Fourier Analysis-based Iterative Combinatorial Auctions*

Jakob Weissteiner1,4†, Chris Wendler2†, Sven Seuken1,4, Ben Lubin3 and Markus Püschel2
1University of Zurich
2ETH Zurich
3Boston University
4ETH AI Center
weissteiner@ifi.uzh.ch, chris.wendler@inf.ethz.ch, seuken@ifi.uzh.ch, blubin@bu.edu, pueschel@inf.ethz.ch

Abstract
Recent advances in Fourier analysis have brought new tools to efficiently represent and learn set functions. In this paper, we bring the power of Fourier analysis to the design of combinatorial auctions (CAs). The key idea is to approximate bidders’ value functions using Fourier-sparse set functions, which can be computed using a relatively small number of queries. Since this number is still too large for practical CAs, we propose a new hybrid design: we first use neural networks (NNs) to learn bidders’ values and then apply Fourier analysis to the learned representations. On a technical level, we formulate a Fourier transform-based winner determination problem and derive its mixed integer program formulation. Based on this, we devise an iterative CA that asks Fourier-based queries. We experimentally show that our hybrid ICA achieves higher efficiency than prior auction designs, leads to a fairer distribution of social welfare, and significantly reduces runtime. With this paper, we are the first to leverage Fourier analysis in CA design and lay the foundation for future work in this area. Our code is available on GitHub: https://github.com/marketdesignresearch/FA-based-ICAs.

1 Introduction
Combinatorial auctions (CAs) are used to allocate multiple heterogeneous items to bidders. CAs are particularly useful in domains where bidders’ preferences exhibit complementarities and substitutabilities as they allow bidders to submit bids on bundles of items rather than on individual items. Since the bundle space grows exponentially in the number of items, it is impossible for bidders to report values for all bundles in settings with more than a modest number of items. Thus, parsimonious preference elicitation is key for the practical design of CAs. For general value functions, Nisan and Segal [2006] have shown that to guarantee full efficiency, exponential communication in the number of items is needed.

Thus, practical CAs cannot provide efficiency guarantees in large domains. Instead, recent proposals have focused on iterative combinatorial auctions (ICAs), where the auctioneer interacts with bidders over rounds, eliciting a limited amount of information, aiming to find a highly efficient allocation.

ICAs have found widespread application; most recently, for the sale of licenses to build offshore wind farms [Ausubel and Crumton, 2011]. For the sale of spectrum licenses, the combinatorial clock auction (CCA) [Ausubel et al., 2006] has generated more than $20 billion in total revenue [Ausubel and Baranov, 2017]. Thus, increasing the efficiency by only 1–2% points translates into monetary gains of millions of dollars.

1.1 Machine Learning-based Auction Design
Recently, researchers have used machine learning (ML) to improve the performance of CAs. Early work by Blum et al. [2004] and Lahaie and Parkes [2004] first studied the relationship between learning theory and preference elicitation in CAs. Dütting et al. [2019], Shen et al. [2019] and Rahme et al. [2021] used neural networks (NNs) to learn whole auction mechanisms from data. Brero et al. [2019] introduced a Bayesian ICA using probabilistic price updates to achieve faster convergence. Shen et al. [2020] use reinforcement learning for dynamic pricing in sponsored search auctions. Most related to the present paper is the work by Brero et al. [2018; 2021], who developed a value-query-based ML-powered ICA using support vector regressions (SVRs) that achieves even higher efficiency than the CCA. In follow-up work, Weissteiner and Seuken [2020] extended their ICA to NNs, further increasing the efficiency. In work subsequent to the first version of this paper, Weissteiner et al. [2022] proposed Monotone-Value Neural Networks (MVNNs), which are particularly well suited to learning value functions in combinatorial assignment domains. However, especially in large domains, it remains a challenge to find the efficient allocation while keeping the elicitation cost low. Thus, even state-of-the-art approaches suffer from significant efficiency losses and often result in unfair allocations, highlighting the need for better preference elicitation algorithms.

1.2 Combining Fourier Analysis and CAs
The goal of preference elicitation in CAs is to learn bidders’ value functions using a small number of queries. Mathematically, value functions are set functions, which are in general
exponentially large and notoriously hard to represent or learn. To address this complexity, we leverage Fourier analysis for set functions [Bernasconi et al., 1996; O’Donnell, 2014; Püschel and Wendler, 2020]. In particular, we consider Fourier-sparse approximations, which are represented by few parameters. These parameters are the non-zero Fourier coefficients (FCs) obtained by a base change with the Fourier transform (FT). We use the framework by Püschel and Wendler [2020], which contains new FTs beyond the classical Walsh-Hadamard transform (WHT) [Bernasconi et al., 1996], providing more flexibility. Until recently, methods for learning Fourier-sparse set functions focused on the WHT, and they placed assumptions on bidders’ value functions that are too restrictive for CAs [Stobbe and Krause, 2012]. However, recently, Amrollahi et al. [2019] proposed a new algorithm that can approximate general set functions by WHT-sparse ones, which is suitable for large CAs and we use it in this work.

1.3 Our Contribution

Our main contribution in this paper is to bring the power of Fourier analysis to CA design (Section 3). In particular, we formulate FT-based winner determination problems (WDPs) and derive corresponding mixed integer programs (MIPs) for several FTs (Section 4). Our MIPs allow for the efficient solution of the FT-based WDP and provide the foundation for using Fourier-sparse approximations in auction design.

We first experimentally show that the WHT performs best among the FTs in terms of induced level of sparsity (Section 5.1) and reconstruction error (Section 5.2). As an initial approach, we develop a WHT-based allocation rule (Section 5.3). However, this requires too many queries for direct use in CAs. To overcome this, we propose a practical hybrid ICA based on NNs and Fourier analysis (Section 6.1). The key idea is to compute Fourier-sparse approximations of NN-based bidder representations, enabling us to keep the number of queries small. The advantage of the NN-based representations is that they capture key aspects of the bidders’ value functions and can be queried arbitrarily often (Section 6.2).

Our efficiency experiments show that our hybrid ICA achieves higher efficiency than state-of-the-art mechanisms, leads to a significant computational speedup, and yields fairer allocations (Section 6.3). This shows that leveraging Fourier analysis in CA design is a promising new research direction.

2 Preliminaries

In this section, we present our formal model and review the MLCA mechanism, which our hybrid ICA builds upon.

2.1 Formal Model for ICAs

We consider a CA with n bidders and m indivisible items. Let \( N = \{1, \ldots, n\} \) and \( M = \{1, \ldots, m\} \) denote the set of bidders and items, respectively. We denote \( x \in X = \{0, 1\}^m \) a bundle of items represented as an indicator vector, where \( x_j = 1 \) iff item \( j \in M \) is contained in \( x \). Bidders’ true preferences over bundles are represented by their (private) value functions \( v_i : X \to \mathbb{R}_+ \), \( i \in N \), i.e., \( v_i(x) \) represents bidder \( i \)'s true value for bundle \( x \). By \( a = (a_1, \ldots, a_n) \in \mathbb{N}^n \) we denote an allocation of bundles to bidders, where \( a_i \) is the bundle bidder \( i \) obtains. We denote the set of feasible allocations by \( \mathcal{F} = \{ a \in \mathbb{N}^m : \sum_{i \in N} a_{ij} \leq 1, \forall j \in M \} \). The (true) social welfare of an allocation \( a \) is defined as \( V(a) = \sum_{i \in N} v_i(a_i) \). We let \( a^* = \text{argmax}_{a \in \mathcal{F}} V(a) \) be a social-welfare maximizing, i.e., efficient, allocation. The efficiency of any \( a \in \mathcal{F} \) is measured by \( V(a)/V(a^*) \). We assume that bidders’ have quasilinear utilities \( u_i \), i.e., for a payments \( p \in \mathbb{R}^n_+ \) it holds that \( u_i(a, p) = v_i(a_i) - p_i \).

An ICA mechanism defines how the bidders interact with the auctioneer and how the final allocation and payments are determined. We denote a bidder’s (possibly untruthful) reported value function by \( \hat{v}_i : X \to \mathbb{R}_+ \). In this paper, we consider ICAs that ask bidders iteratively to report their value \( \hat{v}_i(x) \) for particular bundles \( x \) selected by the mechanism (for early work on value queries see [Hudson and Sandholm, 2003]). A finite set of such reported bundle-value pairs of bidder \( i \) is denoted as \( R_i = \{ (x^{(1)}, \hat{v}_i(x^{(1)})) \}, x^{(1)} \in X \). Let \( R = (R_1, \ldots, R_n) \) denote the tuple of reported bundle-value pairs obtained from all bidders. We define the reported social welfare of an allocation \( a \) given \( R \) as \( \hat{V}(a|R) = \sum_{i \in N} v_i(R_i(a_i)) \), where \( (a_i, \hat{v}_i(a_i)) \in R_i \) ensures that only values for reported bundles contribute. Finally, the optimal allocation \( a^*_R \in \mathcal{F} \) given the reports \( R \) is defined as
\[
\alpha_R^* = \text{argmax}_{a \in \mathcal{F}} \hat{V}(a|R). \tag{1}
\]

The final allocation \( a^*_R \in \mathcal{F} \) and payments \( p(R) \in \mathbb{R}^n_+ \) are computed based on the elicited reports \( R \) only.

As the auctioneer can only ask each bidder \( i \) a limited number of queries \( |R_i| \leq Q^{\text{max}}_i \), the ICA needs a smart preference elicitation algorithm, with the goal of finding a highly efficient \( a^*_R \) with a limited number of value queries.

2.2 A Machine Learning-powered ICA

We now review the machine learning-powered combinatorial auction (MLCA) by Brero et al. [2021]. Interested readers are referred to Appendix A.1, where we present MLCA in detail.

MLCA starts by asking each bidder value queries for \( Q^n_{\text{max}} \) randomly sampled initial bundles. Next, MLCA proceeds in rounds until a maximum number of value queries per bidder \( Q^{\text{max}} \) is reached. In each round, for each bidder \( i \in N \), it trains an ML algorithm \( A_i \) on the bidder’s reports \( R_i \). Next, MLCA generates new value queries \( q^{\text{new}}_{i} = (q^{\text{new}}_{i,j})_{j=1}^n \) with \( q^{\text{new}}_{i,j} \in X \setminus R_i \) by solving a ML-based WDP \( q^{\text{new}}_{i} = \text{argmax}_{a \in \mathcal{F}} \sum_{a_i \in N} A_i(a_i) \). The idea is the following: if \( A_i \) are good surrogate models of the bidders’ true value functions then \( q^{\text{new}}_{i} \) should be a good proxy of the efficient allocation \( a^* \) and thus provide valuable information.

At the end of each round, MLCA receives reports \( R^{\text{new}} \) from all bidders for the newly generated \( q^{\text{new}}_{i} \) and updates \( R \). When \( Q^{\text{max}} \) is reached, MLCA computes an allocation \( a^*_R \) maximizing the reported social welfare (eq. (1)) and determines VCG payments \( p(R) \) (see Appendix A.2).

2.3 Incentives of MLCA and Hybrid ICA

A key concern in the design of ICAs are bidders’ incentives. However, the seminal result by Nisan and Segal [2006] discussed above implies that practical ICAs cannot simply use
VCG to achieve strategyproofness. And in fact, no ICA deployed in practice is strategyproof — including the famous SMRA and CCA auctions used to conduct spectrum auctions. Instead, auction designers have designed mechanisms that, while being manipulable, have “good incentives in practice” (see [Cramton, 2013; Milgrom, 2007]).

Naturally, the MLCA mechanism is also not strategyproof, and Brero et al. [2021] provide a simple example of a possible manipulation. The idea behind the example is straightforward: if the ML algorithm does not learn a bidder’s preferences perfectly, a sub-optimal allocation may result. Thus, a bidder may (in theory) benefit from misreporting their preferences with the goal of “correcting” the ML algorithm, so that, with the misreported preferences, the mechanism actually finds a preferable allocation.

However, MLCA has two features that mitigate manipulations. First, MLCA explicitly queries each bidder’s marginal economy, which implies that the marginal economy term of the final VCG payment is practically independent of bidder i’s bid (for experimental support see [Brero et al., 2021]). Second, MLCA enables bidders to “push” information to the auction which they deem useful. This mitigates certain manipulations of the main economy term in the VCG payment rule, as it allows bidders to increase the social welfare directly by pushing (useful) truthful information, rather than attempting to manipulate the ML algorithm. Brero et al. [2021] argued that with these two design features, MLCA exhibits very good incentives in practice. They performed a computational experiment, testing whether an individual bidder (equipped with more information than he would have in a real auction) can benefit from deviating from truthful bidding, while all other bidders are truthful. In their experiments, they could not identify a beneficial manipulation strategy. While this does not rule out that some (potentially more sophisticated) beneficial manipulations do exist, it provides evidence to support the claim that MLCA has good incentives in practice.

With two additional assumptions, one also obtains a theoretical incentive guarantee for MLCA. Assumption 1 requires that, if all bidders bid truthfully, then MLCA finds an efficient allocation (we show in Appendix D.3 that in two of our domains, we indeed find the efficient allocation in the majority of cases). Assumption 2 requires that, for all bidders i, if all other bidders report truthfully, then the social welfare of bidder i’s marginal economy is independent of his value reports. If both assumptions hold, then bidding truthfully is an ex-post Nash equilibrium in MLCA.

Our hybrid ICA (Algorithm 1 in Section 6.1) is built upon MLCA, leaving the general framework in place, and only changing the algorithm that generates new queries each round. Given this design, the incentive properties of MLCA extend to the hybrid ICA. Specifically, our hybrid ICA is also not strategyproof, but it also has the same design features (including push-bids) to mitigate manipulations.

In future work, it would be interesting to evaluate experimentally whether the improved performance of the hybrid ICA translates into better manipulation mitigation compared to MLCA. However, such an analysis is beyond the scope of the present paper, which focuses on the ML algorithm that is integrated into the auction mechanism.

3 Fourier Analysis of Value Functions

We now show how to apply Fourier analysis to value functions providing the theoretical foundation of FT-based WDPs. Classic Fourier analysis decomposes an audio signal or image into an orthogonal set of sinusoids of different frequencies. Similarly, the classical Fourier analysis for set functions (i.e., functions mapping subsets of a discrete set to a scalar) decomposes a set function into an orthogonal set of Walsh functions [Bernasconi et al., 1996], which are piecewise constant with values 1 and −1 only. Recent work by Püschel and Wendler [2020] extends the Fourier analysis for set functions with several novel forms of set Fourier transforms (FTs). Importantly, because bidders’ value functions are set functions, they are amenable to this type of Fourier analysis, and it is this connection that we will leverage in our auction design.

**Sparsity.** The motivation behind our approach is that we expect bidders’ value functions to be sparse, i.e., they can be described with much less data than is contained in the exponentially-sized full value function. While this sparsity may be difficult to uncover when looking at bidders’ value reports directly, it may reveal itself in the Fourier domain (where then most FCs are zero). As all FTs are changes of basis, each FT provides us with a new lens on the bidder’s value function, revealing structure and thus potentially reducing dimensionality.

**Set function Fourier transform.** We now provide a formal description of FTs for reported value functions \( \hat{v}_i \). To do so, we represent \( \hat{v}_i \) as a vector \((\hat{v}(x))_{x \in \mathcal{X}}\). Each known FT is a change of basis and thus can be represented by a certain matrix \( F \in \{-1, 0, 1\}^{2^m \times 2^m} \) with the form:

\[
\hat{v}_i(y) = (F \hat{v}_i)(y) = \sum_{x \in \mathcal{X}} F_{y,x} \hat{v}_i(x).
\]

There is exactly one Fourier coefficient per bundle, this follows from the theory presented by Püschel and Wendler [2020]. The corresponding inverse transform \( F^{-1} \) is thus:

\[
\hat{v}_i(x) = (F^{-1} \hat{v}_i)(x) = \sum_{y \in \mathcal{X}} F_{x,y}^{-1} \hat{v}_i(y).
\]

\( \hat{v}_i \) is again a set function and we call \( \hat{v}_i(y) \) the Fourier coefficient at frequency \( y \). A value function is Fourier-sparse if \( |\text{ supp}(\hat{v}_i)| = |\{y : \hat{v}_i(y) \neq 0\}| \ll 2^m \). We call \( \text{ supp}(\hat{v}_i) \) the Fourier support of \( \hat{v}_i \).

Classically, the WHT is used as \( F \) [Bernasconi et al., 1996; O’Donnell, 2014], but we also consider two recently introduced FTs (FT3, FT4) due to their information-theoretic interpretation given in [Püschel and Wendler, 2020]:

**FT3:** \( F_{y,x} = (-1)^{|y|-|x|} \mathbb{I}_{\min(x,y)=x} \),

**FT4:** \( F_{y,x} = (-1)^{|\min(x,y)|} \mathbb{I}_{\max(x,y)=1_m} \),

**WHT:** \( F_{y,x} = \frac{1}{2^m} (-1)^{|\min(x,y)|} \).

Here, \( \min \) is the elementwise minimum (intersection of sets), \( \mathbb{I} \) is the indicator function, \( 1_m \) denotes the \( m \)-dimensional vector of 1s, and the indicator function \( \mathbb{I}_P \) is equal to 1 if the predicate \( P \) is true and 0 otherwise.
Notions of Fourier-sparsity. In recent years, the notion of Fourier-sparsity has gained considerable attention, leading to highly efficient algorithms to compute FTs [Stobbe and Krause, 2012; Amrollahi et al., 2019; Wendler et al., 2021]. Many classes of set functions are Fourier-sparse (e.g., graph cuts, hypergraph valuations and decision trees [Abraham et al., 2012]) and can thus be learned efficiently. The benefit of considering multiple FTs is that they offer different, non-equivalent notions of sparsity as illustrated by the following example.

Example 1. Consider the set of items $M = \{1, 2, 3\}$ and the associated reported value function $\tilde{v}_i$ shown in Table 1 (where we use 001 as a shorthand notation for $(0, 0, 1)$), together with the corresponding FCs $\phi_{0i}$: This bidder exhibits complementary effects for bundle containing more than one item, as can be seen, e.g., from $3 = \tilde{v}_i(110) > \tilde{v}_i(100) + \tilde{v}_i(010) = 2$ and $5 = \tilde{v}_i(111) > \tilde{v}_i(100) + \tilde{v}_i(010) + \tilde{v}_i(001) = 3$. Observe that while this value function is sparse in FT4, i.e., $\phi_{0i}(110) = \phi_{0i}(101) = \phi_{0i}(011) = 0$, it is neither sparse in FT3 nor WHT. Note that the coefficients $\phi_{0i}(100), \phi_{0i}(010), \text{and } \phi_{0i}(001)$ capture the value of single items and thus cannot be zero.

The induced spectral energy distributions for each FT, i.e., for each cardinality (i.e., number of items) $d$ from 0 to $m = 3$, we compute $\sum_{y \in X} |y|=d \phi_{0i}(y)^2 / \sum_{y \in X} \phi_{0i}(y)^2$, are shown in Table 2.

Table 2: Spectral energy in % for each cardinality (i.e., number of items) $d$ from 0 to $m = 3$ of all considered FTs.

<table>
<thead>
<tr>
<th>$d$</th>
<th>$d = 0$</th>
<th>$d = 1$</th>
<th>$d = 2$</th>
<th>$d = 3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT3</td>
<td>0.00</td>
<td>42.86</td>
<td>42.86</td>
<td>14.28</td>
</tr>
<tr>
<td>FT4</td>
<td>65.79</td>
<td>31.58</td>
<td>0.00</td>
<td>2.63</td>
</tr>
<tr>
<td>WHT</td>
<td>65.69</td>
<td>33.41</td>
<td>0.68</td>
<td>0.22</td>
</tr>
</tbody>
</table>

Table 1: Example with different induced notions of sparsity of all considered FTs.

Fourier-sparse approximations. In practice, $\tilde{v}_i$ may only be approximately sparse. Meaning that while not being sparse, it can be approximated well by a Fourier-sparse function $\hat{v}_i$. Formally, let $S_i = \text{supp}(\phi_{0i})$ with $|S_i| = k$, we call $\hat{v}_i(x) = \sum_{y \in S_i} F_{x,y}^{-1} \phi_{0i}(y)$ for all $x \in X$ (7) such that $\Vert \hat{v}_i - \tilde{v}_i \Vert_2$ is small a $k$-Fourier-sparse approximation of $\tilde{v}_i$. We denote the vector of FCs by $\phi_{0i}|_{S_i} = (\phi_{0i}(y))_{y \in S_i}$.

4 Fourier Transform-based WDPs

To leverage Fourier analysis for CA design, we represent bidders’ value functions using Fourier-sparse approximations. A key step in most auction designs is to find the social-welfare-maximizing allocation given bidder’s reports, which is known as the Winner Determination Problem (WDP). To apply FTs, we need to be able to solve the WDP efficiently. Accordingly, we next derive MIPs for each of the FTs.

For each bidder $i \in N$, let $\tilde{v}_i : \mathcal{X} \rightarrow \mathbb{R}_+$ be a Fourier-sparse approximation of the bidders’ reported value function $\tilde{v}_i$. Next, we define the Fourier transform-based WDP.

Definition 1. (Fourier Transform-based WDP)

$$\arg\max_{a \in \mathcal{F}} \sum_{i \in N} \tilde{v}_i(a_i).$$ (FT-WDP)

For $x, y \in \mathbb{R}^d$, let $x \leq y$, $\max(x, y)$ and $(-1)^x$ be defined component-wise, and let $\langle \cdot, \cdot \rangle$ denote the Euclidean scalar product. First, we formulate succinct representations of $\tilde{v}_i$.

Lemma 1. For $i \in N$ let $S_i = \{y^{(1)}, \ldots, y^{(k)}\}$ be the support of a $k$-Fourier-sparse approximation $\tilde{v}_i$ and $W_i \in \{0, 1\}^{k \times m}$ be defined as $W_i = \mathbb{1}_{y^{(k)}=1}$. Then it holds that

$$FT3: \tilde{v}_i(x) = \langle \phi_{0i}|_{S_i}, \max(0_0, 1 - W_i(1_m - x)) \rangle$$ (8)

$$FT4: \tilde{v}_i(x) = \langle \phi_{0i}|_{S_i}, \max(0_k, 1 - W_i x) \rangle$$ (9)

$$WHT: \tilde{v}_i(x) = \langle \phi_{0i}|_{S_i}, (-1)^{W_i x} \rangle.$$ (10)

See Appendix B.1 for the proof. With Lemma 1 and rewriting $\max(x, y)$ and $(-1)^x$ as linear constraints, we next encode (FT-WDP) as a MIP (see Appendix B.2 for the proof).

Theorem 1. (FT-based MIPs) Let $\tilde{v}_i : \mathcal{X} \rightarrow \mathbb{R}_+$ be a $k$-Fourier-sparse approximation from (8), (9), or (10). Then there exists a $C > 0$ s.t. the MIP defined by the objective

$$\arg\max_{a \in \mathcal{F}, \beta \in \{0, 1\}^k \in N} \sum_{i \in N} \langle \phi_{0i}|_{S_i}, \alpha_i \rangle,$$ (11)

and for $i \in N$ one set of transform specific constraints (12)–(14), or (15)–(17), or (18)–(20), is equivalent to (FT-WDP).

$$FT3: \quad s.t. \quad \alpha_i \geq 1_k - W_i (1_m - a_i)$$ (12)

$$\alpha_i \leq 1_k - W_i (1_m - a_i) + C\beta_i$$ (13)

$$0_k \leq \alpha_i \leq C(1_k - \beta_i)$$ (14)

$$FT4: \quad s.t. \quad \alpha_i \geq 1_k - W_i a_i$$ (15)

$$\alpha_i \leq 1_k - W_i a_i + C\beta_i$$ (16)

$$0_k \leq \alpha_i \leq C(1_k - \beta_i)$$ (17)

WHT: \quad s.t. \quad $\alpha_i = -2\beta_i + 1_k$

$$\beta_i = W_i a_i - 2\gamma_i$$ (19)

$$\gamma_i \in \mathbb{Z}^k$$ (20)

5 Analyzing the Potential of a FT-based CA

In this section, we experimentally evaluate the FTs and propose an FT-based allocation rule that motivates our practical hybrid ICA mechanism presented later in Section 6.

For our experiments, we use the spectrum auction test suite (SATS) [Weiss et al., 2017].1 SATS enables us to generate

---

1We used SATS version 0.6.4 for our experiments. The implementations of GSVM and LSVM have changed slightly in newer SATS versions. This must be considered when comparing the performance of different mechanisms in those domains.
5.1 Notions of Fourier Sparsity

We first experimentally show that different notions of FT lead to different types of sparsity in LSVM (for other domains see Appendix C.1). For this we first compute the FTs of all bidders and then calculate their spectral energy distribution. That is, for each cardinality $d$ (#items) from 0 to $m$, we compute $\sum_{y_i \in Y} \phi_{v_i}(y)^2 / \sum_{y_i \in Y} \phi_{v_i}(y)^2$. In Figure 1 we present the mean over 30 LSVM instances and bidder types.

Figure 1 shows that while the energy is spread among FCs of various degrees in FT3 and FT4, in WHT the low degree ($\leq 2$) FCs contain most of the energy, i.e., the WHT has much fewer dominant FCs that accurately describe each value function. As the WHT is orthogonal, learning low degree WHT-sparse approximations leads to low reconstruction error. Low degree WHT-sparse approximations can be learnt efficiently and accurately from a small number of queries using compressive sensing [Stobbe and Krause, 2012].

Note that the FT3 is identical to the classical polynomial value function representation [Lahaie, 2010] defined as

$$\hat{v}_{v_i}^{\text{poly}}(x) = \sum_{i=1}^{m} \sum_{j=(j_1, \ldots, j_l) \in M} x_{j_1} \cdots x_{j_l} c_j(i). \quad (21)$$

where the coefficient $c_j(i)$ is equal to the FT3 FC at frequency $y$ with $y_i = 1$ for $i \in \{j_1, \ldots, j_l\}$ and $y_i = 0$ else.\footnote{This can be seen by calculating the inverse in (4), i.e., $F_{y,x}^{-1} = f_{\min(x,y)=x}$, and plug $F_{y,x}^{-1}$ into (3).}

Thus, converting $\hat{v}_{v_i}^{\text{poly}}$ into another FT basis (here WHT) can indeed be very helpful for the design of ML-based CAs.

5.2 Reconstruction Error of Fourier Transforms

Next we validate the FT approach by comparing the reconstruction error of the FTs in the medium-sized GSVM and LSVM, where we can still compute the full FT (in contrast to MRVM). For now, we assume that we have access to bidders’ full $\hat{v}_{v_i}$. In Procedure 1, we determine the best $k$-Fourier-sparse approximation $\hat{v}_{\hat{v}_i}$ (see Appendix C.2 for details).

Procedure 1. (Best FCs Given Full Access to $\hat{v}_{v_i}$)

- Compute $\hat{v}_{\hat{v}_i}$ using the $k$ absolutely largest FCs $\phi_{v_{\hat{v}_i}}(S)$ from the full FT for each bidder’s reported value function $\phi_{v_{\hat{v}_i}} = F_{\hat{v}_{v_i}}$.

Remark 1. Since the WHT is orthogonal and the simulated auction data is noise-free, its approximation error is exactly equal to the residual of the FCs. Thus, Procedure 1 is optimal for the WHT. This is not the case for FT3 and FT4 because they are not orthogonal.

We then calculate the RMSE ($\frac{1}{|X|} \sum_{x \in X} (\hat{v}_{\hat{v}_i}(x) - \hat{v}_{\hat{v}_i}(x))^2)^{1/2}$ averaged over 100 instances and bidder types. In Table 3, we present the RMSEs for the three FTs and for NNs, where we used the architectures from Weissteiner and Seuken [2020].

For GSVM, we observe that we can perfectly reconstruct $\hat{v}_{\hat{v}_i}$ with the 200 best FCs, which shows that GSVM is 200-sparse. In contrast, LSVM is non-sparse, and we do not achieve perfect reconstruction with 200 FCs. Overall, we observe that the WHT outperforms FT3 and FT4. Moreover, we see that, if we could compute the $k$ best FCs of the WHT from $k$ training points, the WHT would outperform the NNs.

However, in practice, we do not have access to full value functions. Instead, we must use an algorithm that computes the best FCs using a reasonable number of value queries.

Remark 2. Thanks to its orthogonality the WHT has strong theoretical guarantees for sparse recovery from few samples using compressive sensing (see [Stobbe and Krause, 2012]). Thus, we focus on the WHT in the remainder of this paper.

5.3 A Fourier Transform-based Allocation Rule

We now present an FT-based allocation rule using the robust sparse WHT algorithm (RWHT) by Amrollahi et al. [2019]. RWHT learns a Fourier-sparse approximation $\hat{v}_{\hat{v}_i}$ of $\hat{v}_{v_i}$ from value queries. Procedure 2 finds the allocation $\hat{a}$.

Procedure 2. (WHT-Based Allocation Rule)

1. Use RWHT to compute $k$-sparse approximations $\hat{v}_{\hat{v}_i}$, $i \in N$.
2. Solve $\hat{a} \in \text{argmax} \sum_{i \in N} \hat{v}_{\hat{v}_i}(a_i)$ using Theorem 1.
A Practical Hybrid ICA Mechanism

In this section, we introduce and experimentally evaluate a practical hybrid ICA mechanism, based on FTs and NNs.

6.1 The Hybrid ICA Mechanism

The main issue of the FT-based allocation rule in Section 5.3 is the large number of queries, which we now address. The idea is the following: instead of directly applying a sparse FT algorithm (like RWHT) to bidders, we apply it to a NN-based representation. In this way, we query NNs instead of bidders. Based on the FCs of the NNs, we determine a Fourier-sparse approximation \( \tilde{v}_i \) with only few value queries, where the idea is that the FCs of each NN concentrate on the most dominant FCs of its respective value function. Indeed, recent evidence suggests that a NN trained by SGD can learn the Fourier-support [Rahaman et al., 2019]. We analyze our NN support discovery rule in Section 6.2. We now present HYBRID ICA, leaving details of the sub-procedures to Appendix D.1.

HYBRID ICA (Algorithm 1) consists of 3 phases: the MLCA, the Fourier reconstruction, and the Fourier allocation phase. It is parameterized by an FT \( \mathcal{F} \) and the numbers \( \ell_1, \ell_2, \ell_3, \ell_4 \) of different query types. In total, it asks each bidder \( \sum_{i=1}^{4} \ell_i \) queries: \( \ell_1 \) random initial, \( \ell_2 \) MLCA, \( \ell_3 \) Fourier reconstruction, and \( \ell_4 \) Fourier allocation queries.

1. **MLCA Phase.** We first run MLCA such that the NNs can then be trained on "meaningfully" elicited reports. In MLCA, we request reports for \( \ell_1 \) random initial bundles and for \( \ell_2 \) MLCA queries (Lines 1-2).

2. **Fourier Reconstruction Phase.** Next, we compute a Fourier-sparse approximation \( \tilde{v}_i \). For this, we first fit a NN \( \mathcal{N}_i \) on the reports \( R_i \) (Line 4). Then we compute the best FCs of the fitted NNs (Line 5, Procedure 3) in order to discover which FCs are important to represent the bidders. Based on these FCs, we determine \( \ell_3 \) Fourier reconstruction queries \( \hat{S}_i \) (Line 6, Procedure 4), send them to the bidders and fit \( \tilde{v}_i \) to the reports \( R_i \) received so far (Line 7, Procedure 5).

3. **Fourier Allocation Phase.** We use the fitted \( \tilde{v}_i \) to generate \( \ell_4 \) Fourier allocation queries. Here, we solve the FT-based WDP (Line 9) to get candidate queries \( q \), ensure that all queries are new (Lines 11–14), and set a total query budget of \( N \) (Line 15) and refit \( \tilde{v}_i \) (Line 16). Finally in Line 17, HYBRID ICA computes based on all reports \( R \) a welfare-maximizing allocation \( a_B^* \) and VCG payments \( p(R) \) (see Appendix A.2).

**Experiment Setup.** For HYBRID ICA and MLCA, we use the NN architectures from Weissteiner and Seuken [2020] and set a total query budget of 100 (GSVM, LSVM) and 500 (MRVM). For HYBRID ICA, we optimized the FTs and query parameters \( \ell_i \) on a training set of CA instances. Table 4 shows the best configurations.

<table>
<thead>
<tr>
<th>NN ARCHITECTURES</th>
<th>FT</th>
<th>( \ell_1 )</th>
<th>( \ell_2 )</th>
<th>( \ell_3 )</th>
<th>( \ell_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSVM</td>
<td>R: [32, 32]</td>
<td>N: [10, 10]</td>
<td>WHT</td>
<td>30</td>
<td>21</td>
</tr>
<tr>
<td>LSVM</td>
<td>R: [32, 32]</td>
<td>N: [10, 10, 10]</td>
<td>WHT</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>MRVM</td>
<td>L-R-N: [16, 16]</td>
<td>WHT</td>
<td>30</td>
<td>220</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4: Best configuration of HYBRID ICA. R: \([d_1, d_2] \) denotes a 3-hidden-layer NN for the regional bidder with \( d_1 \) and \( d_2 \) nodes.

6.2 NNs Support Discovery Experiments

In HYBRID ICA we use the NNs for support discovery where it is key that the FCs of these NNs concentrate on the dominant FCs of its value function, i.e. \( \text{supp}(\phi_{S_i}) \approx \text{supp}(\phi_{i}) \).

To evaluate the NN-based support discovery (Line 5), we consider the spectral energy ratio obtained by dividing the spectral energies of the \( k \) frequencies selected from the NN and the \( k \) best frequencies (for the WHT the best FCs are the ones with the largest absolute value). Formally, for each bidder \( i \), let the \( k \) frequencies selected from the NN be \( S_i = \{y^{(1)}, \ldots, y^{(k)}\} \) and the best ones be \( \hat{S}_i = \{y^{(1)}, \ldots, y^{(k)}\} \). Then, bidder \( i \)'s energy ratio is given by \( \sum_{\phi \in S_i} \phi_i(y)^2 / \sum_{\phi \in \hat{S}_i} \phi_i(y)^2 \in [0, 1] \) (see Appendix D.2 for details). This ratio is equal to one if \( \hat{S}_i = S_i \). Figure 3
Third, it distributes the welfare more evenly (= fairer) to bidding queries (training NNs and solving the NN-based MIP). This approach leads to higher efficiency, a computational speedup, and a fairer distribution of social welfare than state-of-the-art.

Fourier-based queries. As we leveraged this to design a new hybrid ICA that uses NN and Fourier-based allocation queries practically feasible. We have shown that our Fourier-based auction is especially powerful in sparse domains. In practice, bidders are often composed of larger bundles (i.e., 17 items c.p. to 4 in MLCA queries) and thus allocate large bundles to bidders that would have been overlooked. In MRVM, the optimal query split for HYBRID ICA uses $\ell_3 = 0$ Fourier reconstruction queries such that HYBRID ICA is equal to HYBRID ICA (NO FR). Thus, in MRVM, HYBRID ICA’s increased efficiency and fairer distribution results from the Fourier allocation queries.

Overall, we see that our Fourier-based auction is especially powerful in sparse domains. In practice, bidders are often limited by their cognitive abilities [Scheffel et al., 2012] or use a low-dimensional computational model to represent their value function. Thus, their reported preferences typically exhibit only a limited degree of substitutability and complementarity, which is captured well by Fourier-sparcity.

### 6.3 Efficiency Experiments

We now evaluate the efficiency of HYBRID ICA vs MLCA.

**Results.** Table 5 contains our main results in all domains. We show efficiency, distribution of efficiency to bidder types, revenue ($\sum_{i \in N} p(R_i)/v(\alpha)$), and runtime. First, we see that HYBRID ICA statistically significantly outperforms MLCA w.r.t. efficiency in GSVM and MRVM and performs on par in LSVM. Second, it also leads to a computational speedup ($\times 6$ GSVM, $\times 3$ LSVM, $\times 2$ MRVM). The reason for this computational speedup is that the generation of the $\ell_3 + \ell_4$ Fourier queries (estimating the superset of the support using RWHT on the NNs, fitting the FT models using compressive sensing and solving our new FT-based MIPs) is faster than the generation of the NN-based MLCA allocation queries (training NNs and solving the NN-based MIP). Third, it distributes the welfare more evenly (= fairer) to bidder types. This also leads to a distribution that more closely resembles that of the efficient allocation (see Efficient Allocation). We present full efficiency path plots for the different phases of HYBRID ICA in Appendix D.3.

**Fourier queries.** To verify the importance of the $\ell_3$ Fourier reconstruction and $\ell_4$ Fourier allocation queries, we also present HYBRID ICA (NO FR) and HYBRID ICA (NO FR/FA), which use random queries in place of the $\ell_3$ Fourier reconstruction and the $\ell_3 + \ell_4$ Fourier-based queries. As we see in Table 5, using the Fourier queries leads to significantly better efficiency and HYBRID ICA (NO FR/FA) does not achieve a fairer efficiency distribution. A comparison of HYBRID ICA to HYBRID ICA (NO FR) reveals that, in GSVM and LSVM, the Fourier reconstruction queries cause the fairer distribution. We empirically verified that these queries are statistically significantly outperforming the standard notion of egalitarian social welfare.

Table 5: HYBRID ICA vs. MLCA, HYBRID ICA (NO FR), and HYBRID ICA (NO FR/FA). All results are averages over a test set of 100 (GSVM and LSVM) and 30 (MRVM) CA instances. For efficiency we give a 95% confidence interval and mark the best mechanisms in grey.

![Image of Table 5](image-url)

**Figure 3:** Average energy ratio (y-axis) with 97.5% and 2.5% empirical quantiles for a number of selected frequencies $k$ (x-axis) over 30 instances in GSVM and LSVM and over 5 instances in MRVM. This shows that the NN-based supports are almost on par with the best supports given a fixed budget of $k$ frequencies.

7 Conclusion

We have introduced Fourier analysis for the design of CAs. The main idea was to represent value functions using Fourier-sparse approximations, providing us with a new lens on bidder’s values in the Fourier domain.

On a technical level, we have derived succinct MIPs for the Fourier transform-based WDPs, which makes computing Fourier-based allocation queries practically feasible. We have leveraged this to design a new hybrid ICA that uses NN and Fourier-queries. Our experiments have shown that our approach leads to higher efficiency, a computational speedup and a fairer distribution of social welfare than state-of-the-art.

With this paper, we have laid the foundations for future work leveraging Fourier analysis for representing and eliciting preferences in combinatorial settings.

### Acknowledgments

We thank the anonymous reviewers for helpful comments. This paper is part of a project that has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (Grant agreement No. 805542). Furthermore, this material is based upon work supported by the National Science Foundation under grant no. CMMI-1761163.

### References


[Amrollahi et al., 2019] A. Amrollahi, A. Zandieh, M. Kapralov, and A. Krause. Efficiently learning...


