'_{t,j}$, we have $|X'_{t,j}| = |X_{t,j}|, \forall t \in T^2$. Thus, in the instance ca^1, we have:

$$\begin{aligned}
\sum_{i \in W} \pi_i &= \sum_{t \in T^1} \sum_{i \in W_f^t} \pi_i + \sum_{t \in T^2} \sum_{i \in W_f^t} \pi_i \\
&\le \sum_{t \in T^2} \sum_{j=1}^{t-1} \sum_{i \in X_{t,j}} \pi_i + \sum_{t \in T^2} \sum_{i \in W_f^t} \pi_i \\
&\le \sum_{t \in T^2} (\sum_{j=1}^{t-1} |X_{t,j}| \Pi^j + |W_f^t| \Pi^t) \\
&= \sum_{t \in T^2} (\sum_{j=1}^{t-1} |X'_{t,j}| \Pi^j + |W_f^t| \Pi^t) \\
&= \sum_{t \in T^2} \pi'_{\beta_t} = \sum_{i \in W} \pi'_i
\end{aligned} \tag{48}$$

The first inequality is established because $X_{t,j} = X'_{t,j}$ for those subsets satisfying that $X'_{t,j} = W_f^t$ where $t \in T^2$. Therefore, $\sum_{t \in W} \pi'_t$ is also an upper bound for the total utilities in ca^1. Since this upper bound is the total utility of the core outcome $\{\pi'_i\}_{i \in W}$. Thus, this lemma is established. □

Therefore, we have:

$$r^{ca^1} \le \frac{\sum_{i \in W} \pi'_i}{\sum_{t \in W} \pi_i^{BLO}} = r^{ca^2} \tag{49}$$

After the operation OP_2, we get the new instance ca^2 whose ratio is no less than that of ca^1. For convenience, we use $X_{i,j}$ to represent $X'_{i,j}$ in the instance ca^2 in the following proof. Based on the MRC utility outcome in Eqs.46, the ratio of ca^2 can be rewritten as follows:

$$r^{ca^2} = \frac{k_1 \Pi^1 + k_2 \Pi^2 +, ..., k_T \Pi^T}{|W_f^1| \Pi^1 + |W_f^2| \Pi^2 +, ..., + |W_f^T| \Pi^T} \tag{50}$$

where

$$k_t = \begin{cases} \sum_{j \in \{t+1,...,T\}} |X_{j,t}|, \text{ if } t \in T^1 \\ |W_f^t| + \sum_{j \in \{t+1,...,T\}} |X_{j,t}|, \text{ if } t \in T^2 \end{cases} \tag{51}$$

D.3 Operation OP_3

In this section, we introduce the operation OP_3 and prove $r^{ca^2} \le r^{ca^3}$.

According to the construction of instance ca^2, we can see that the water-filling process is the same with the prerequisite that $0 \le \Pi^1 \le \Pi^2 \le ... \le \Pi^T$. $\{\Pi^t\}_{i \in [1,T]}$ is a group of real numbers to construct the instance. Let $\Pi^0 = 0, \Pi^{T+1} = \infty$, and we begin with the two operations below to operate on these numbers.

Definition 7 (Utility-change Operation)**.** *Given that $t \in [1, T]$, the utility-change operation changes the utility Π^t to $\tilde{\Pi}^t$ such that $\Pi^{t-1} \le \tilde{\Pi}^t \le \Pi^{t+1}$.*

The utility-change operation is achieved by changing the bid profile of the instance ca^2 based on Eqs.42. After this operation, the ratio becomes:

$$\tilde{r} = \frac{k_1 \Pi^1 + k_2 \Pi^2 +, ... + k_t \tilde{\Pi}^t +, ..., + k_T \Pi^T}{|W_f^1| \Pi^1 + |W_f^2| \Pi^2 +, ..., + |W_f^t| \tilde{\Pi}^t +, ... + |W_f^T| \Pi^T} \tag{52}$$

Definition 8 (Cut-off Operation)**.** *Given that $t \in [2, T]$, the cut-off operation cut off the winners' utilities that are frozen before step t, i.e., let $\Pi^j = 0, \forall j < t$.*

The cut-off operation is achieved by decreasing the utility Π^i to zero in the order of $i = 1, 2, ..., t-1$ through the utility-change operation. After the cut-off operation, the ratio becomes:

$$\frac{k_t \Pi^t + k_{t+1} \Pi^{t+1}, ..., + k_T \Pi^T}{|w_f^t| \Pi^t + |w_f^{t+1}| \Pi^{t+1}, ... + |w_f^T| \Pi^T} \tag{53}$$

Then we can define the final operation OP_3 as follows:

Definition 9 (Operation OP_3)**.** *Given the instance ca^2, operation OP_3 return a new instance ca^3 through the algorithm 3:*

Operation OP_3 also is an operation with multiple steps. In step t, we assign the value for r^t as follows:

$$r^t = \frac{\sum_{i=t}^T k_i}{\sum_{i=t}^T |W_f^i|} \tag{54}$$

Then we compare r^t with the current ratio r^{ca^3}. If $r^t < r^{ca^3}$, the value of $\Pi^i (i \ge t)$ is assigned to be Π^{t-1} through the duplicate utility-change operations. Otherwise, we just cut off at t and get the final instance ca^3.

We begin with a simple lemma as follows:

Algorithm 3 Operation OP_3 to generate the instance ca^3

$ca^3 \leftarrow ca^2$
for $t = T, T-1, ..., 2$ **do**
  $r^t \leftarrow \frac{\sum_{i=t}^{T} k_i}{\sum_{i=t}^{T} |W_f^i|}$
  if $r^t < r^{ca^3}$ **then**
    for $j = t, ..., T$ **do**
      $\Pi^j \leftarrow \Pi^{t-1}$ ▷ Utility-change operation
  else
    Carry the cut-off operation at t
    break
return ca^3

Lemma 6. *If $x_1, x_1', x_2, x_2' \in R$ and satisfy $x_1 > x_1' > 0, x_2 > x_2' > 0$, we have the following inequality:*

$$\frac{x_1 - x_1'}{x_2 - x_2'} > \frac{x_1}{x_2}, \text{ if } \frac{x_1}{x_2} > \frac{x_1'}{x_2'}$$

Then we have the following lemma:

Lemma 7. *After the operation OP_3, the ratio between the MRC outcome and the BLO outcome does not decrease, i.e., $r^{ca^2} \leq r^{ca^3}$.*

Proof. At the beginning of step t, the ratio of ca^3 is

$$r^{ca^3} = \frac{k_1\Pi^1 +, ... + k_t\Pi^t +, ..., + k_T\Pi^T}{|W_f^1|\Pi^1 +, ..., + |W_f^t|\Pi^t +, ... + |W_f^T|\Pi^T} \tag{55}$$

On the one hand, if $r^t < r^{ca^3}$, we would let $\Pi^j = \Pi^{t-1}, \forall j \geq t$ in instance ca^3. Denote by r^{new} the ratio of this new instance, then we have:

$$\begin{aligned} r^{new} &= \frac{k_1\Pi^1 +, ..., + k_T\Pi^{t-1}}{|W_f^1|\Pi^1 +, ..., + |W_f^T|\Pi^t} \\ &= \frac{k_1\Pi^1 +, ..., + k_T\Pi^t - (\Pi^t - \Pi^{t-1})\sum_{j\in[t,T]} k_j}{|W_f^1|\Pi^1 +, ..., + |W_f^T|\Pi^t - (\Pi^t - \Pi^{t-1})\sum_{j\in[t,T]} |W_f^j|} \end{aligned} \tag{56}$$

Since $r^t = \frac{\sum_{i=t}^{T} k_i \Pi^i}{\sum_{i=t}^{T} |W_f^i|\Pi^i} < r^{ca^3}$, then we have $r^{new} \geq r^{ca^3}$ according to lemma 6. Besides, $r^{new} = r^{ca^3}$ if and only if $\Pi^t = \Pi^{t-1}$. Thus, the ratio does not decrease in this step.

On the other hand, if $r^t \geq r^{ca^3}$, by the cut-off operation, OP_3 returns the final instance with the ratio r^t. we know that $r^t \geq r^{ca^3} \geq r^{ca^2}$ such that the ratio does not decrease, either. Therefore, this lemma is proved. □

In the instance ca^3, the BLO utility outcome becomes

$$\pi_i = \begin{cases} 0, & \text{if } i \text{ is frozen before or at the } z\text{-th iteration} \\ \hat{\Pi}, & \text{otherwise} \end{cases} \tag{57}$$

where z is the demarcation point and $z \in [1, T]$. $\hat{\Pi} = \Pi^z$ when the cut-off operation is executed at step z. If the cut-off operation is not operated, we have $z = 1$ and $\hat{\Pi} = \Pi^1$.

For the winners that are frozen at step t where $t \leq z$ in instance ca^3, we have the corresponding core constraints as follows:

$$\sum_{j\in\{1,...,t-1\}} \sum_{i\in X_{t,j}} \pi_i + \sum_{i\in W_f^t} \pi_i \leq 0, \ \forall t \in [1, z] \tag{58}$$

According to these constraints, we have $\pi_i = 0, \forall i \in W_f^t$ for all $t \leq z$, which is true for all the core outcomes.

D.4 Final Worst-case Ratio

In this section, we prove that $r^{ca^3} \leq \frac{|W|+2+\frac{|W| \mod 2}{|W|}}{4}$. First, we have the following lemmas:

Lemma 8. *For any Pareto optimal core outcome $\{\pi_i\}_{i\in W}$, each winner j must have a binding core constraint to restrict his utility as follows:*

$$\pi_j + \sum_{i\in W\setminus(S\cup\{j\})} \pi_i = w(N) - w(S), \ S \subseteq N \tag{59}$$

Proof. Assuming that no binding core constraint exists for one winner j, the utility of j can increase by ϵ (ϵ is small enough) so that new payment $\pi_j + \epsilon$ is still in the core. This conflicts with the Pareto optimal definition. □

For a Pareto optimal core outcome $\{\pi_i\}_{i\in W}$, we divide the winners into three types as follows:

- $T_1 : \{i | \pi_i^{MRC} > \pi_i\}$
- $T_2 : \{i | \pi_i^{MRC} = \pi_i\}$
- $T_3 : \{i | \pi_i^{MRC} < \pi_i\}$

Then we have the following lemma:

Lemma 9. *For a Pareto optimal core outcome $\{\pi_i\}_{i\in W}$, the following inequality is true:*

$$\pi_j^{MRC} - \pi_j \leq \sum_{i\in T_3} (\pi_i - \pi_i^{MRC}), \forall j \in T_1 \tag{60}$$

Proof. Consider the binding constraint of j as follows:

$$\pi_j + \sum_{i\in W\setminus(S\cup\{j\})} \pi_i = w(N) - w(S) \tag{61}$$

For any MRC outcome $\{\pi_i^{MRC}\}_{i\in W}$, we have:

$$\pi_j^{MRC} + \sum_{i\in W\setminus(S\cup\{j\})} \pi_i^{MRC} \leq w(N) - w(S) \tag{62}$$

Substitute Eq.61 into Eq.62, and we have:

$$\begin{aligned} \pi_j^{MRC} - \pi_j &\leq \sum_{i\in W\setminus(S\cup\{j\})} (\pi_i - \pi_i^{MRC}) \\ &\leq \sum_{i\in T_3} (\pi_i - \pi_i^{MRC}) \end{aligned}$$

Thus, this lemma is proved. □

Since the BLO outcome is Pareto optimal, we can also divide the winners into three types as follows:

- $T_1 : \{i|\pi_i^{MRC} > \pi_i^{BLO}\}$
- $T_2 : \{i|\pi_i^{MRC} = \pi_i^{BLO}\}$
- $T_3 : \{i|\pi_i^{MRC} < \pi_i^{BLO}\}$

Then we have:

$$\begin{aligned} r^{ca^3} &= \frac{\sum_{i\in W}\pi_i^{MRC}}{\sum_{i\in W}\pi_i^{BLO}} \\ &\le \frac{\sum_{i\in W}\pi_i^{BLO} + |T_1|\sum_{i\in T_3}(\pi_i^{BLO}-\pi_i^{MRC})}{\sum_{i\in W}\pi_i^{BLO}} \\ &\le 1+\frac{(|T_1|-1)\sum_{i\in T_3}(\pi_i^{BLO}-\pi_i^{MRC})}{\sum_{i\in W}\pi_i^{BLO}} \\ &\le 1+\frac{(|T_1|-1)|T_3|\hat{\Pi}}{(|W|-z)\hat{\Pi}} \\ &= 1+\frac{(|T_1|-1)|T_3|}{(|W|-z)} \end{aligned} \tag{63}$$

Due to that $|T_1|+|T_3| \le |W|-z$, we have:

$$\begin{aligned} 1+\frac{(|T_1|-1)|T_3|}{(|W|-z)} &\le 1+\frac{(|T_1|+|T_3|-1)^2}{4(|W|-z)} \\ &\le 1+\frac{(|W|-z-1)^2}{4(|W|-z)} \\ &\le \frac{(|W|-z)+\frac{1}{|W|-z}+2}{4} \\ &\le \frac{|W|+2+\frac{1}{|W|}}{4} \end{aligned} \tag{64}$$

The conditions for the above inequalities to hold are $|T_1|-1=|T_3|=\frac{|W|-1}{2}$ and $z=0$. These conditions can be satisfied when $|W|$ is odd. Instead, when $|W|$ is even, there are two cases: 1) $z=0$, the tight upper bound is $\frac{|W|+2}{4}$; 2) $z>0$, the tight upper bound is $\frac{|W|+1+\frac{1}{|W|-1}}{4}$ where $z=1$, which is smaller than that in the former case. Therefore, the final worst-case ratio can be written as:

$$\frac{\sum_{i\in W}\pi_i^{MRC}}{\sum_{i\in W}\pi_i^{BLO}} \le \frac{|W|+2+\frac{|W| \mod 2}{|W|}}{4} \tag{65}$$

Theorem 3 is established.

For both cases, we offer the worst-case instance with this ratio achieved. When $|W|$ is even, the bid profile is constructed as follows:

$$\begin{cases} \text{bidder } i : \{I_i\} = 2,\ \forall i \in [1,|W|]\ \textbf{(winners)} \\ \text{bidder } |W|+i : \{I_1,...,I_{\frac{|W|}{2}}, I_{\frac{|W|}{2}+i}\} = \dfrac{|W|}{2}+1, \\ \forall i \in [1, \dfrac{|W|}{2}] \end{cases} \tag{66}$$

where $I_1,...,I_{|W|}$ are the bid item bundles of the winners. Then the core of this instance is as follows:

$$Core = \begin{cases} \pi_1 \ge 0, \pi_2 \ge 0, ..., \pi_{|W|} \ge 0 \\ \pi_1+...+\pi_{\frac{|W|}{2}}+\pi_{\frac{|W|}{2}+i} \le \dfrac{|W|}{2}+1, \\ \forall i \in [1, \dfrac{|W|}{2}] \end{cases} \tag{67}$$

The MRC outcome is that $\pi_i = 0, \forall i \le \frac{|W|}{2}$ and $\pi_i = \frac{|W|}{2}+1, \forall i \in [\frac{|W|}{2}, |W|]$. The BLO outcome is $\pi_i = 1, \forall i \in [1,|W|]$. Thus, the ratio is:

$$\frac{\sum_{i\in W}\pi_i^{MRC}}{\sum_{i\in W}\pi_i^{BLO}} = \frac{(\frac{|W|}{2}+1)\frac{|W|}{2}}{|W|} = \frac{|W|+2}{4} \tag{68}$$

When $|W|$ is odd, the bid profile is constructed as follows:

$$\begin{cases} bidder\ i : \{I_i\} = 2,\ \forall i \in [1,|W|]\ \textbf{(winners)} \\ bidder\ |W|+i : \{I_1,...,I_{\frac{|W|-1}{2}}, I_{\frac{|W|-1}{2}+i}\} = \dfrac{|W|+1}{2}, \\ \forall i \in [1, \dfrac{|W|+1}{2}] \end{cases} \tag{69}$$

where $I_1,...,I_{|W|}$ are the bid item bundles of the winners.

E The worst-case ratio of normal Pareto optimal outcome

In this section, we analyze the worst-case ratio of normal Pareto optimal outcomes. The theorem is as follows:

Theorem 5. *Assuming that bidders bid truthfully, given a Pareto optimal utility outcome $\{\pi_i\}_{i\in N}$, we have the following tight lower bound:*

$$\sum_{i\in N}\pi_i \ge \frac{1}{|W|-1}\sum_{i\in N}\pi_i^{MRC}$$

where $\{\pi_i^{MRC}\}_{i\in N}$ is an MRC outcome.

Proof. According to Lemma 9, we have

$$\begin{aligned} \frac{\sum_{i\in W}\pi_i^{MRC}}{\sum_{i\in W}\pi_i} &\le \frac{\sum_{i\in W}\pi_i + |T_1|\sum_{i\in T_3}(\pi_i-\pi_i^{MRC})}{\sum_{i\in W}\pi_i} \\ &\le 1+\frac{(|T_1|-1)\sum_{i\in T_3}\pi_i}{\sum_{i\in W}\pi_i} \\ &\le |T_1| \\ &\le |W|-1 \end{aligned} \tag{70}$$

Thus, $\frac{1}{|W|-1}$ is the lower bound of the worst-case ratio for a Pareto optimal outcome. Next, we construct the worst-cause instance to prove the tightness. The bid profile is as follows:

$$\begin{cases} \text{bidder } i : \{I_i\} = 2,\ \forall i \in [1,|W|]\ \textbf{(winners)} \\ \text{bidder } |W|+i : \{I_1, I_{i+1}\} = 2, \\ \forall i \in [1, |W|-1] \end{cases} \tag{71}$$

where $I_1,...,I_{|W|}$ are the bid item bundles of the winners. Then the core of this instance is as follows:

$$Core = \begin{cases} \pi_1 \ge 0, \pi_2 \ge 0, ..., \pi_{|W|} \ge 0 \\ \pi_1+\pi_i \le 2, \forall i \in [2,|W|] \end{cases} \tag{72}$$

The MRC outcome is that $\pi_1 = 0$ and $\pi_i = 2, \forall i \in [2,|W|]$ and the total utility is $2(|W|-1)$. We consider a Pareto optimal outcome where $\pi_1 = 2$ and $\pi_i = 0, \forall i \in [2,|W|]$, with the total utility of 2. Thus, the ratio is $\frac{1}{|W|-1}$ in this case, and theorem 5 is proved. □

F Proof for the Correctness and Complexity of the WF-CGS-CR Algorithm

In this section, we would like to prove the correctness and complexity of the WF-CGS-CR algorithm from its three components.

F.1 Water-filling Algorithm

Proof for lemma 2. The maximum utility increment corresponds to a boundary constraint in the core. According to Eqs.35, these boundary constraints are given as follows:

$$\sum_{i\in W\backslash B^t}\pi_i \le w(N)-w(B^t),\ \forall t\in[1,T] \tag{73}$$

where $B^t \subseteq N$ is the optimal coalition for problem P1. All the above boundary inequalities take the equal sign in the final result $\{\pi_i^T\}_{i\in N}$.

Assume that $\{\pi_i^T\}_{i\in W}$ is not the BLO outcome, then there would be another outcome $\{\pi_i'\}_{i\in W}$ such that $\{\pi_i'\}_{i\in W} \succ_{Lex} \{\pi_i^T\}_{i\in W}$. Let $\{\hat{\pi}_i'\}$ and $\{\hat{\pi}_i^T\}$ be the corresponding leximin sequence of these two outcomes and $\hat{k}$ be the index in the leximin sequence that satisfies that $\hat{\pi}_i' = \hat{\pi}_i^T, \forall i \le \hat{k}$ and $\hat{\pi}_{\hat{k}}' > \hat{\pi}_{\hat{k}}^T$. Denote by k the winner with the index $\hat{k}$. Without loss of generation, assuming that winner k is frozen at the t-th iteration, we have the corresponding boundary constraint as follows:

$$\sum_{i\in W\backslash B^t}\pi_i \le w(N)-w(B^t) \tag{74}$$

where $k\in W\backslash B^t$. For winner i in the subset $W\backslash B^t$, if the index of this winner is smaller than k, we have $\pi_i' = \pi_i^T$. On the other hand, if the index of this winner is larger than $\hat{k}$, we have $\pi_i' \ge \hat{\pi}_{\hat{k}}' > \pi_k^T$. Thus, we have:

$$\sum_{i\in W\backslash B^t}\pi_i' > \sum_{i\in W\backslash B^t}\pi_i^T = w(N)-w(B^t) \tag{75}$$

This breaks the above core constraint in Eq.74, which produces a contradiction. Thus, this lemma is proved. □

F.2 Constraint Generation Search Subroutine

In this section, we prove the correctness of the CGS subroutine from three aspects:

1. Eq.15 is the correct condition to check whether the utility distribution is in the core.
2. Eq.16 provides the correct upper bound update, and Eq.17 provides the correct frozen winner set update.
3. $|\bar{W}_f| = 1$ is a correct condition for the termination.

To prove these, we begin with three lemmas as follows:

Lemma 10. *For bidder set S, a bid profile $\{b_i(\cdot)\}_{i\in S}$ and a utility outcome $\{\pi_i\}_{i\in S}$, we have:*

$$w(S)-\sum_{i\in S}\pi_i \le w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in S}) \tag{76}$$

Proof. Assuming that the optimal allocation result is $\{a_i^S\}_{i\in S}$ in the bid profile $\{b_i(\cdot)\}_{i\in S}$, then we have:

$$\begin{aligned} w(S)-\sum_{i\in S}\pi_i &= \sum_{i\in S}b_i(a_i^S)-\sum_{i\in S}\pi_i \\ &\le \sum_{i\in S}\max(b_i(a_i^S)-\pi_i,0) \\ &\le w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in S}) \end{aligned} \tag{77}$$

The last inequality is true because $\{a_i^S\}_{i\in S}$ is a feasible allocation solution while $w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in S})$ is the value of the optimal allocation result. □

Lemma 11. *A utility outcome $\{\pi_i\}_{i\in N}$ is in the core if and only if*

$$w(N)-\sum_{i\in N}\pi_i = w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}) \tag{78}$$

Proof. Assuming that this equality is not true, we let B be a winner set under the truncated bid profile $\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}$ and let $\{a_i^B\}_{i\in B}$ be the allocation result. According to Lemma 10, we have:

$$\begin{aligned} w(N)-\sum_{i\in N}\pi_i &< w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}) \\ &= \sum_{i\in B}b_i(a_i^B)-\sum_{i\in B}\pi_i \\ &\le w(B)-\sum_{i\in B}\pi_i \end{aligned} \tag{79}$$

Then B is a block coalition, which produces a contradiction. Conversely, assuming this equality is true, and this point is out of the core. Then, there must be an unsatisfying core constraint as follows:

$$\begin{aligned} W(N)-\sum_{i\in N}\pi_i &< w(S)-\sum_{i\in S}\pi_i \\ &\le w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in S}) \\ &\le w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}) \end{aligned} \tag{80}$$

This also produces a contradiction; thus, this lemma is proved. □

Lemma 12. *If a utility outcome $\{\pi_i\}_{i\in N}$ is not in the core, let B be one winner set under the bid profile $\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}$, then $B\in\arg\max_{S\in N} w(S)-\sum_{i\in S}\pi_i$ and it satisfies:*

$$w(B)-\sum_{i\in B}\pi_i = w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}) \tag{81}$$

Proof. Assuming that the allocation result is $\{a_i^B\}_{i\in B}$ under the truncated bid profile $\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}$, we have:

$$\begin{aligned} w(\{\max(b_i(\cdot)-\pi_i,0)\}_{i\in N}) &= \sum_{i\in B}\max(b_i(a_i^B)-\pi_i,0) \\ &= \sum_{i\in B}(b_i(a_i^B)-\pi_i) \\ &\le w(B)-\sum_{i\in B}\pi_i \end{aligned} \tag{82}$$

According to lemma 10, this equality is established. For any coalition S, we have:

$$
\begin{aligned}
w(S) - \sum_{i \in S} \pi_i &\leq w(\{\max(b_i(\cdot) - \pi_i, 0)\}_{i \in S}) \\
&\leq w(\{\max(b_i(\cdot) - \pi_i, 0)\}_{i \in N}) \\
&= w(B) - \sum_{i \in B} \pi_i
\end{aligned} \tag{83}
$$

Thus, this lemma is proved. $\square$

According to lemma 11, Eq.15 is the correct condition to check whether the current utility is in the core.

Next, we consider Eq.16 and Eq.17. For any coalition $S \subseteq N$, we have:

$$
\begin{aligned}
\Delta\pi^S &= \frac{(w(N) - \sum_{i \in N} \pi_i^t) - (w(S) - \sum_{i \in S} \pi_i^t)}{|W_a^t \backslash S|} \\
&= \Delta\bar{\pi} + \frac{(w(N) - \sum_{i \in W} \pi_i^t) - (w(S) - \sum_{i \in S} \pi_i^t) - |W_a^t \backslash S| \Delta\bar{\pi}}{|W_a^t \backslash S|} \\
&= \Delta\bar{\pi} - \frac{(w(S) - \sum_{i \in S} \tilde{\pi}_i) - (w(N) - \sum_{i \in N} \tilde{\pi}_i)}{|W_a^t \backslash S|}
\end{aligned} \tag{84}
$$

According to lemma 12, we have:

$$
\Delta\pi^B = \Delta\bar{\pi} - \frac{w(\{\max(b_i(\cdot) - \tilde{\pi}_i, 0)\}_{i \in N}) - (w(N) - \sum_{i \in N} \tilde{\pi}_i)}{|W_a^t \backslash B|} \tag{85}
$$

where RHS is the formulation in Eq.16. Thus Eq.16 represents an upper bound corresponding to the coalition B, i.e., $\Delta\pi^B$. The corresponding frozen winner set is thus updated by Eq.17.

Lastly, the condition of $|\bar{W}_f| = 1$ is included in the following proof lemma 3:

Proof for lemma 3. Assuming that there has been a coalition B in the CGS subroutine, for any subset $S \subseteq N$ such that $|W_a^t \backslash S| \geq |W_a^t \backslash B|$, we have:

$$
\begin{aligned}
\Delta\pi^S &= \Delta\bar{\pi} - \frac{(w(S) - \sum_{i \in S} \tilde{\pi}_i) - (w(N) - \sum_{i \in N} \tilde{\pi}_i)}{|W_a^t \backslash S|} \\
&\geq \Delta\bar{\pi} - \frac{(w(B) - \sum_{i \in B} \tilde{\pi}_i) - (w(N) - \sum_{i \in N} \tilde{\pi}_i)}{|W_a^t \backslash B|} \\
&= \Delta\pi^B
\end{aligned} \tag{86}
$$

This inequality is true according to lemma 12. Since the utility increment $\Delta\bar{\pi}$ decreases monotonously, such upper bound $\Delta\pi^S$ would not be smaller than the future utility increment. Therefore, only the coalition S such that $|W_a \backslash S| < |W_a \backslash B|$ needs to be searched by CGS.

For all the coalition $S \subseteq N$, there are $|W_a^t|$ integers for $|W_a^t \backslash S|$, and CGS removes at least one at each iteration. Eventually, there exists no potential upper bound where $|W_a^t \backslash S| < 1$; thus, CGS breaks the loop when $|\bar{W}_f| = 1$. Therefore, the CGS subroutine needs at most $|W_a^t|$ queries to the oracle. $\square$

F.3 Constraint-reuse Strategy

In this section, we explain the constraint-reuse strategy's recurrence formula and analyze the WF-CGS-CR algorithm's complexity.

We know that $\pi_i^t = \pi_i^{t-1} + \Delta\pi^{t-1}, \forall i \in W_a^t$ and the new active winner set is W_a^t. For any binary tuple $(W_f^S, \Delta\pi^S)$ at the t-1-th iteration, denote by $(\acute{W}_f^S, \Delta\acute{\pi}^S)$ the binary tuple at the t-th iteration, and we have:

$$
\begin{aligned}
\Delta\acute{\pi}^S &= \frac{w(N) - w(S) - \sum_{i \in N \backslash S} \pi_i^t}{|W_a^t \backslash S|} \\
&= \frac{w(N) - w(S) - \sum_{i \in W_a^t \backslash S} \pi_i^{t-1} - |W_a^t \backslash S| \Delta\pi^{t-1}}{|W_a^t \backslash S|} \\
&= \frac{|W_a^{t-1} \backslash S|}{|W_a^t \backslash S|} (\Delta\pi^S - \Delta\pi^{t-1})
\end{aligned} \tag{87}
$$

where $|W_a^{t+1} \backslash S| > 0$. If $|W_a^{t+1} \backslash S| = 0$, then the corresponding constraint has no potential to be tight in the following iterations; thus, we remove it. Given that $W_a^{t+1} \subset W_a^t$, we have:

$$
W_a^t \backslash S = W_a^t \cap (W_a^{t-1} \backslash S) = W_a^t \cap W_f^S \tag{88}
$$

Thus, the recurrence formula is as follows:

$$
\begin{cases}
\acute{W}_f^S = W_a^t \cap W_f^S \\
\Delta\acute{\pi}^S = \dfrac{|W_f^S|}{|\acute{W}_f^S|} (\Delta\pi^S - \Delta\pi^{t-1})
\end{cases} \tag{89}
$$

In conclusion, there are at most $\frac{|W|(|W|+1)}{2}$ calls to the CGS subroutine. Besides, the constraint set may include the quadratic binary tuples at most. Therefore, except for the queries to the oracle, the extra time complexity for the WF-CGS-CR algorithm is $O(|W|^4)$. Thus, Theorem 4 is established.

G Experiment

G.1 Implement Details

In this section, we provide the implementation details for our experiments. Here is a detailed description for the payment rules we implemented in this paper:

1. The Vickrey-Clarke-Groves (VCG) payment rule [Vickrey, 1961; Clarke, 1971; Groves, 1973]: The VCG utility is computed by $w(N) - w(N \backslash \{i\}), \forall i \in W$ so that it needs $|W|$ queries to the oracle.
2. Minimum-Revenue-Core payment rule (MRC) [Day and Raghavan, 2007]: This algorithm is based on a heuristic called Core Constraint Generation (CCG). We use the version in [Day and Raghavan, 2007] in our experiment. This algorithm starts from the VCG utility point and an initial small linear program (LP) where utilities are above zero and below VCG utilities, and in each iteration, finds the most violated core constraint by the current point by calling the oracle $\mathcal{WD}$ once and adds this constraint to the LP. It then resolves the LP to find the next point and iterates until it finds a feasible core point, which is one MRC point.

3. VCG-nearest core payment rule (MRC-VCG) [Day and Cramton, 2012]: This algorithm finds the closest MRC point to the VCG point. At the first step, it finds an MRC point as above. Then, by adding a new constraint that restricts the total utility equal to that of this MRC point, it searches for another point that has the minimum l_2-distance to the VCG point by the same CCG process. This search is done using convex quadratic programming.
4. Zero-nearest core payment rule (MRC-Zero) [Day and Cramton, 2012]: Similar to MRC-VCG, this algorithm finds the closest MRC point to the all-zero point in l_2 distance. We use this payment rule to determine how many winners are forced to get zero utility due to the MRC rule. This is because MRC-Zero prefers non-zero points given the fixed total utility.
5. Fast Core [Niazadeh *et al.*, 2022]: Fast Core is an approximation algorithm based on the water-filling framework. In Fast Core, all the bid prices are normalized to section [0,1] at first. In each iteration, it searches the maximum utility increase by binary search from $[0, 1]$ until it finds a final section $[\Delta_l, \Delta_h]$ where Δ_l is the lower bound while Δ_h is the upper bound and $\Delta_h - \Delta_l \leq \epsilon$. Then it uses Δ_l to be the actual utility increase and uses Δ_h to find the next active winner set. This iteration continues until there are no active winners.

We use the XOR bid language in this paper, and the following integer program defines the winner determination problem:

$$
\begin{aligned}
\max \sum_{i \in N} \sum_{a \subseteq M} & x_i(a) \cdot b_i(a) \\
\text{s.t.} \quad \sum_{a \supseteq j} \sum_{i \in N} & x_i(a) \leq 1, \ \forall j \in M \\
\sum_{a \supseteq M} & x_i(a) \leq 1, \ \forall i \in N \\
& x_i(a) \in \{0, 1\}, \forall (a, i), \text{ such that a bid } b_i(a) \\
& \text{was submitted.}
\end{aligned}
$$

This standard integer program has a linear objective function and constraints. The variable $x_i(a) = 1$ means the item bundle S is allocated to bidder i, otherwise $x_i(a) = 0$. This first constraint means an item bundle cannot be allocated to two bidders. This second constraint means that one bidder has one winning bid at most, achieved through dummy items [Leyton-Brown *et al.*, 2000].

G.2 More Experimental Results

We show the results of the winners' utility distribution for six distributions in Table 4-9.We also explore the relationship between the run time and the number of bids and goods. The results are shown in Figure 1-6.

Table 4: Results for the winners' utility distribution for the CATS distribution of Arbitrary. All results are averages over 50 CA instances.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	4202.95	728.13	14.79	17.24	0
MRC-VCG	4776.17	154.90	0	13.16	58.19
MRC-Zero	4776.17	154.90	0	12.35	56.18
Fast Core	4825.88	105.20	1.87	6.13	23.06
BLO	4809.83	121.25	2.41	6.64	0

Table 5: Results for the winners' utility distribution for the CATS distribution of Decay(L4). All results are averages over 50 CA instances.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	61287.17	1592.40	4.02	38.39	0
MRC-VCG	62247.57	632.00	0	29.35	47.92
MRC-Zero	62247.57	632.00	0	27.26	41.52
Fast Core	62422.41	457.16	0.31	17.67	25.52
BLO	62339.78	539.79	1.18	21.21	0

Table 6: Results for the winners' utility distribution for the CATS distribution of Matching. All results are averages over 50 CA instances.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	199.68	38.48	0.05	1.20	0
MRC-VCG	206.2	31.96	0	1.14	9.71
MRC-Zero	206.2	31.96	0.01	1.09	8.20
Fast Core	207.25	30.91	0.04	1.05	0.71
BLO	207.12	31.04	0.04	1.05	0

Table 7: Results for the winners' utility distribution for the CATS distribution of Paths. All results are averages over 50 CA instances.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	26.28	2.18	0	0.06	10.11
MRC-VCG	26.67	1.78	0	0.05	19.96
MRC-Zero	26.67	1.78	0	0.05	18.77
Fast Core	26.75	1.70	0	0.05	11.12
BLO	26.74	1.71	0	0.05	10.11

Table 8: Results for the winners' utility distribution for the CATS distribution of Regions. All results are averages over 50 CA instances.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	4174.11	687.20	11.01	22.44	0
MRC-VCG	4667.61	193.69	0	15.97	43.68
MRC-Zero	4667.61	193.69	0.21	14.03	36.53
Fast Core	4716.96	144.34	2.53	8.37	13.95
BLO	4698.67	162.63	3.21	9.31	0

Table 9: Results for the winners' utility distribution for the CATS distribution of Scheduling. All results are averages over 50 CA instances.

Payment rule	Revenue	Total utility	Min ↑	Std↓	Zero ratio↓
VCG	79.08	29.84	0.30	2.45	0
MRC-VCG	81.33	27.59	0.06	2.48	10.29
MRC-Zero	81.33	27.59	0.07	2.46	10.21
Fast Core	82.47	26.45	0.21	2.36	1.93
BLO	82.32	26.60	0.21	2.37	0

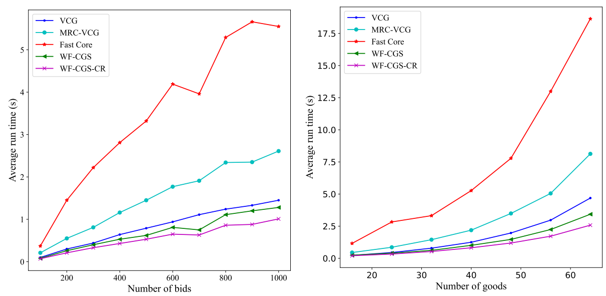

Figure 1: **Left**: results on various numbers of bids with 32 goods. **Right**: results on various numbers of goods with 500 bids. All results are averages over 50 CA instances, following the CATS distribution of Arbitrary.

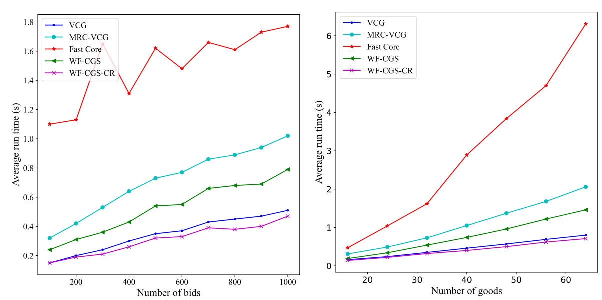

Figure 2: **Left**: results on various numbers of bids with 32 goods. **Right**: results on various numbers of goods with 500 bids. All results are averages over 50 CA instances, following the CATS distribution of Decay(L4).

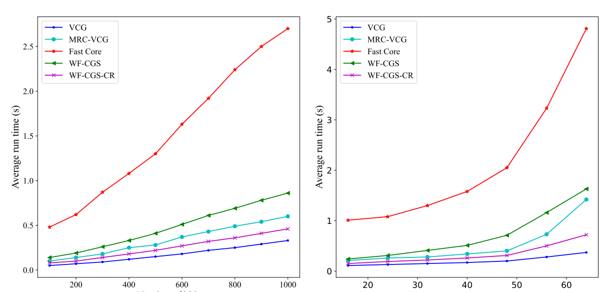

Figure 3: **Left**: results on various numbers of bids with 32 goods. **Right**: results on various numbers of goods with 500 bids. All results are averages over 50 CA instances, following the CATS distribution of Matching.

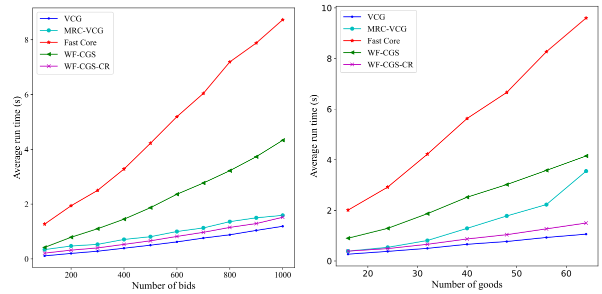

Figure 4: **Left**: results on various numbers of bids with 32 goods. **Right**: results on various numbers of goods with 500 bids. All results are averages over 50 CA instances, following the CATS distribution of Paths.

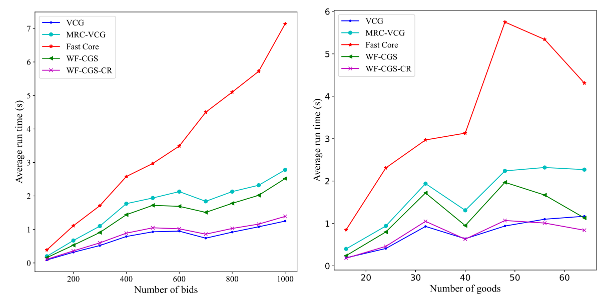

Figure 5: **Left**: results on various numbers of bids with 32 goods. **Right**: results on various numbers of goods with 500 bids. All results are averages over 50 CA instances, following the CATS distribution of Regions.

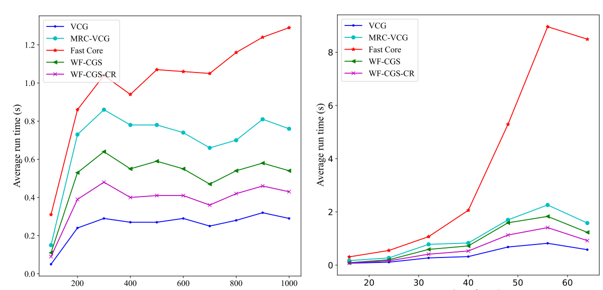

Figure 6: **Left**: results on various numbers of bids with 32 goods. **Right**: results on various numbers of goods with 500 bids. All results are averages over 50 CA instances, following the CATS distribution of Scheduling.

H Future Work

One future direction is to explore Bayesian equilibrium of the BLO payment rule. In this paper, we assume that the bidders will bid truthfully, but core-selecting CAs are not guaranteed to be truthful. Thus, Bayesian equilibrium is an effective way to analyze the incentive properties for the core-selecting payment rules[Bosshard *et al.*, 2017; Bünz *et al.*, 2018].

Moreover, the weighted BLO payment rule is a natural expansion of the BLO rule. For example, Niazadeh proposed to use the ratio to the VCG utility as the measurement to sort by the leximin principle [Niazadeh *et al.*, 2022]. Thus, future research will study how to design the weight to achieve the specific goals of fairness and design effective pricing algorithms.