Clock Auctions: Allocation-Based Characterization, Computational Complexity, and Economic Efficiency

HANRUI ZHANG, Chinese University of Hong Kong, China

Clock auctions are a natural class of simple auction mechanisms, with numerous desirable properties including obvious strategyproofness, credibility, and unconditional winner privacy. These properties make clock auctions an ideal solution for real-world allocation problems, such as radio spectrum (re)allocation. Accordingly, significant effort has been made in understanding the performance (and in particular, economic efficiency) of clock auctions. In this paper, we make progress along this direction on multiple fronts.

Computationally, we investigate two natural problems: implementation and optimization. The former asks to find a clock auction protocol that implements a particular input allocation function whenever there exists one, and declare that this is impossible otherwise; the latter asks to find a clock auction protocol that maximizes social welfare among all clock auction protocols on a particular input prior distribution. We give an efficient algorithm for the implementation problem, and show that the optimization problem is NP-complete. To our knowledge, these are the first results regarding the computational complexity of clock auctions. En route to these results, we develop a complete characterization of allocation functions that can be implemented using clock auctions, which may be of independent interest.

Information-theoretically, we present a framework connecting the economic efficiency of clock auctions to a much cleaner problem that we call "upper tail extraction". In particular, the inexistence of constant-factor clock auctions for upper tail extraction would immediately imply a super-constant efficiency gap for clock auctions. On the other hand, the existence of constant-factor clock auctions for upper tail extraction would imply approximate efficiency in an important class of instances, strongly suggesting approximate efficiency in general. We then construct constant-factor clock auctions for upper tail extraction in the special case with iid agents, which, through our framework, implies approximate efficiency of clock auctions with independent groups of agents that are each homogeneous. This removes the immediate technical obstacle to unconditional approximate efficiency identified in prior work, and showcases the power of our framework.

CCS Concepts: • **Theory of computation → Algorithmic game theory and mechanism design**.

Additional Key Words and Phrases: Mechanism Design, Clock Auctions, Simple Mechanisms, Computational Complexity

ACM Reference Format:
Hanrui Zhang. 2025. Clock Auctions: Allocation-Based Characterization, Computational Complexity, and Economic Efficiency. In *The 26th ACM Conference on Economics and Computation (EC '25), July 7–10, 2025, Stanford, CA, USA*. ACM, New York, NY, USA, 26 pages. https://doi.org/10.1145/3736252.3742493

1 Introduction

Clock auctions are a natural class of simple auction mechanisms. A clock auction proceeds roughly in the following way: Suppose for concreteness that an auctioneer wants to sell k identical items to $n > k$ buyers. Each buyer i is interested in at most one item, and is willing to pay at most v_i for it. The auctioneer maintains a vector p of *clock prices*, one p_i for each buyer i. These prices are

Author's Contact Information: Hanrui Zhang, hanrui@cse.cuhk.edu.hk, Chinese University of Hong Kong, Hong Kong, China.

EC '25, Stanford, CA, USA

ACM ISBN 979-8-4007-1943-1/2025/07
https://doi.org/10.1145/3736252.3742493

generally 0 at the beginning of the auction, and they keep rising until the auction terminates. Each buyer i reacts to the rising price p_i, by *quitting* as soon as p_i exceeds v_i. The auction terminates once there are precisely k buyers left, at which point each remaining buyer i receives an item, and pays the current price p_i faced by them. The freedom of the auctioneer lies in the fact that they can raise the clock prices in any way they like, possibly depending on which buyers have quit at which times.

Clock auctions are known to be *obviously strategyproof* [Li, 2017]: In principle, each buyer could choose to quit at any time, but the "obviously rational" strategy is to quit when p_i exceeds v_i, as prescribed by the auction mechanism. In addition, they are also *credible* [Akbarpour and Li, 2020], meaning that the auctioneer has no incentive to deviate from the rules of the auction. In fact, clock auctions even guarantee *unconditional winner privacy* [Brandt and Sandholm, 2005, Milgrom and Segal, 2020], which ensures that the winners only need to release as much information as required to "prove they should win". As argued by Milgrom and Segal [2020], all these properties make clock auctions an ideal solution for real-world high-stakes (binary) allocation problems. This renders one question particularly important: How can we design *good* clock auctions? In fact, since there are numerous reasonable objectives in auction design, let us be more specific and focus on one of the most natural and important objectives: *social welfare*. The question we aim to answer in this paper, therefore, is:

(How) can we design clock auctions that (approximately) maximize social welfare?

The nature of the above question is twofold. The first aspect concerns what we can *ideally* achieve using clock auctions. It is known that clock auctions generally cannot achieve the first-best welfare [Dütting et al., 2017, Feldman et al., 2022], even if we disregard all the computational issues to be discussed below. One would then naturally turn to approximation and investigate the following question: What is the worst-case gap between the best welfare achievable using clock auctions, and the first-best? We will refer to the former as the *clock-best* welfare.[1] Indeed, this question has already received considerable attention. To quickly summarize what we currently know: In the prior-free setting, there is a deterministic clock auction that achieves $O(\log n)$-approximation (where n is the number of buyers or agents) against the first-best [Christodoulou et al., 2022], which matches a lower bound for the same setting [Dütting et al., 2017]. If we allow either randomization or access to prior distributions, then the ratio can be improved to $O(\log \log k)$ [Feldman et al., 2022], where k is the number of "maximal feasible sets" (formally defined later). It remains open if the clock-best is always within a *constant* factor of the first-best in the latter (i.e., Bayesian) setting.

The second aspect concerns what we can *practically* achieve using clock auctions. Given a problem instance (comprising a set of buyers or agents, a type space, feasibility constraints, and a prior distribution, all of which will be formally defined below), efficiently *computing* a clock auction that achieves the clock-best welfare appears just as nontrivial, despite the fact that by definition, there always exists one such auction. It is known that in practice, carefully designed heuristics perform remarkably well [Milgrom and Segal, 2020, Newman et al., 2024]. However, it is unclear whether they offer any nontrivial worst-case guarantee. On the other hand, the constructions discussed above that provably approximate the first-best welfare can all be computed efficiently, but they do not seem to provide better guarantees (in any obvious way) against the clock-best than they do against the first-best. So, despite all the efforts made so far, the computational complexity of the clock-best is still wide open. In fact, taking a step back, given a particular way of allocating items to agents, i.e., an *allocation function*, it is not even clear whether we can efficiently construct a clock auction that implements this allocation function, or declare that it is impossible to do so.

[1]This might appear somewhat ambiguous since we consider both prior-free and Bayesian settings, but here we intend to keep the discussion informal. We will explain further where needed.

1.1 Our Results

In this paper, we focus on single-parameter agents and binary allocation problems. Roughly speaking, these are environments where each agent can either be served or not, and the type of each agent is a single nonnegative number capturing how much the agent values being served. Moreover, the mechanism is constrained in terms of which subsets of agents can be served simultaneously.[2] This is the predominant setting in which clock auctions are studied. We make progress on both fronts discussed above, concerning the gap between the clock-best and the first-best, and that between efficiently computable clock auctions and the clock-best, respectively.

Computational complexity of clock auctions. On the computational front, we consider two concrete problems:

- Implementation: given a particular way of allocating items to agents (i.e., an allocation function), construct a clock auction that implements this allocation function, or declare it is impossible to do so. For this problem, we present an efficient (i.e., polynomial-time) algorithm (Corollary 1, which combines Lemma 2, Algorithm 1, and Theorem 2).
- Optimization: given a problem instance, find the clock auction that maximizes social welfare in expectation. We show this problem is NP-complete (Theorem 3), which roughly means one cannot efficiently find a clock auction that is good enough — or tell if one exists — unless P = NP; however, if there in fact is a good clock auction, then in hindsight, one can easily "prove" its existence. Note that this computational hardness is fundamentally different from the kind discussed by Milgrom and Segal [2020] and Newman et al. [2024]: The kind of hardness they face originates directly from a classical NP-hard problem embedded in the specific feasibility constraints arising from spectrum reallocation, while we show that optimizing clock auctions remains hard even for extremely simple feasibility constraints (i.e., two disjoint maximal feasible sets, one of which being a singleton). In other words, our result establishes "endogenous" hardness of clock auctions by themselves, independent of "exogenous" hardness embedded in feasibility constraints.

To our knowledge, our results are the first to study the (endogenous) computational complexity of clock auctions.

Allocation-based characterization of clock auctions. A crucial technical ingredient of our computational results is a complete characterization of allocation functions that can be implemented using clock auctions (Theorem 1), which we believe is of independent interest.[3] The characterization itself requires a few definitions to be introduced later, so here we only discuss how it enables our computational results.

For implementation, if the input allocation function can be implemented using clock auctions, then our characterization guarantees that in any reasonable "state" that might be reached during a clock auction, there always exists an agent whose clock price we can safely raise without causing conflicts with the target allocation function. Given this fact, the algorithm proceeds in the natural way: At each time, we pick an arbitrary agent whose price we can safely increase, and increase it to the next critical value. Then, either we eventually reach a state where all agents that are still active (possibly an empty set) should be served, in which case we serve them and terminate the auction; or at some point no agent's price can be safely raised, and we can declare the allocation

[2] For example, "no more than k agents can be served simultaneously" means a subset of agents can be served simultaneously iff its cardinality is at most k.

[3] To build a consistent technical flow, in later sections, we will present the characterization first, and the computational results after that.

function not implementable because we have found a witness that the allocation function does not satisfy our characterization.

For optimization, we construct a family of instances that encode the 3-SAT problem, which is one of the most famous NP-complete problems. Here, our allocation-based characterization enforces the semantics of the 3-SAT instance, making sure that if we "choose" a type vector corresponding to a clause, then we also must choose at least one literal it contains. In light of the above, we view the characterization as a technically more tangible definition of clock auctions. As such, we believe it will prove useful in other technical contexts concerning clock auctions.

A framework for bounding the clock-best against the first-best. As for the gap between the clock-best and the first-best, the main open question is the (in)existence of constant-factor clock auctions in the Bayesian setting, which is closely related to randomized clock auctions in the prior-free setting. We focus on the former in this paper for simplicity. Our first result is a framework for designing constant-factor approximations, and / or proving impossibility results thereof (Lemma 3). The framework applies to what we call independent groups, where all maximal subsets that can be simultaneously served are independent in terms of the types of the agents in each maximal subset. This subsumes an important class of instances identified by Feldman et al. [2022], namely "Disjoint-Maximal-Sets" (they consider independent agents by default whereas we allow correlation within groups), which appears hard enough to resist better approximations than what is known for the general case. In particular, the lower bound constructed by Feldman et al. [2022] lies within this class.

Within the class of independent groups, our framework establishes equivalence between bounding the clock-best and the much cleaner problem of "upper tail extraction", which roughly asks to accurately identify realizations of the type vector whose ℓ_1 norm is large enough using clock auctions.[4] Technically, the equivalence builds on ideas similar to those used in the study of prophet inequalities, e.g., an ex-ante relaxation of the prophet benchmark and the fact that the two quantities are within a constant factor of each other. Conceptually, the framework identifies a much more restrictive and seemingly tractable problem, such that (1) the inexistence of constant-factor clock auctions for upper tail extraction would immediately imply a super-constant gap between the clock-best and the first-best,[5] and (2) the existence of constant-factor clock auctions for upper tail extraction would imply a constant gap in the case of independent groups, which would be strong evidence that the gap is constant in general.

Constant-factor clock auctions for independent homogeneous groups. Utilizing the power of our equivalence framework, we construct constant-factor clock auctions for independent groups that are each "homogeneous", i.e., agents in each group have iid types (Corollary 2). This in particular moves the immediate technical obstacle to constant-factor clock auctions identified by Feldman et al. [2022] out of the way. Technically, we take a "hard" approach, as opposed to "softer" ones which appear more often in the study of prophet inequalities and simple mechanisms. We consider clock auctions using a uniform price, and show that there is a price good enough to approximately extract any desired upper tail; then we invoke our framework to translate this into a constant-factor clock auction for welfare maximization. To establish the existence of a good price, we lower bound and upper bound various tail probabilities. Part of this is done through careful manipulation of the moment generating function to derive a new Chernoff-Hoeffding-style inequality. Our construction

[4]Feldman et al. [2022] also hint at the possibility of such an equivalence (or at least one direction thereof) in the special case of "Disjoint-Maximal-Sets with equal size maximal sets and iid bidders". We formalize and generalize this idea, and in the process handle a number of technical issues left out of their informal discussion.

[5]Such a gap in the Bayesian setting would immediately imply a similar super-constant gap in the prior-free setting.

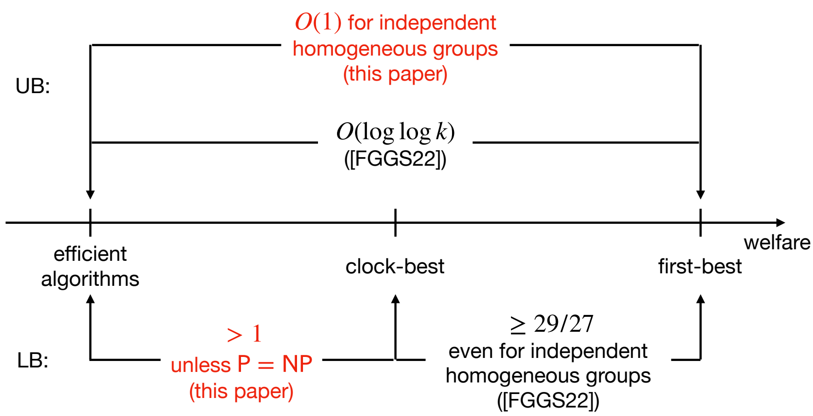

Fig. 1. Best known bounds on economic efficiency of Bayesian clock auctions.

also proves a technical conjecture by Feldman et al. [2022], which connects the clock-best-vs-first-best question to large deviations theory. In fact, here we need to handle not only large deviations, but also "moderate" ones.

We summarize the best known bounds for Bayesian clock auctions in Figure 1.

1.2 Further Related Work

There is a long line of work on the power of clock auctions in binary allocation problems. As briefly discussed above, Dütting et al. [2017] establish a logarithmic lower bound for deterministic clock auctions in prior-free settings, which holds even under knapsack feasibility constraints. They also give an efficient construction of deterministic clock auctions that achieve $O(\log n)$-approximation udner the same class of feasibility constraints, where n is the number of agents. Christodoulou et al. [2022] generalize Dütting et al.'s construction and give an efficient construction that achieves $O(\log n)$-approximation for arbitrary downward-closed feasibility constraints. Moreover, they show the possibility of $O(\sqrt{\log n})$-approximation in prior-free settings using randomized clock auctions. Feldman et al. [2022] achieve significant improvement along this line by giving an efficient construction that achieves $O(\log\log k)$-approximation, both with randomization and in the Bayesian setting, where k is the number of "maximal feasible sets".

Beyond the more traditional prior-free and Bayesian settings, Gkatzelis et al. [2021] study a prior-free setting with interdependent types, where they establish parametrized bounds in terms of both revenue and welfare. Gkatzelis et al. [2024] study clock auctions with unreliable advice in the prior-free setting, where they give constructions that achieve $O(\log n)$-approximation in the worst case, and much better guarantees when the advice turns out to be informative. Variants of clock auctions have also been used in budget-feasible mechanism design [Balkanski et al., 2022] and two-sided markets [Loertscher and Marx, 2020]. In particular, combinatorial clock auctions, which are a two-stage generalization of classical clock auctions, have received considerable attention [Ausubel and Baranov, 2017, Ausubel et al., 2006, Bousquet et al., 2016, Janssen and Kasberger, 2019, Levin and Skrzypacz, 2016].

Another related line of research is that of simple mechanisms. A series of work (see, e.g., [Ferraioli and Ventre, 2023, Ferraioli et al., 2021, Li, 2017, Milgrom and Segal, 2020]) connects clock auctions to greedy algorithms, and further to obvious strategyproofness, which is an important notion capturing strategic simplicity. Here, we refrain from a prolonged discussion, in particular since this direction is orthogonal to the focus of this paper. Technically, our results are also related to the study of prophet inequalities (see, e.g., [Correa et al., 2019, Cristi and Ziliotto, 2024, Hajiaghayi et al., 2007, Kleinberg and Weinberg, 2012, Lucier, 2017]) and posted-price mechanisms (see, e.g., [Alaei, 2014, Banihashem et al., 2024, Chawla et al., 2010, Correa et al., 2024, Feldman et al., 2014]). At a high level, clock auctions are more powerful (and accordingly, less simple) compared to posted-price mechanisms. As such, clock auctions achieve better welfare and / or revenue guarantees in many natural settings, including ones studied in this paper. We will discuss the specific technical connections in detail where appropriate.

2 Preliminaries

Agents, types, allocation functions, prior distributions, and feasibility constraints. There are n single-parameter agents $N = [n]$. Each agent i has a type $v_i \in \mathbb{R}_+$, representing how much they value receiving an item. Let $v = (v_1, \ldots, v_n) \in \mathcal{V}$ denote the vector of all agents' types, where $\mathcal{V} \subseteq \mathbb{R}_+^n$ is the joint type space. Note that $\mathcal{V}$ is not necessarily a product space. Throughout the paper, we assume $m = |\mathcal{V}| < \infty$ for simplicity and compactibility with computational considerations.[6] For each $i \in [n]$, let $\mathcal{V}_i = \{v_i \mid \exists v_{-i} \in \mathbb{R}_+^{n-1} : (v_i, v_{-i}) \in \mathcal{V}\}$, i.e., $\mathcal{V}_i$ is the set of all types i can have. Moreover, let $\bar{\mathcal{V}} = \mathcal{V}_1 \times \cdots \times \mathcal{V}_n$, which we will refer to as the extended type space. For any two vectors (including type vectors and price vectors to be introduced later) $v, v' \in \mathbb{R}_+^n$ and $S \subseteq [n]$, we say $v \geq_S v'$ iff $v_i \geq v'_i$ for all $i \in S$, and $v_i = v'_i$ for all $i \in [n] \setminus S$. We say $v \geq v'$ if $v \geq_{[n]} v'$. We define $\leq_S$, etc., in the same way.

An allocation function $\alpha : \mathcal{V} \to 2^{[n]}$ maps each vector of types to an allocation, represented by the set of agents who receive an item. When dealing with the optimization problem of finding welfare-maximizing clock auctions, we will consider a Bayesian setting with nontrivial feasibility constraints. This involves a prior distribution $\mathcal{D} \in \Delta(\mathcal{V})$. For each $v \in \mathcal{V}$, we let $\mathcal{D}(v) \in [0, 1]$ be the probability that the agents' types are v. In addition, there are feasibility constraints described by the family of feasible sets $\mathcal{F} \subseteq 2^{[n]}$. We assume $\mathcal{F}$ is downward-closed, meaning that if some $S \in \mathcal{F}$, then for any subset $T \subseteq S$, $T \in \mathcal{F}$. Let k be the number of maximal sets in $\mathcal{F}$, which measures the richness of $\mathcal{F}$. Given $\mathcal{F}$, an allocation function α is feasible iff $\alpha(v) \in \mathcal{F}$ for all $v \in \mathcal{V}$. We will not need to explicitly deal with payments in this paper.

Clock auctions. We adopt the formulation used by Milgrom and Segal [2020], with only minor notational changes. A clock auction proceeds in discrete time periods $t = 1, 2, \ldots$. The set of active agents at time t is denoted by $A_t \subseteq [n]$. A history H of length t consists of a sequence of subsets of agents $H = (A_1, \ldots, A_t)$, where $A_t \subseteq A_{t-1} \subseteq \cdots \subseteq A_1$. For two histories H and H', we say $H \sqsubseteq H'$ if H is a prefix of H'. Let $\mathcal{H}_t$ be the family of all possible length-t histories, and $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2 \cup \ldots$ be the set of all finite-length histories. A clock auction is specified by a clock price $p : \mathcal{H} \to \mathbb{R}_+^n \cup \{\perp\}$, where for any $H, H' \in \mathcal{H}$ where $H \sqsubseteq H'$, $p(H) \leq p(H')$. Here, $\perp$ means "termination": the auction terminates and the agents who are still active each receive an item. One of the purposes of introducing termination is to decouple allocation from any feasibility constraints.

A clock auction p describes an allocation function α_p — a partial allocation function, to be precise, because it may not always terminate — in the following way:

- Fix a type vector $v \in \mathcal{V}$. Initially, all agents are active, i.e., $A_1 = [n]$.
- At each time $t = 1, 2, \ldots$:

[6] While we believe our characterization holds more generally, it is not the goal of this paper to deal with subtleties introduced by continuity.

– If $p(H_t) \neq \perp$, then the price vector (weakly) rises to $p(H_t)$, where $H_t = (A_1, \ldots, A_t)$. Each agent i whose type is still no smaller then the respective price remains active, i.e., $A_{t+1} = \{i \in [n] \mid v_i \geq p_i(H_t)\}$.
– Otherwise, the auction terminates, and the allocation at v is $\alpha_p(v) = A_t$.

We say a clock auction p implements an allocation function α on $\mathcal{V}$, iff α_p and α are both defined on $\mathcal{V}$ and $\alpha_p(v) = \alpha(v)$ for all $v \in \mathcal{V}$. An allocation function α is implementable by clock auctions on $\mathcal{V}$ iff there exists a clock auction p that implements α on $\mathcal{V}$.

3 Allocation-Based Characterization

In this section, we present our characterization of allocation functions implementable using clock auctions. This will serve as a technical building block of our computational results. We first introduce a few essential definitions.

Definition 1 (free-riding). Given a type space $\mathcal{V}$ and an allocation function $\alpha : \mathcal{V} \to 2^{[n]}$, we say that agent $i \in [n]$ free-rides a subset of agents $S \subseteq [n] \setminus \{i\}$ at $u \in \bar{\mathcal{V}}$, iff there exists $v \in \mathcal{V}$ such that (1) $v \geq_S u$ and (2) $i \in \alpha(v)$.

The term "free-riding" comes from the following intuitive interpretation: When i free-rides S at u, if we view u as "bids", then i might become a winner without doing anything when agents in S raise their bids. In other words, it is possible for i to become a winner by free-riding S.

Definition 2 (free-riding group). Given a type space $\mathcal{V}$ an allocation function $\alpha : \mathcal{V} \to 2^{[n]}$, we say $S \subseteq [n]$ is a free-riding group at $u \in \bar{\mathcal{V}}$, iff each $i \in S$ free-rides $S \setminus \{i\}$ at u.

In words, in a free-riding group, every agent free-rides the rest of the group. Given the above definitions, we are ready to state our characterization.

Theorem 1. *An allocation function α is implementable by clock auctions on $\mathcal{V}$ iff the following condition holds: If $S \subseteq [n]$ is a free-riding group at $u \in \bar{\mathcal{V}}$, then $S \subseteq \alpha(v)$ for all $v \in \mathcal{V}$ where $v \geq_S u$.*

We break the proof of Theorem 1 into two parts, necessity (Lemma 1) and sufficiency (Lemma 2). We show necessity first.

Lemma 1. *If an allocation function α is implementable by clock auctions on $\mathcal{V}$, then the following condition holds: if $S \subseteq [n]$ is a free-riding group at $u \in \bar{\mathcal{V}}$, then $S \subseteq \alpha(v)$ for all $v \in \mathcal{V}$ where $v \geq_S u$.*

Then we present a constructive argument (through Algorithm 1) for sufficiency.

Lemma 2. *There is an efficient algorithm that computes the clock prices that implement an allocation function α on $\mathcal{V}$ if the following condition holds: If $S \subseteq [n]$ is a free-riding group at $v \in \bar{\mathcal{V}}$, then $S \subseteq \alpha(v')$ for all $v' \in \mathcal{V}$ where $v' \geq_S v$.*

We defer the proofs of Lemma 1 and Lemma 2 to Appendix A. Instead, here we provide some intuition behind the characterization: Imagine we are running a clock auction. Suppose at some point, the set of active agents is A, and some agent i free-rides all other active agents $S = A \setminus \{i\}$ at u, where u is the current clock prices (or types for inactive agents). Then given everything we currently know, it might be the case that the true types v satisfy $v \geq_S u$ and $i \in \alpha(v)$, which means we cannot raise i's clock price, because we would run the risk of forcing i to quit while i should actually be served according to α. Now further suppose the set of active agents A is a free-riding group at u. Then, by the same logic, we cannot raise the clock price of any active agent $i \in A$, which leaves us with no choice but to terminate the auction and serve everyone in A. Alternatively, if A is not a free-riding group at u, then we must be able to find some active agent i which does not free-ride the other active agents, and we can safely raise i's clock price without potentially violating the allocation dictated by α. This is the intuition behind the constructive direction of the characterization, i.e., Algorithm 1.

ALGORITHM 1: An algorithm that computes the clock prices that implements any allocation function implementable by clock auctions.

```
Input: type space 𝒱, allocation function α, history H = (A_1, ..., A_t) of some length t ≥ 0.
Output: clock prices p(H) that implements α on 𝒱.
1  if t = 1 then
2      for i ∈ [n] do
3          let u_i ← min 𝒱_i;
4      end
5  end
6  else
7      let u ← p(A_1, ..., A_{t-1});
       /* p(A_1, ..., A_{t-1}) has been computed before time t and only needs to be retrieved */
8  end
9  for i ∈ [n] \ A_t do
10     u_i ← max{v ∈ 𝒱_i | v < u_i};
11 end
12 if A_t ⊆ α(v) for all v ≥_{A_t} u then
13     return ⊥;
14 end
15 else
16     if A_t \ {i ∈ A_t | i free-rides A_t \ {i} at u} = ∅ then
17         return error;
18     end
19     else
20         let i* ← min A_t \ {i ∈ A_t | i free-rides A_t \ {i} at u};
           /* such an i* always exists when (1) α satisfies the condition in Theorem 1,
              and (2) the if-condition for termination in Line 12 does not hold       */
21     end
22     if u_{i*} = max 𝒱_{i*} then
23         let u_{i*} ← ∞;
24     end
25     else
26         let u_{i*} ← min{v ∈ 𝒱_{i*} | v > u_{i*}};
27     end
28     for i ∈ [n] \ A_t do
29         u_i ← min{v ∈ 𝒱_i | v > u_i};
30     end
31     return u;
32 end
```

4 Computational Complexity of Clock Auctions

With the allocation-based characterization in hand, we are ready to discuss the computational complexity of clock auctions. We will discuss two specific computational problems of interest: checking whether a given allocation function is implementable by clock auctions, and computing the welfare-maximizing allocation function that can be implemented by clock auctions given a prior distribution.

Input encoding. In order to discuss computational complexity, we first need to specify the way a problem instance is encoded. In the problem of checking implementability, an input instance consists of the number of agents n, the type space $\mathcal{V}$ and the allocation function $\alpha : \mathcal{V} \to 2^{[n]}$ to be checked. We assume these components are given in the natural way: an input instance begins with integers n and m, followed by $m = |\mathcal{V}|$ vectors, each in $\mathbb{R}^n$. The j-th of these vectors specifies the j-th possible type vector in the type space. Then we have m sets, each of which is a subset of $[n]$, corresponding to the allocation given by α for each possible type vector. One can check that the total length of the input instance is polynomial in n and m.

In the problem of welfare maximization, an input instance consists of the number of agents n, the type space $\mathcal{V}$, the prior distribution $\mathcal{D} \in \Delta(\mathcal{V})$, and the family of feasible sets $\mathcal{F}$. Again we use the natural representation: an input instance begins with integers n, m and k, followed by $m = |\mathcal{V}|$ vectors, each in $\mathbb{R}^n$. Then for $\mathcal{D}$, we have m nonnegative numbers representing the probability of each type vector.[7] As for $\mathcal{F}$, we assume it is specified by all maximal feasible sets, since $|\mathcal{F}|$ is typically very large. That is, $\mathcal{F}$ is given by k subsets of $[n]$ (with each subset represented, for example, by a binary string of length n), where k is the number of maximal sets in $\mathcal{F}$. Again observe that the total length of the input is polynomial in n, m, and k. We remark that the above conventions are not unique, and any reasonable encoding of polynomial length would lead to the same complexity results.

We first consider the problem of checking implementability, and show that this can be done in polynomial time.

Theorem 2. *There is a polynomial-time algorithm for checking whether a given allocation function is implementable by clock auctions.*

Proof. We will establish the theorem essentially as a corollary of Theorem 1 and Algorithm 1. In fact, any algorithm that computes clock prices with the properties used in the proof would suffice for the purpose of establishing Theorem 2. Here we use Algorithm 1 for concreteness. Observe that unless Algorithm 1 returns `error`, it always generates clock prices that are feasible (i.e., weakly increasing) under all circumstances. Moreover, these prices faithfully implement the input allocation function whenever it is implementable by clock auctions. We thus have the following algorithm that checks whether any given α is implementable by clock auctions in polynomial time: For each $v \in \mathcal{V}$, run a clock auction with prices given by Algorithm 1. This entire procedure takes time polynomial in n and m. If there exists some $v \in \mathcal{V}$ on which Algorithm 1 returns `error`, or if the auction terminates with an allocation different from $\alpha(v)$, then α is not implementable by clock auctions. Otherwise, α is implementable by clock auctions, because we have computed the clock prices that faithfully implement α pointwise on $\mathcal{V}$. □

Note that the polynomial dependency on $m = |\mathcal{V}|$, which can be exponential in the number of agents n, is generally unavoidable: m is the length of the input allocation function α in its flat representation (i.e., $\alpha(v)$ for each $v \in \mathcal{V}$), so merely reading α would take time linear in m. While there are natural succinct representations of $\mathcal{V}$ (e.g., as the product of each agent's marginal type space), it is unclear how α can be represented in a similar way. So, unless we focus on very specific

[7]Strictly speaking, we also need to specify how each number is given, and one natural way is to encode each number as a fraction where the numerator and the denominator are integers with a certain number of digits. Our complexity results hold for any reasonable choice of representation. Also, we do not necessarily require these numbers to be normalized, since it makes no material difference for the welfare maximization problem. So in principle, one could also encode $\mathcal{D}$ by assigning an integral weight to each possible type vector, which has a natural representation.

cases (where α belongs to a highly restrictive class with a natural succinct representation), the polynomial dependency on m is necessary simply because the algorithm needs to see the entire α.

We also remark that Lemma 2 implies an algorithm (Algorithm 1) that efficiently computes clock prices whenever the input allocation function can be implemented using clock auctions. In other words, given an input allocation function, we can efficiently compute clock prices that implement it, or declare it is impossible to do so. We restate this fact in the following claim for completeness.

Corollary 1. *There is an efficient algorithm that computes the clock prices that implement an allocation function α on $\mathcal{V}$ whenever such prices exist, and declares it is impossible to do so otherwise.*

Also note that Theorem 2 implies that the decision version of the welfare maximization problem, i.e., checking whether there is a clock auction that guarantees at least a certain welfare, is in NP (see the proof of Theorem 3 in Appendix A for details), which is not clear at all without Theorem 2. Below we show that this problem is in fact NP-complete, which means it is extremely unlikely that there exists an efficient algorithm for computing welfare-maximizing clock auctions in general.

Theorem 3. *The following decision problem is* NP*-complete: Given n, a type space $\mathcal{V}$, a prior distribution $\mathcal{D}$ over $\mathcal{V}$, a family of feasible sets $\mathcal{F}$ in the form of all maximal feasible sets, and a number W, decide whether there exists a clock auction on $\mathcal{V}$ that (1) always allocates to a feasible set of agents, and (2) guarantees expected welfare of at least W over $\mathcal{D}$. This is true even if $\mathcal{D}$ is the uniform distribution over $\mathcal{V}$, and $\mathcal{F}$ has only 2 disjoint maximal sets, one of which is a singleton.*

We deferred the proof of Theorem 3 to Appendix A.

5 Economic Efficiency Independent of Computation

In this section, we investigate the information-theoretical approximation power of clock auctions in terms of welfare, aiming to pin down the gap between the clock-best and the first-best. In other words, our goal is to figure out how much welfare, as a fraction of the first-best welfare, can be achieved using clock auctions, regardless of computational considerations. We focus on problem instances with independent groups, as defined below.

Independent groups. Recall that a problem instance is specified by the number n of agents, the joint type space $\mathcal{V} \subseteq \mathbb{R}_+^n$, the joint prior distribution $\mathcal{D} \in \Delta(\mathcal{V})$, and the family of feasible sets $\mathcal{F} \subseteq 2^{[n]}$. For brevity, we let $\mathcal{M}(\mathcal{F})$ be the family of maximal sets in $\mathcal{F}$, i.e.,

$$\mathcal{M}(\mathcal{F}) = \{S \in \mathcal{F} \mid \nexists T \in \mathcal{F} : S \subsetneq T\}.$$

Definition 3 (independent groups). We say a problem instance $(n, \mathcal{V}, \mathcal{D}, \mathcal{F})$ consists of independent groups, iff the collection of random vectors $\{v_S\}_{S \in \mathcal{M}(\mathcal{F})}$ are independent, where $v \sim \mathcal{D}$, and v_S is the subvector of v restricted to S.

Note that the above definition in particular implies that all sets in $\mathcal{M}(\mathcal{F})$ are disjoint.[8] We note that such instances are considered "hard enough". In particular, the lower bound by Feldman et al. [2022] hold for such instances. We will discuss below the implications of our results for the general problem.

5.1 An Approximation Framework for Independent Groups

We first present a general framework for independent groups, which establishes equivalence between approximating the first-best welfare with independent groups and approximately "extracting" an upper tail of the total value in one group. We first define the latter problem.

[8] In degenerate cases sets in $\mathcal{M}(\mathcal{F})$ might overlap: When the types of some agents are constant, these constant agents can appear in multiple sets in $\mathcal{M}(\mathcal{F})$. Without loss of generality, such constant agents can be removed, and we disregard such degenerate cases in the rest of the section.

Definition 4 (upper tail extraction)**.** Given a set of m agents, a distribution $\mathcal{D}$ over $\mathbb{R}_+^m$ and a target $t \in \mathbb{R}_+$, we say a clock auction protocol (given by the allocation function α) (λ, μ)-extracts the upper tail defined by t iff (1) $\mathbb{E}_{v\sim\mathcal{D}}\left[\|v_{\alpha(v)}\|_1 \mid \alpha(v) \neq \emptyset\right] \geq \lambda \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t]$, and (2) $\min\{\Pr_{v\sim\mathcal{D}}[\alpha(v) \neq \emptyset], \Pr[\|v\|_1 \geq t]\} \geq \mu \cdot \Pr[\|v\|_1 \geq t]$.

In words, the above definition says that the clock auction α should consistently identify cases where the total value is large, and it should be able to do so without forcing too large a fraction of the agents to quit as measured by their total value. λ and μ are generally both no larger than 1, and the closer they are to 1, the better α is in terms of extracting the upper tail. Below we estasblish the equivalence between welfare maximization with independent groups and upper tail extraction.

Lemma 3. *The following two claims are equivalent:*

- *There exists an absolute constant $C_1 \in (0,1)$, such that for any problem instance $(n, \mathcal{V}, \mathcal{D}, \mathcal{F})$ with independent groups, there exists a clock auction protocol α which C_1-approximates the first-best welfare on the problem instance, i.e.,* $\mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha(v)}\|_1] \geq C_1 \cdot \mathbb{E}_{v\sim\mathcal{D}}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v_S\|_1\right]$.
- *There exists an absolute constant $C_2 \in (0,1)$, such that for any problem instance $(m, \mathcal{D}, t)$, there exists a clock auction protocol that (λ, μ)-extracts the upper tail defined by t where $\lambda \cdot \mu \geq C_2$.*

The proof of the lemma is deferred to Appendix A. We make a few remarks regarding the lemma.

- The lemma is agnostic to randomization. In particular, if either claim holds for randomized auction protocols, then the other holds for deterministic auction protocols. This is because auction protocols for both problems can be easily derandomized, as argued in the proof of the lemma.
- The lemma also holds for restricted classes of prior distributions. That is, if the first claim holds when the marginal distribution for each group satisfies a certain condition, then the second claim also holds when the prior distribution satisfies the same condition, and vice versa. Later we will utilize this fact to establish the existence of constant-factor clock auction protocols with independent groups when each group consists of iid agents (i.e., when each group is homogeneous).
- In the upper tail extraction problem, when correlation is allowed, without loss of generality, one may assume the prior distribution is exchangeable, i.e., it is invariant under permutations of the agents. This means we only need to consider highly symmetric (and therefore essentially of much lower dimensionality) prior distributions in the upper tail extraction problem to establish either upper bounds or lower bounds for the welfare maximization problem with independent groups.
- The equivalence also preserves super-constant approximation in certain ways. We choose not to discuss that in detail, since the precise equivalence depends crucially on how the approximation ratio is parametrized. As such, a rigorous and meaningful discussion would be excessively burdensome. However, one could easily instantiate the equivalence for a reasonable parametrization of the approximation ratio, e.g., to establish a super-constant lower bound for the welfare maximization problem.

5.2 Constant-Factor Approximation with Homogeneous Groups

In this subsection, we utilize the approximation framework introduced above to establish the existence of constant-factor clock auction protocols in the special case of independent groups that are each homogeneous. That is, each group consists of iid agents (which may not be identically distributed across different groups). We do so naturally by designing a constant-factor protocol for the upper tail extraction problem with iid agents.

Lemma 4. *There exists an absolute constant $C > 0$, such that for any upper tail extraction instance $(m, \mathcal{D}, t)$ with iid agents, there exists a clock auction protocol that (λ, μ)-extracts the upper tail defined by t where $\lambda \cdot \mu \geq C$.*

Proof. Let $\mathcal{D}_0$ be the marginal distribution of each agent (so $\mathcal{D} = \mathcal{D}_0^m$). Let $P_t = \Pr[\|v\|_1 \geq t]$ and $E_t = \mathbb{E}[\|v\|_1 \mid \|v\|_1 \geq t]$. We first handle the special case where $\mathbb{E}\left[\sum_i v_i \cdot \mathbb{I}[v_i \geq t]\right] \geq \frac{1}{100} P_t \cdot E_t$. When this happens, the following simple protocol (λ, μ)-extracts the upper tail defined by t where $\lambda \cdot \mu \geq 1/100$: raise all agents' prices to t and accept all remaining agents. In particular, the probability that the protocol returns a non-empty set is no larger than P_t, and the (unconditional) expected total value of accepted agents is at least $1/100$ of the benchmark $P_t \cdot E_t$. From now on, we focus on the case where $\mathbb{E}\left[\sum_i v_i \cdot \mathbb{I}[v_i \geq t]\right] < \frac{1}{100} P_t \cdot E_t$. In such cases, without loss of generality we can assume $\Pr_{x \sim \mathcal{D}_0}[x \leq t] = 1$, since the contribution of each v_i on (t, ∞) to the benchmark is relatively small, which means ignoring this part of v_i affects the approximation ratio only by a constant factor.

We then further restrict the problem instance while preserving the approximation ratio within a constant factor. First we assume $\Pr_{x \sim \mathcal{D}_0}[0 < x < E_t/(2m)] = 0$. This is without loss of generality up to a constant factor, because the total contribution of each v_i on $(0, E_t/(2m))$ to the benchmark does not exceed $\frac{1}{2} P_t \cdot E_t$, which means ignoring this part of the distribution affects the approximation ratio only by a constant factor. Then we assume $m \geq 100$, because otherwise one can trivially focus on one agent and get a constant fraction of the benchmark. Moreover, one can assume v_i only takes values in $\{0, E_t/(2m), E_t/m, 2E_t/m, \ldots, t/2\}$ (without loss of generality assuming mt/E_t is a power of 2). This is because one can round v_i down to the closest value in the above set losing only a factor of 2, and as a result, there is some $t' \in [t/2, t]$ such that after rounding each v_i (say, into v_i'), we have

$$\Pr[\|v'\|_1 \geq t'] \leq P_t, \quad \Pr[\|v'\|_1 \geq t'] \cdot \mathbb{E}[\|v'\|_1 \mid \|v'\|_1 \geq t'] \geq \frac{1}{2} P_t \cdot E_t.$$

In other words, we lose another constant factor in the benchmark. In addition, one may assume $100m \cdot \mathbb{E}_{x \sim \mathcal{D}_0}[x] \leq E_t$, because otherwise the trivial protocol that always allocates to all agents $(\lambda, 1)$-extracts the tail defined by t for some absolute constant λ. These restrictions (which are without loss of generality in any case) are non-essential, and only serve the purpose of simplifying the presentation.

Let $s = \log_2(2mt/E_t)$. For each $j \in [s]$, let $u_j = t/2^j$, and $q_j = \Pr_{x \sim \mathcal{D}_0}[x = u_j]$. Moreover, let $q_0 = \Pr_{x \sim \mathcal{D}_0}[x = 0]$. Consider s candidate clock auction policies, where candidate policy $j \in [s]$ raises every agent's price to u_j and accepts all active agents if the conditional expected total value of them is at least $t/100$. We will argue below that at least one of these s candidate policies is good enough. In particular, we will argue that E_t is not too much larger than t, and the probability that one of these policies accepts a non-empty set is at least P_t. For these purposes, we need to upper bound the tail of $\|v\|_1$ when $v \sim \mathcal{D}$, and lower bound the tail of $v^{(j)}$ for each $j \in [s]$, where $v^{(j)}$ is the sum of m iid variables, each taking value $\mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j]$ with probability $\Pr_{x \sim \mathcal{D}_0}[x \geq u_j]$, and 0 otherwise. For each $j \in [s]$, let

$$k_j = \lceil t/(100\mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j]) \rceil.$$

This is the number of positive summands we need so that $v^{(j)} \geq t/100$. Note that

$$k_j \leq \frac{t}{100\mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j]} + 1 \leq \frac{t}{100u_j} + 1 = \frac{t}{100t/2^j} + 1 \leq \frac{m}{50} + 1 \leq m.$$

In other words, we never need more than m positive summands (which would be impossible and make the reasoning below ill-formed). It is known that the tail of a binomial variable X with parameters n and p satisfies the following claim (see, e.g., Lemma 4.7.2 in [Ash, 2012]): For any $k \in \{0, \ldots, n\}$,

$$\Pr[X \geq k] \geq \frac{1}{\sqrt{8k(1-k/n)}} \cdot \exp(-nD(k/n \,\|\, p)),$$

where

$$D(a \,\|\, p) = a \ln \frac{a}{p} + (1-a) \ln \frac{1-a}{1-p} \leq a \ln \frac{a}{p},$$

whenever $0 \leq p \leq a \leq 1$. Utilizing this fact, we have

$$\begin{aligned}
\Pr_{v \sim \mathcal{D}}[v^{(j)} \geq t/100] &\geq \frac{1}{\sqrt{8k_j(1-k_j/m)}} \cdot \exp\left(-mD\left(k_j/m \,\middle\|\, \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right) \\
&\geq \frac{1}{\sqrt{8k_j}} \cdot \exp\left(-k_j \ln\left(k_j/\left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right)\right) \\
&\geq \exp\left(-k_j \ln\left(k_j/\left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right) - \ln k_j - 2\right) \\
&\geq \exp\left(-4k_j \ln\left(4k_j/\left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right)\right).
\end{aligned}$$

Let

$$c_j = \frac{4k_j}{m} \ln\left(4k_j/\left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right),$$

and $c^* = \min_j c_j$. Note that this means that out of the s candidate policies, there is one that produces total (conditional expected) value at least t with probability at least $\exp(-mc^*)$. This is our desired good policy.

Now we turn to the tail of $\|v\|_1$. We break $\|v\|_1$ into two parts: the contribution of small summands and that of large ones. To be specific, let j^* be the largest integer in $[s]$ such that $2^j/m \leq c^*$. Let

$$v^s = \sum_{i \in [m]} v_i \cdot \mathbb{I}[v_i \leq u_{j^*}], \quad v^\ell = \sum_{i \in [m]} v_i \cdot \mathbb{I}[v_i > u_{j^*}].$$

Clearly $\|v\|_1 = v^s + v^\ell$, and as a result, for any $\alpha \geq 0$ (we will later focus on cases where $\alpha \geq t/2$), we have

$$\Pr[\|v\|_1 \geq 2\alpha] \leq \Pr[v^s \geq \alpha] + \Pr[v^\ell \geq \alpha].$$

We will bound the two quantities on the right hand side separately.

For v^ℓ, we take the standard approach through the moment generating function. For any $\alpha \geq t/2$ and $\theta > 0$, we know that

$$\Pr[v^\ell \geq \alpha] = \Pr[\exp(\theta \cdot v^\ell) \geq \exp(\theta \cdot \alpha)] \leq \frac{\mathbb{E}[\exp(\theta \cdot v^\ell)]}{\exp(\theta \cdot \alpha)} = \left(\frac{\mathbb{E}_{x \sim \mathcal{D}_0}[\exp(\theta \cdot x \cdot \mathbb{I}[x \geq u_{j^*}])]}{\exp(\theta \cdot \alpha/m)}\right)^m.$$

In other words,

$$\Pr[v^\ell \geq \alpha] \leq \exp(-m \cdot I(\alpha/m)),$$

where

$$I(\alpha/m) = \sup_\theta \left[\theta \cdot \alpha/m - \ln\left(\mathbb{E}_{x \sim \mathcal{D}_0} \exp(\theta \cdot x \cdot \mathbb{I}[x \geq u_{j^*}])\right)\right].$$

By choosing $\theta = 5m \cdot c^*/t$, we have

$$I(\alpha/m) \geq 5(m \cdot c^*) \cdot \alpha/(mt) - \ln\left(q_0 + \sum_{j \in [s] \setminus [j^*]} q_j + \sum_{j \in [j^*]} q_j \cdot \exp(5m \cdot c^* \cdot u_j/t)\right)$$

$$\geq 5c^* \cdot \frac{\alpha}{t} - \ln\left(1 + \sum_{j \in [j^*]} q_j \cdot \exp(5m \cdot c_j \cdot u_j/t)\right) \qquad (c_j \geq c^*, q_0 + \textstyle\sum_{j \in [s] \setminus [j^*]} q_j \leq 1)$$

$$= 5c^* \cdot \frac{\alpha}{t} - \ln\left(1 + \sum_{j \in [j^*]} q_j \exp\left(20k_j u_j t^{-1} \ln\left(4k_j \Big/ \left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right)\right)\right).$$

(plugging in the definition of c_j)

Now we can (partially) plug in the choice of k_j and get:

$$I(\alpha/m) \geq 5c^* \cdot \frac{\alpha}{t} - \ln\left(1 + \sum_{j \in [j^*]} q_j \exp\left(\frac{u_j \ln\left(4k_j/(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j])\right)}{\mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j]}\right)\right)$$

$$\geq 5c^* \cdot \frac{\alpha}{t} - \ln\left(1 + \sum_{j \in [j^*]} q_j \left(4k_j \Big/ \left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right)\right) \qquad (u_j \leq \mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j])$$

$$\geq 5c^* \cdot \frac{\alpha}{t} - \ln\left(1 + \sum_{j \in [j^*]} 4k_j/m\right) \qquad (q_j \leq \Pr_{x \sim \mathcal{D}_0}[x \geq u_j])$$

$$\geq 5c^* \cdot \frac{\alpha}{t} - \sum_{j \in [j^*]} 4k_j/m. \qquad (\ln(1+z) \leq z \text{ for } z \geq 0)$$

Now recall that for each $j \in [j^*]$,

$$k_j = \lceil t/(100\mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j]) \rceil \leq t/(50\mathbb{E}_{x \sim \mathcal{D}_0}[x \mid x \geq u_j]) \leq t/(50u_j) = t/(50t/2^j) = 2^j/50.$$

So, by the choice of j^*,

$$\sum_{j \in [j^*]} 4k_j/m \leq \sum_{j \in [j^*]} 2^j/(10m) \leq 2^{j^*+1}/(10m) \leq \frac{1}{5}c^*.$$

Plugging this back, we get

$$I(\alpha/m) \geq 5c^* \cdot \frac{\alpha}{t} - \frac{1}{5}c^* \geq 2.3c^* \cdot \frac{\alpha}{t},$$

where the latter inequality is because $\alpha \geq t/2$. In other words, we have shown that for any $\alpha \geq t/2$,

$$\Pr[v^\ell \geq \alpha] \leq \exp\left(-2.3mc^* \cdot \frac{\alpha}{t}\right).$$

Then we consider v^s. If $c^* > 2$ then $v^s = 0$, so in the following we always assume $c^* \leq 2$. We take a much coarser approach: We view each summand $x \cdot \mathbb{I}[x < u_{j^*}]$ where $x \sim \mathcal{D}_0$ as an arbitrary random variable supported on $[0, u_{j^*+1}] \subseteq [0, t/(2mc^*)]$ whose mean is at most $\mathbb{E}_{x \sim \mathcal{D}_0}[x] \leq t/(100m)$, and apply the Chernoff bound. In particular, the worst case is when $x \cdot \mathbb{I}[x < u_{j^*}]$ only takes values in $\{0, t/(2mc^*)\}$, and is equal to $t/(2mc^*)$ with probability precisely $\mathbb{E}[x]/(t/(2mc^*)) \leq$

$t/(100m)/(t/(2mc^*)) = c^*/50$. Concretely, for any $\alpha \geq t/2$, we have

$$\begin{aligned}\Pr[v^s \geq \alpha] &\leq \exp(-m \cdot D(\alpha/(m \cdot (t/2mc^*)) \,\|\, c^*/50)) \\ &= \exp(-m \cdot D(2c^*\alpha/t \,\|\, c^*/50)) \\ &\leq \exp\left(-m \cdot \left(2c^* \cdot \frac{\alpha}{t} + \ln(100\alpha/t)\right)\right) \\ &\leq \exp\left(-2mc^* \cdot \frac{\alpha}{t} - m\right).\end{aligned}$$

Note that the above derivation implicitly assumes α is not too large, so it is possible that $v^s \geq \alpha$ (i.e., we need no more than m positive summands). For α large enough, $\Pr[v^s \geq \alpha] = 0$, and the bound above holds trivially.

Now we can put the bounds for v^ℓ and v^s together and conclude that for each $\alpha \geq t$,

$$\Pr[\|v\|_1 \geq \alpha] \leq \exp\left(-2.3mc^* \cdot \frac{\alpha}{2t}\right) + \exp\left(-2mc^* \cdot \frac{\alpha}{2t} - m\right) \leq \exp\left(-mc^* \cdot \frac{\alpha}{t}\right).$$

Here we use the fact that $mc^* \geq 1$, which follows from the choice of c_j for each $j \in [s]$. In particular, choosing $\alpha = t$, we get

$$P_t \leq \exp(-mc^*),$$

which is no larger than the probability that our good policy extracts total value t.

The one thing left to be shown is that E_t is not too much larger than t. If $E_t \leq 2t$ then we are done, so from now on we assume $E_t > 2t$. Observe that each $v^{(j)}$ lower bounds $\|v\|_1$. For each $j \in [s]$, by essentially repeating the argument used to lower bound $\Pr[v^{(j)} \geq t/100]$, one can show that

$$\Pr[v^{(j)} \geq t] \geq \exp\left(-400k_j \ln\left(400k_j \Big/ \left(m \cdot \Pr_{x \sim \mathcal{D}_0}[x \geq u_j]\right)\right)\right) \geq \exp(-1000mc_j).$$

Note that we need the fact that $E_t > 2t$ so that it is always possible that $v^{(j)} \geq t$ even when $j = s$ (i.e., we need no more than m summands to be positive), which ensures that the derivation of our bound is sensible. So we have

$$P_t \geq \max_j \Pr[v^{(j)} \geq t] \geq \max_j \exp(-1000mc_j) = \exp(-1000mc^*).$$

On the other hand, we also have: For any $\alpha \geq t$,

$$\Pr[\|v\|_1 \geq \alpha] \leq \exp\left(-mc^* \cdot \frac{\alpha}{t}\right).$$

These two facts together imply

$$E_t \leq t \cdot \mathbb{E}_{y \sim \mathrm{Exp}(mc^*)}[y \mid y \geq 1000] = (1000 + 1/(mc^*))t \leq 1001t,$$

where $\mathrm{Exp}(mc^*)$ denotes the exponential distribution with rate mc^*, and the last inequality is because $mc^* \geq 1$ due to the choice of c_j for each $j \in [s]$.

Now we can put everything together and finish the proof. We have constructed a clock auction protocol α satisfying:

$$\Pr[\alpha(v) \neq \emptyset] \geq \exp(-mc^*), \quad \mathbb{E}[\|v_{\alpha(v)}\|_1 \mid \alpha(v) \neq \emptyset] \geq t.$$

On the other hand, we have

$$P_t \leq \exp(-mc^*), \quad E_t \leq 1001t.$$

So our protocol α $(1/1001, 1)$-extracts the upper tail defined by t. Combined with the simplifying restrictions which impose another constant-factor gap, this implies the existence of an absolute constant C as desired. □

Lemma 4 in fact proves a technical conjecture by Feldman et al. [2022], which connects clock auctions to the theory of large deviations. As a direct corollary of Lemma 3 and Lemma 4, we have the following result.

Corollary 2. *There is an absolute constant $C > 0$, such that for any welfare maximization instance $(n, \mathcal{V}, \mathcal{D}, \mathcal{F})$ with independent groups where each group consists of iid agents, there exists a clock auction protocol α which C-approximates the first-best welfare on the instance.*

We view Corollary 2 as strong evidence that constant-factor clock auctions exist for the general problem. In particular, Corollary 2 removes the immediate technical obstacle to constant-factor clock auctions identified by Feldman et al. [2022]. We also remark that although not emphasized, our construction for Corollary 2 through Lemma 3 and Lemma 4 can be computed efficiently.

6 Discussion and Future Research

In this paper, we make progress on several fronts towards a more complete understanding of the power of clock auctions. Computationally, we investigate implementation and optimization, arguably the most natural and important problems in this contex. Our results show that the former is "easy", while the latter is "hard but not too hard". These are the first results regarding the computational complexity of clock auctions. En route, we develop a complete characterization of allocation functions implementable using clock auctions, which may be of independent interest.

We then turn to the economic efficiency of clock auctions independent of computational issues. We present a framework connecting approximate welfare maximization to the much cleaner problem of upper tail extraction, which can be viewed as a handy tool for proving both positive and negative results. We further present a constant-factor construction for upper tail extraction in the special case of iid agents, which, through our framework, immediately implies a constant-factor construction for welfare maximization in the special case of independent groups that are each homogeneous. These results pave the way for tight bounds for the general problem.

Moving forward, the most important open question is to figure out the right gap between the first-best and the clock-best. Concretely, we believe the immediate next step is to fully understand the upper tail extraction problem, which would either establish a super-constant lower bound on the gap of interest, or provide strong evidence that the gap is in fact constant. To this end, we believe our construction for the iid case has the potential to generalize to the case with independent but non-identical agents. The case with correlation may require a fundamentally different (and likely "softer") approach.

In terms of computation, one natural question is whether there are efficient approximation algorithms for welfare maximization (with the clock-best being the benchmark). Our hardness reduction rules out the possibility of a fully polynomial-time approximation scheme (FPTAS), but we believe even a constant-factor algorithm would be practically meaningful. Designing such a constant-factor algorithm (without closing the gap between the clock-best and the first-best) would likely involve heavy use of our allocation-based characterization.

References

Mohammad Akbarpour and Shengwu Li. Credible auctions: A trilemma. *Econometrica*, 88(2):425–467, 2020.

Saeed Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *SIAM Journal on Computing*, 43(2):930–972, 2014.

Robert B Ash. *Information theory.* Courier Corporation, 2012.

Lawrence M Ausubel and Oleg Baranov. A practical guide to the combinatorial clock auction. *The Economic Journal*, 127 (605):F334–F350, 2017.

Lawrence M Ausubel, Peter Cramton, and Paul Milgrom. The clock-proxy auction: A practical combinatorial auction design. *Handbook of spectrum auction design*, pages 120–140, 2006.

Eric Balkanski, Pranav Garimidi, Vasilis Gkatzelis, Daniel Schoepflin, and Xizhi Tan. Deterministic budget-feasible clock auctions. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2940–2963. SIAM, 2022.

Kiarash Banihashem, MohammadTaghi Hajiaghayi, Dariusz R Kowalski, Piotr Krysta, and Jan Olkowski. Power of posted-price mechanisms for prophet inequalities. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4580–4604. SIAM, 2024.

Nicolas Bousquet, Yang Cai, Christoph Hunkenschröder, and Adrian Vetta. On the economic efficiency of the combinatorial clock auction. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1407–1423. SIAM, 2016.

Felix Brandt and Tuomas Sandholm. Unconditional privacy in social choice. In *Proceedings of the 10th conference on Theoretical aspects of rationality and knowledge*, pages 207–218, 2005.

Shuchi Chawla, Jason D Hartline, David L Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 311–320, 2010.

Giorgos Christodoulou, Vasilis Gkatzelis, and Daniel Schoepflin. Optimal deterministic clock auctions and beyond. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2022.

Vincent Conitzer, Andrew Davenport, and Jayant Kalagnanam. Improved bounds for computing kemeny rankings. In *AAAI*, volume 6, pages 620–626, 2006.

Jose Correa, Patricio Foncea, Ruben Hoeksma, Tim Oosterwijk, and Tjark Vredeveld. Recent developments in prophet inequalities. *ACM SIGecom Exchanges*, 17(1):61–70, 2019.

José Correa, Andrés Cristi, Andrés Fielbaum, Tristan Pollner, and S Matthew Weinberg. Optimal item pricing in online combinatorial auctions. *Mathematical Programming*, 206(1):429–460, 2024.

Andrés Cristi and Bruno Ziliotto. Prophet inequalities require only a constant number of samples. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 491–502, 2024.

Paul Dütting, Vasilis Gkatzelis, and Tim Roughgarden. The performance of deferred-acceptance auctions. *Mathematics of Operations Research*, 42(4):897–914, 2017.

Michal Feldman, Nick Gravin, and Brendan Lucier. Combinatorial auctions via posted prices. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 123–135. SIAM, 2014.

Michal Feldman, Vasilis Gkatzelis, Nick Gravin, and Daniel Schoepflin. Bayesian and randomized clock auctions. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 820–845, 2022.

Diodato Ferraioli and Carmine Ventre. On the connection between greedy algorithms and imperfect rationality. In *Proceedings of the 24th ACM Conference on Economics and Computation*, pages 657–677, 2023.

Diodato Ferraioli, Paolo Penna, and Carmine Ventre. Two-way greedy: Algorithms for imperfect rationality. In *International Conference on Web and Internet Economics*, pages 3–21. Springer, 2021.

Vasilis Gkatzelis, Rishi Patel, Emmanouil Pountourakis, and Daniel Schoepflin. Prior-free clock auctions for bidders with interdependent values. In *International Symposium on Algorithmic Game Theory*, pages 64–78. Springer, 2021.

Vasilis Gkatzelis, Daniel Schoepflin, and Xizhi Tan. Clock auctions augmented with unreliable advice. *arXiv preprint arXiv:2408.06483*, 2024.

Gagan Goel, Renato Paes Leme, Jon Schneider, David Thompson, and Hanrui Zhang. Eligibility mechanisms: Auctions meet information retrieval. In *Proceedings of the ACM Web Conference 2023*, pages 3541–3549, 2023.

Mohammad Taghi Hajiaghayi, Robert Kleinberg, and Tuomas Sandholm. Automated online mechanism design and prophet inequalities. In *AAAI*, volume 7, pages 58–65, 2007.

Maarten Janssen and Bernhard Kasberger. On the clock of the combinatorial clock auction. *Theoretical Economics*, 14(4): 1271–1307, 2019.

Robert Kleinberg and Seth Matthew Weinberg. Matroid prophet inequalities. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 123–136, 2012.

Jonathan Levin and Andrzej Skrzypacz. Properties of the combinatorial clock auction. *American Economic Review*, 106(9): 2528–2551, 2016.

Shengwu Li. Obviously strategy-proof mechanisms. *American Economic Review*, 107(11):3257–3287, 2017.

Simon Loertscher and Leslie M Marx. Asymptotically optimal prior-free clock auctions. *Journal of Economic Theory*, 187: 105030, 2020.

Brendan Lucier. An economic view of prophet inequalities. *ACM SIGecom Exchanges*, 16(1):24–47, 2017.

Paul Milgrom and Ilya Segal. Clock auctions and radio spectrum reallocation. *Journal of Political Economy*, 128(1):1–31, 2020.

Neil Newman, Kevin Leyton-Brown, Paul Milgrom, and Ilya Segal. Incentive auction design alternatives: A simulation study. *Management Science*, 2024.

A Omitted Proofs

Proof of Lemma 1. Fix some α and suppose it is implemented by clock prices p. Consider any $u \in \bar{\mathcal{V}}$ and $S \subseteq [n]$, such that S is a free-riding group at u. We will show that for any $v \in \mathcal{V}$ where $v \geq_S u$, $S \subseteq \alpha(v)$.

Let $(A_1, \ldots, A_t)$ be the sequence of active sets produced by p when the types are v. We know $p(A_1, \ldots, A_t) = \perp$ and $A_t = \alpha(v)$, since p implements α on $\mathcal{V}$. For each $i \in [n]$, let $t_i = \min\{\tau \mid p(A_1, \ldots, A_\tau) > u_i\}$. In particular, if $p_i(A_1, \ldots, A_{t-1}) \leq u_i$, then we let $t_i = t$. Observe that if $t_i = t$ for all $i \in S$, then for each $i \in S$, we must have $v_i \geq u_i \geq p_i(A_1, \ldots, A_{t-1})$, which means $i \in A_t$ and $i \in \alpha(v)$. So we only need to show that $t_i = t$ for all $i \in S$.

To this end, let $i^* = \operatorname{argmin}_{i \in S} t_i$. Suppose towards a contradiction that $t_{i^*} < t$. Since S is a free-riding group at u and $i^* \in S$, there must exists some $v' \in \mathcal{V}$ such that $v' \geq_{S \setminus \{i^*\}} u$ and $i^* \in \alpha(v')$. Let $(A'_1, \ldots, A'_{t'})$ be the sequence of active sets produced by p when the types are v'. Again, we must have $p(A'_1, \ldots, A'_{t'}) = \perp$ and $A'_{t'} = \alpha(v')$, since p implements α on $\mathcal{V}$. Now observe that p cannot distinguish between v and v' before time t_{i^*}, i.e., for all $\tau \leq t_{i^*}$, $A_\tau = A'_\tau$. Intuitively, this is because for any $i \in S$, the clock price never rises above $u_i \leq \min\{v_i, v'_i\}$ before time t_{i^*}, and for any $i \in [n] \setminus S$, $v_i = v'_i = u_i$. We will formally prove this fact momentarily, but for now, let us show how it leads to a contradiction, which establishes the lemma to be proved.

Given that $A_\tau = A'_\tau$ for all $\tau \leq t_{i^*}$, we have $p_{i^*}(A_1, \ldots, A_{t_{i^*}}) = p_{i^*}(A'_1, \ldots, A'_{t_{i^*}})$. Since $t_{i^*} < t$ and $v' \geq_{S \setminus \{i\}} u$, we have $v'_{i^*} = u_{i^*} \leq v_{i^*} < p_{i^*}(A_1, \ldots, A_{t_{i^*}}) = p_{i^*}(A'_1, \ldots, A'_{t_{i^*}})$, which means $i^* \notin A'_{t^*_i+1}$. This implies $i^* \notin A'_{t'} = \alpha(v')$ since $A'_{t'} \subseteq A'_{t_{i^*}+1}$, contradicting the choice of v'.

Now we only need to show that for all $\tau \leq t_{i^*}$, $A_\tau = A'_\tau$. We proceed by induction. Observe that $A_1 = A'_1 = [n]$. Suppose for some $k \leq t_{i^*}$, the following is true: for any $\tau < k$, $A_\tau = A'_\tau$. We need to show $A_k = A'_k$. Let $q = p(A_1, \ldots, A_{k-1}) = p(A'_1, \ldots, A'_{k-1})$. By definition, $A_k = \{i \in [n] \mid v_i \geq q_i\}$, $A'_k = \{i \in [n] \mid v'_i \geq q_i\}$. We argue that for any $i \in [n]$, $i \in A_k \iff i \in A'_k$. There are two cases:

- When $i \in S$, by the choice of i^* we know that $t_i \geq t_{i^*} > k - 1$, which means $u_i \geq q_i$. Since $v \geq_S u$, we have $v_i \geq u_i \geq q_i$, and $i \in A_k$. Since $v' \geq_{S \setminus \{i\}} u$, we have $v'_i = u_i \geq q_i$, and $i \in A'_k$.
- When $i \in [n] \setminus S$, for similar reasons we know that $v_i = u_i = v'_i$. So $v_i \geq q_i \iff v'_i \geq q_i$, which means $i \in A_i \iff i \in A'_i$.

This finishes the proof. □

Proof of Lemma 2. We first show that Algorithm 1 is efficient, i.e., it runs in time polynomial in n and $m = |\mathcal{V}|$. The bottleneck of the algorithm is Line 11, where we need to check for every $i \in A_t$ whether i free-rides $A_t \setminus \{i\}$ at u. This can be done by enumerating $v \in \mathcal{V}$ and checking whether (1) $v \geq_{A_t \setminus \{i\}}$ and (2) $i \in \alpha(v)$, which can be done in time $O(|\mathcal{V}|) = O(m)$. Line 11 therefore takes time $O(nm)$. Then in Line 14, we can reuse the result computed in Line 11 and retrieve i^* directly in time $O(n)$. Another tricky step is Line 5, where we either retrieve u from the previous time or re-compute it through a recursive call. Even if we always re-compute u, the overhead incurred is still polynomial, since we make precisely one recursive call to the previous time, and the recursion tree is of size $O(t)$. Moreover, we will see below that the entire auction must terminate in nm steps, which means we always have $t \leq mn$, and the total time it takes for the algorithm to compute all clock prices throughout an auction is polynomial.

Also, observe that Algorithm 1 outputs feasible (i.e., weakly increasing) clock prices. This can be easily verified by examining the algorithm. Now we prove that Algorithm 1 in fact outputs clock prices that implement α on $\mathcal{V}$. First observe that the clock auction induced by the algorithm terminates for every $v \in \mathcal{V}$. This is simply because at each time t, if the algorithm does not output $\perp$, then the clock price strictly increases for some $i \in [n]$, to either the next feasible value in $\mathcal{V}_i$ or to ∞. Such an increase can happen at most $\sum_i |\mathcal{V}_i|$ times (by which time all clock prices are ∞ and no agent remains active), which means the auction must terminate in $n|\mathcal{V}| = nm$ steps. So we only need to argue that the auction correctly identifies the set of winners. Fix any $v \in \mathcal{V}$ and consider the behavior of the algorithm. Suppose the auction terminates at time t and the sequence of active sets is $(A_1, \dots, A_t)$, with $p(A_1, \dots, A_t) = \perp$. We need to show that $A_t = \alpha(v)$.

We first show that $\alpha(v) \subseteq A_t$, i.e., for each $i \in \alpha(v)$, $i \in A_t$. In fact, we argue inductively that for any $\tau \le t$:

- For each $i \in \alpha(v)$, $i \in A_\tau$.
- For each $i \in [n] \setminus \alpha(v)$, $p_i(A_1, \dots, A_{\tau-1}) \le \min\{v \in \mathcal{V}_i \mid v > v_i\}$.

Suppose this is true for $\tau = k - 1$, and below we establish it for $\tau = k$.

Since $i \in A_{k-1}$, we have $p_i(A_1, \dots, A_{k-2}) \le v_i$. For the first bullet point, we have two possible cases:

- When $p_i(A_1, \dots, A_{k-2}) < v_i$, since p_i can only increase to the next feasible value in $\mathcal{V}_i$, we know that $p_i(A_1, \dots, A_{k-1}) \le v_i$, and $i \in A_k$.
- When $p_i(A_1, \dots, A_{k-2}) = v_i$, we argue that $p_i(A_1, \dots, A_{k-1}) = v_i$. Consider the behavior of Algorithm 1 at time $k-1$. When i^* is computed, we know that for any $i' \in [n] \setminus A_{k-1}$, $u_{i'} = v_{i'}$. This is because by the induction hypothesis, $p_{i'}(A_1, \dots, A_{k-2}) = \min\{v \in \mathcal{V}_{i'} \mid v > v_{i'}\}$, and $u_{i'}$ is obtained by decreasing $p_{i'}(A_1, \dots, A_{k-2})$ to the next feasible value. Moreover, for any $i' \in A_{k-1}$, clearly $u_{i'} \le v_{i'}$. As a result, $v \ge_{A_{k-1} \setminus \{i\}} u$ at the time when i^* is computed, and i^* cannot be i. This means $p_i(A_1, \dots, A_{k-1}) = v_i$, and $i \in A_k$.

For the second bullet point, again we have two possible cases:

- When $p_i(A_1, \dots, A_{k-2}) \le v_i$, since the clock price for each agent can only increase to the next feasible value, we always have $p_i(A_1, \dots, A_{k-1}) \le \min\{v \in \mathcal{V}_i \mid v > v_i\}$.
- When $p_i(A_1, \dots, A_{k-2}) = \min\{v \in \mathcal{V}_i \mid v > v_i\}$, we know $i \notin A_{k-1}$, and u_i changes only in Line 7 and Line 20. In Line 7, u_i is decreased to the next feasible value, and in Line 20, it is increased to the next feasible value. As a result, $p_i(A_1, \dots, A_{k-1})$ returned by the algorithm (value of u_i at Line 21) is the same as $p_i(A_1, \dots, A_{k-2})$ (value of u_i at Line 5).

Now we show that $A_t \subseteq \alpha(v)$, i.e., for each $i \in [n] \setminus \alpha(v)$, $i \notin A_t$. Suppose towards a contradiction that there exists $i \in [n] \setminus \alpha(v)$ such that $i \in A_t$. Then, since $\alpha(v) \subseteq A_t$ and $p_{i'}(A_1, \dots, A_{t-1}) = \min\{v \in \mathcal{V}_{i'} \mid v > v_{i'}\}$ for all $i' \in [n] \setminus A_t$, we know u at Line 8 satisfies $v \ge_{A_t} u$. Now clearly, $i \in A_t \not\subseteq \alpha(v)$, which means the if-condition in Line 8 is not satisfied and the algorithm cannot output $\perp$, a contradiction. This finishes the proof. ◻

Proof of Theorem 3. We first show the problem is in NP. That is, we show there is a certificate that can be checked in polynomial time iff the target welfare W is achievable. The certificate we use is an allocation function α on $\mathcal{V}$ that (1) is implementable by clock auctions on $\mathcal{V}$, (2) is feasible, and (3) results in expected welfare at least W. Such a certificate can be efficiently checked because (1) by Theorem 2, there is an algorithm that checks the implementability of α in polynomial time, and (2) given α and $\mathcal{F}$ in the maximal sets representation, it is easy to check whether α is feasible, and (3) given α and $\mathcal{D}$, it is easy to compute the expected welfare guaranteed by α and compare it against W. There exists such an allocation function α iff W can be achieved.

Now we show the problem is NP-hard. We present a reduction from 3-SAT. Given a 3-SAT instance with a variables and b clauses, we create an instance of the above decision problem with $n = 2a + 1$ agents, $m = 2a + b$ points in the type space, and $k = 2$ disjoint maximal sets describing feasibility, one of which is a singleton. We let $\mathcal{D}$ be the uniform distribution over the type space. Without loss of generality, we assume the 3-SAT instance consists of at least 10 (or any number that is large enough) copies of the clause $x_i^+ \vee x_i^-$ for each variable x_i. Intuitively, this ensures that x_i must "have a value". We describe the construction below.

- The two maximal feasible sets are $\{1\}$ and $\{2, \ldots, 2a+1\} = \{2, \ldots, n\}$.
- Variables: Intuitively, each variable x_i where $i \in [a]$ "involves" agents $2i$ (corresponding to x_i^+) and $(2i+1)$ (corresponding to x_i^-). Agent 1 is reserved for special use. For each variable x_i where $i \in [a]$, create two type vectors $v^{x_i^+}$ and $v^{x_i^-}$ in $\mathcal{V}$. $v^{x_i^+}$ is constructed as follows:
  - $v_1^{x_i^+} = 3a - 7/2$.
  - $v_{2i}^{x_i^+} = 1$.
  - $v_j^{x_i^+} = 3$ for all $j \in [2a+1] \setminus \{1, 2i\}$.

  $v^{x_i^-}$ is constructed similarly, except that $v_{2i+1}^{x_i^-} = 1$ and $v_{2i}^{x_i^-} = 3$:
  - $v_1^{x_i^-} = 3a - 7/2$.
  - $v_{2i+1}^{x_i^-} = 1$.
  - $v_j^{x_i^-} = 3$ for all $j \in [2a+1] \setminus \{1, 2i+1\}$.
- Clauses: Intuitively, each clause c_i where $i \in [b]$ "involves" agents corresponding to the literals that appear in c_i. That is, we create a type vector v^{c_i} for each clause c_i where $i \in [b]$ in the following way:
  - $v_1^{c_i} = 3a - 7/2$.
  - For each x_j^s appearing in c_i, $v_{2j+\mathbb{I}[s=-]}^{c_i} = 1$, where s is either $+$ or $-$, and $\mathbb{I}[\cdot]$ is the indicator function.
  - For all other $j \in [2a+1]$, $v_j^{c_i} = 3$.

We pick

$$W = \frac{a \cdot (3a-2) + a \cdot (3a-3) + b \cdot (3a - 7/2)}{2a+b},$$

and claim that the target welfare of W is achievable iff the 3-SAT instance can be satisfied.

Below we prove that W is achievable iff the 3-SAT instance can be satisfied. We first show the "if" direction, i.e., given an assignment that satisfies the clauses, W is achievable. We construct an allocation function α in the following way:

- For each variable x_i:
  - If $x_i = +$ in the assignment, then

$$\alpha(v^{x_i^+}) = \{2, \ldots, 2a+1\} \setminus \{2i\} \quad \text{and} \quad \alpha(v^{x_i^-}) = \{2, \ldots, 2a+1\}.$$

  - If $x_i = -$ in the assignment, then

$$\alpha(v^{x_i^+}) = \{2, \ldots, 2a+1\} \quad \text{and} \quad \alpha(v^{x_i^-}) = \{2, \ldots, 2a+1\} \setminus \{2i+1\}.$$

- For each clause c_i, $\alpha(v^{c_i}) = \{1\}$.

One can check that α is feasible, and the welfare guaranteed by α is precisely W. Below we argue that α is implementable by clock auctions. We only need to verify the condition in Theorem 1.

Suppose there is a free-riding group S at $u \in \bar{\mathcal{V}}$, and moreover, there exists $v \in \mathcal{V}$ where $v \geq_S u$. We show that $S \subseteq \alpha(V)$. In doing so, without loss of generality we assume $|\{i \in \{2, \ldots, 2a+1\} \mid u_i = 1\}| > 1$, because otherwise u is a maximal type vector and the claim holds trivially. Let $T = \{i \in S \mid u_i = 1\}$. We claim that for all $i \in \{2, \ldots, 2a+1\} \setminus T$, $u_i \neq 1$ (which means $u_i = 3$). To see why this must be true, suppose otherwise. There are two cases that both lead to contradictions:

- If $T = \emptyset$, then for any $v' \in \mathcal{V}$ where $v' \geq_{S \setminus \{i\}}$ for some $i \in S$, $|\{j \in \{2, \ldots, 2a+1\} \mid v'_j = 1\}| > 1$ (recall the assumption we made above without loss of generality), and therefore $i \notin \alpha(v')$ by the construction of α because v' must be a type vector corresponding to a clause.

- If $T \neq \emptyset$, then for any $v' \in \mathcal{V}$ where $v' \geq_{S \setminus \{i\}}$ for some $i \in T$, $|\{j \in \{2, \dots, 2a+1\} \mid v'_j = 1\}| > 1$, and therefore $i \notin \alpha(v')$ by the construction of α because v' must be a type vector corresponding to a clause.

Now for each $i \in \{2, \dots, 2a+1\}$, define $\mathrm{ind}(i) \in [a]$ and $\mathrm{sgn}(i) \in \{+, -\}$ such that

$$i = 2 \cdot \mathrm{ind}(i) + \mathbb{I}[\mathrm{sgn}(i) = -].$$

That is, i is the agent corresponding to $x_{\mathrm{ind}(i)}^{\mathrm{sgn}(i)}$ in our reduction. One can check that for each $i \in T$, in order for i to free-ride $S \setminus \{i\}$ at u, we must have $x_{\mathrm{ind}(i)} \neq \mathrm{sgn}(i)$ in the assignment that satisfies the 3-SAT instance, because only in this case there is some $v' \in \mathcal{V}$ where $i \in \alpha(v')$ and $v'_i = 1$, which is a necessary condition for i to free-ride $S \setminus \{i\}$ at u.

Given the above, we claim that v cannot be a type vector corresponding to some clause. Suppose towards a contradiction that there is a clause c_i where $i \in [b]$ such that $v = v^{c_i}$. Then for any $j \in \{2, \dots, 2a+1\}$, $v_j = 1 \implies u_j = 1 \implies j \in T$ because $v \geq_S u$. Recall that $v_j = 1$ iff $x_{\mathrm{ind}(j)}^{\mathrm{sgn}(j)}$ appears in c_i. The above implies that for any literal x_j^s appearing in c_i, $x_j \neq s$ in the assignment that satisfies the 3-SAT instance, which is impossible because the assignment by definition must satisfy c_i. Moreover, the above reasoning also implies $1 \notin S$, because for 1 to free-ride $S \setminus \{1\}$, there must be some c_i such that $v^{c_i} \geq_{S \setminus \{i\}} u \implies v^{c_i} \geq_S u$.

Now the only possibility left for v is $v = v^{x_i^s}$ for some $i \in [a]$ and $s \in \{+, -\}$. Then we must have $2i + \mathbb{I}[s = -] \in T$, and $2i + \mathbb{I}[s = -] \in \alpha(v)$ because this is the only way for $2i + \mathbb{I}[s = -]$ to free-ride $S \setminus \{2i + \mathbb{I}[s = -]\}$ at u. As for each $i' \in S \setminus \{2i + \mathbb{I}[s = -]\}$ (note that $i' \neq 1$ because $1 \notin S$), we know $v_{i'} = 3$ and by construction, $i' \in \alpha(v)$. This means $S \subseteq \alpha(v)$.

Now we prove the other direction, i.e., if W is achievable by clock auctions, then there is an assignment that satisfies the 3-SAT instance. Let α be a feasible allocation function implementable by clock auctions that guarantees welfare at least W. We first argue that this is possible only when the following are true:

- For any $i \in [a]$,
$$\sum_{j \in \alpha(v^{x_i^+})} v_j^{x_i^+} + \sum_{j \in \alpha(v^{x_i^-})} v_j^{x_i^-} = 6a - 5.$$
- For any $i \in [b]$, $\alpha(v^{c_i}) = \{1\}$.

These conditions ensure that α essentially encodes an assignment of the 3-SAT variables.

To see why the above is true, first observe that the welfare guaranteed by α under these conditions is precisely W. Moreover, it is impossible to get higher welfare by changing α on v^{c_i} for any $i \in [b]$. The only possibility left that W might be achieved without the above conditions holding is if there is some $i \in [a]$ where $\alpha(v^{x_i^+}) = \alpha(v^{x_i^-}) = \{2, \dots, 2a+1\}$, in which case

$$\sum_{j \in \alpha(v^{x_i^+})} v_j^{x_i^+} + \sum_{j \in \alpha(v^{x_i^-})} v_j^{x_i^-} = 6a - 4 > 6a - 5.$$

However, doing so would cause significant welfare loss on other type vectors, and thus sacrifice welfare optimality. In particular, recall that we assume without loss of generality there are at least 10 copies of the clause $x_i^+ \vee x_i^-$ in the 3-SAT instance. Consider any one of these copies c_j. Suppose $\alpha(v^{x_i^+}) = \alpha(v^{x_i^-}) = \{2, \dots, 2a+1\}$. Then $S = \{2i, 2i+1\}$ is a free-riding group at v^{c_j}, because (1) $v^{x_i^+} \geq_{\{2i+1\}} v^{c_j}$ and $2i \in \alpha(v^{x_i^+})$, and (2) $v^{x_i^-} \geq_{\{2i\}} v^{c_j}$ and $2i+1 \in \alpha(v^{x_i^-})$. Since α is implementable by clock auctions, this means $\{2i, 2i+1\} \subseteq \alpha(v^{c_j})$, which means $\alpha(v^{c_j}) \subseteq \{2, \dots, 2a+1\}$. The value we get on type vector v^{c_j} is therefore at most $3a-4$, which is smaller by $1/2$ than the maximum value possible, $3a - 7/2$, obtained when $\alpha(v^{c_j}) = \{1\}$. So, by picking $\alpha(v^{x_i^+}) = \alpha(v^{x_i^-}) = \{2, \dots, 2a+1\}$, we get an improvement of 1 on $v^{x_i^+}$ and $v^{x_i^-}$ combined (not weighted by the prior probability $1/(2a+1)$)

but must also suffer a loss of at least $10 \times 1/2 = 5$ (again, unweighted by $1/(2a+1)$) on all v^{c_j} where c_j is a copy of $x_i^+ \vee x_i^-$, resulting in a net loss of 4. We therefore conclude that the conditions above must hold for α.

Now we can construct an assignment of the variables given α, since the conditions above guarantee that α is structured. For each $i \in [a]$, we let $x_i = +$ if $\alpha(v^{x_i^+}) = \{2, \dots, 2a+1\} \setminus \{2i\}$, and $x_i = -$ otherwise. The goal now is to show α cannot be implementable by clock auctions, and in particular, there exists a clause c_i, with S being the set of agents corresponding to the literals appearing in c_i, such that S is a free-riding group at v^{c_i}, which, given that $\alpha(v^{c_i}) = \{1\}$, means α is not implementable by clock auctions. We establish this claim based on the fact that the corresponding 3-SAT instance is unsatisfiable. In fact, since we have a legitimate assignment of the variables, there must be a clause $c_i = x_{i_1}^{s_1} \vee x_{i_2}^{s_2} \vee x_{i_3}^{s_3}$ (without loss of generality we assume there are 3 literals involved; the case with only 2 literals is similar) such that $x_{i_1} \neq s_1$, $x_{i_2} \neq s_2$, and $x_{i_3} \neq s_3$ in the assignment constructed from α. Consider $S = \{2i_1 + \mathbb{I}[s_1 = -], 2i_2 + \mathbb{I}[s_2 = -], 2i_3 + \mathbb{I}[s_3 = -]\}$. We only need to show S is a free-riding group at v^{c_i}. This is true because for (i_1, s_i), (1) $v^{x_{i_1}^{s_1}} \geq_{S \setminus \{2i_1 + \mathbb{I}[s_1 = -]\}} v^{c_i}$ and (2) $2i_1 + \mathbb{I}[s_1 = -] \in \alpha(v^{x_{i_1}^{s_1}})$ by the two conditions about α and the construction of the assignment. In other words, $2i_1 + \mathbb{I}[s_1 = -]$ free-rides $S \setminus \{2i_1 + \mathbb{I}[s_1 = -]\}$ at v^{c_i}. The same is true for (i_2, s_2) and (i_3, s_3) for similar reasons, and we know S is a free-riding group at v^{c_i}, which contradicts implementability given $\alpha(v^{c_i}) = \{1\}$. This means the only possible way α can guarantee W turns out to be impossible, and there is no way to achieve W using clock auctions. This finishes the proof of Theorem 3. □

Proof of Lemma 3. The proof consists of two parts. We first prove that the first claim implies the second.

Direction $\implies$ *:* Suppose the first claim holds for some $C_1 > 0$, and we want to find $C_2 > 0$ such that the second claim holds. Pick any instance $(m, \mathcal{D}, t)$. If $\Pr_{v \sim \mathcal{D}}[\|v\|_1 \geq t] \geq 1/100$, then the clock auction protocol that always allocates to everyone without increasing any agent's clock price $(1/100, 1)$-extracts the upper tail defined by t. Otherwise, we construct a problem instance for welfare maximization by copy-pasting the upper tail extraction instance.

Let $q = \Pr_{v \sim \mathcal{D}}[\|v\|_1 \geq t]$, and $k = \lfloor 1/q \rfloor$. Let $n = mk$, $\mathcal{F}$ be such that $\mathcal{M}(\mathcal{F}) = \{\{1 + im, \dots, m + im\} \mid i \in \{0, \dots, k-1\}\}$, and $\mathcal{D}' = \mathcal{D}^k$. In particular, for each $S \in \mathcal{M}(\mathcal{F})$, $v_S \sim \mathcal{D}$ when $v \sim \mathcal{D}'$. Let $\mathcal{V}$ be the support of $\mathcal{D}'$. We first argue that the following two quantities: (1) the (expected) first-best welfare in the welfare maximization problem, i.e., $\mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right]$, and (2) the conditional expectation of the total value in the tail defined by t in the upper tail extraction problem, i.e., $\mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t]$, are within a constant factor of each other. This fact will be useful momentarily in both directions of the proof. To see why the two quantities closely bound each other, observe that for any event $\mathcal{E}$ where $\Pr_{v' \sim \mathcal{D}'}[\mathcal{E}] \geq \Pr_{v_S \sim \mathcal{D}}[\|v\|_1 \geq t]$, we must have

$$\mathbb{E}_{v' \sim \mathcal{D}'}[\|v'_S\|_1 \mid \mathcal{E}] \leq \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t],$$

for all $S \in \mathcal{M}(\mathcal{F})$. Let $\mathcal{E}_S$ be the event that $\|v'_S\|_1 > \|v'_{S'}\|_1$ for all $S' \in \mathcal{M}(\mathcal{F}) \setminus \{S\}$. Then $\Pr_{v' \sim \mathcal{D}'}[\mathcal{E}_S] = 1/k \geq \Pr_{v \sim \mathcal{D}}[\|v\|_1 \geq t]$ for all $S \in \mathcal{M}(\mathcal{F})$.[9] As a result,

$$\begin{aligned}\mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] &= \sum_{S \in \mathcal{M}(\mathcal{F})} \Pr_{v' \sim \mathcal{D}'}[\mathcal{E}_S] \cdot \mathbb{E}_{v' \sim \mathcal{D}'}[\|v'_S\|_1 \mid \mathcal{E}_S] \\ &\leq \sum_{S \in \mathcal{M}(\mathcal{F})} \Pr_{v' \sim \mathcal{D}'}[\mathcal{E}_S] \cdot \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t] \\ &= \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t].\end{aligned}$$

On the other hand, one can lower bound the first-best welfare using the tail conditional expectation via the following prophet-style argument (see, e.g., [Alaei, 2014, Chawla et al., 2010] for earlier applications of similar arguments). For $v' \sim \mathcal{D}'$, we iterate through all $S \in \mathcal{M}(\mathcal{F})$ one by one, and "accept" an S with probability $1/2$ if $\|v'_S\|_1 \geq t$; the procedure terminates once we accept a set. Without loss of generality, number these sets $S_1, \ldots, S_k$. Then, the expected total value in the set we accept (which lower bounds the first-best welfare) is

$$\begin{aligned}&\mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] \\ &\geq \sum_{i \in [k]} \Pr[\text{we did not accept } S_1, \ldots, S_{i-1}] \cdot \frac{1}{2} \Pr_{v' \sim \mathcal{D}'}[\|v'_{S_i}\|_1 \geq t] \cdot \mathbb{E}_{v' \sim \mathcal{D}'}[\|v'_{S_i}\|_1 | \|v'_{S_i}\|_1 \geq t].\end{aligned}$$

Taking the union bound, we get $\Pr[\text{we did not accept } S_1, \ldots, S_{i-1}] \leq 1/2$, and therefore

$$\begin{aligned}\mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] &\geq \sum_{i \in [k]} \frac{1}{2} \cdot \frac{1}{2} \Pr_{v' \sim \mathcal{D}'}[\|v'_{S_i}\|_1 \geq t] \cdot \mathbb{E}_{v' \sim \mathcal{D}'}[\|v'_{S_i}\|_1 | \|v'_{S_i}\|_1 \geq t] \\ &= \frac{1}{4} \cdot k \cdot \Pr_{v \sim \mathcal{D}}[\|v\|_1 \geq t] \cdot \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t].\end{aligned}$$

Then, since $k = \lfloor 1/q \rfloor \geq 100$, we have $k \cdot \Pr_{v \sim \mathcal{D}}[\|v\|_1 \geq t] = k \cdot q \geq 100/101$, and

$$\mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] \geq \frac{1}{5} \cdot \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t].$$

To summarize, we have

$$\frac{1}{5} \cdot \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t] \leq \mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] \leq \mathbb{E}_{v \sim \mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t].$$

It is possible to get tighter bounds (see, e.g., Lemma 2 in [Goel et al., 2023]), but for our purposes any constant would suffice.

Now pick a clock auction protocol α' that C_1-approximates the first-best welfare on $(n, \mathcal{V}, \mathcal{D}', \mathcal{F})$. We will construct a clock auction protocol for the upper tail extraction instance with the desired guarantees based on α'. Clearly for each $v \in \mathcal{V}$, $\alpha'(v) \subseteq S$ for some $S \in \mathcal{M}(\mathcal{F})$. So we have

$$\begin{aligned}C_1 \cdot \mathbb{E}_{v' \sim \mathcal{D}'}\left[\max_{S \in \mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] &\leq \mathbb{E}_{v' \sim \mathcal{D}'}[\|v'_{\alpha'(v')}\|] \\ &= \sum_{S \in \mathcal{M}(\mathcal{F})} \Pr_{v' \sim \mathcal{D}'}[\alpha'(v') \subseteq S] \cdot \mathbb{E}_{v' \sim \mathcal{D}'}[\|v'_{\alpha'(v')}\|_1 \mid \alpha'(v') \subseteq S].\end{aligned}$$

[9]Without loss of generality we are assuming $\mathcal{D}'$ is non-atomic. The general case can be easily handled with a bit more care.

Let $\mathcal{M}'$ be the collection of maximal feasible sets S satisfying:

$$\mathbb{E}_{v'\sim\mathcal{D}'}[\|v'_{\alpha'(v')}\|_1 \mid \alpha'(v') \subseteq S] \geq \frac{C_1}{2} \cdot \mathbb{E}_{v'\sim\mathcal{D}'}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v'_S\|_1\right].$$

Then

$$\sum_{S\in\mathcal{M}'} \Pr_{v'\sim\mathcal{D}'}[\alpha'(v') \subseteq S] \cdot \mathbb{E}_{v'\sim\mathcal{D}'}[\|v'_{\alpha'(v')}\|_1 \mid \alpha'(v') \subseteq S] \geq \frac{C_1}{2} \cdot \mathbb{E}_{v'\sim\mathcal{D}'}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v'_S\|_1\right].$$

Furthermore, among all sets in $\mathcal{M}'$, there must be one S^* (pick an arbitrary one if there are many) such that

$$\begin{aligned}\Pr_{v'\sim\mathcal{D}'}[\alpha'(v') \subseteq S^*] \cdot \mathbb{E}_{v'\sim\mathcal{D}'}[\|v'_{\alpha'(v')}\|_1 \mid \alpha'(v') \subseteq S^*] &\geq \frac{C_1}{2|\mathcal{M}'|} \cdot \mathbb{E}_{v'\sim\mathcal{D}'}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] \\ &\geq \frac{C_1}{2k} \cdot \mathbb{E}_{v'\sim\mathcal{D}'}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v'_S\|_1\right].\end{aligned}$$

With this S^* identified, here is how we construct the upper tail extraction protocol:

- Given $v \sim \mathcal{D}$, draw $k-1$ more iid samples $v^{(1)}, \dots, v^{(k-1)}$ from $\mathcal{D}$. Combine these k type vectors (each of length m) into v' of length $n = mk$; make sure (1) $v'_{S^*} = v$, and (2) for all other $S \in \mathcal{M}(\mathcal{F})$, v'_S corresponds bijectively to one of the $k-1$ additional samples drawn from $\mathcal{D}$.
- Run the welfare maximization protocol α' on v' and get $\alpha'(v')$. Return $\alpha(v) = \alpha'(v') \cap S^*$ as the output of the upper tail extraction protocol.

To see why this construction has the desired properties, first observe that it is in fact a (randomized) clock auction protocol for the upper tail extraction problem (we will discuss derandomization momentarily). Moreover, we have $\Pr_{v\sim\mathcal{D}}[\alpha(v) \neq \emptyset] = \Pr_{v'\sim\mathcal{D}'}[\alpha'(v') \subseteq S^*]$, and

$$\mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha(v)}\|_1 \mid \alpha(v) \neq \emptyset] = \mathbb{E}_{v'\sim\mathcal{D}'}[\|v'_{\alpha'(v')}\|_1 \mid \alpha'(v') \subseteq S^*].$$

So by the choice of S^*, we have

$$\mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha(v)}\|_1 \mid \alpha(v) \neq \emptyset] \geq \frac{C_1}{2} \cdot \mathbb{E}_{v'\sim\mathcal{D}'}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] \geq \frac{C_1}{10} \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t].$$

and

$$\Pr_{v\sim\mathcal{D}}[\alpha(v) \neq \emptyset] \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha(v)}\|_1 \mid \alpha(v) \neq \emptyset] \geq \frac{C_1}{2k} \cdot \mathbb{E}_{v'\sim\mathcal{D}'}\left[\max_{S\in\mathcal{M}(\mathcal{F})} \|v'_S\|_1\right] \geq \frac{C_1}{10k} \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t].$$

Let λ and μ be the largest real numbers such that the protocol α constructed above (λ, μ)-extracts the upper tail defined by t. Then we have

$$\lambda = \frac{\mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha(v)}\|_1 \mid \alpha(v) \neq \emptyset]}{\mathbb{E}_{v\sim\mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t]}, \quad \mu = \min\left\{1, \frac{\Pr_{v\sim\mathcal{D}}[\alpha(v) \neq \emptyset]}{\Pr_{v\sim\mathcal{D}}[\|v\|_1 \geq t]}\right\}.$$

Plugging in the bounds above, we get

$$\lambda \geq \frac{C_1}{10}, \quad \lambda \cdot \mu \geq \min\left\{\lambda, \frac{C_1}{10k \cdot \Pr_{v\sim\mathcal{D}}[\|v\|_1 \geq t]}\right\} \geq \min\left\{\frac{C_1}{10}, \frac{C_1}{11}\right\} = \frac{C_1}{11}.$$

To summarize, there is either (1) a trivial protocol that $(1/100, 1)$-extracts the upper tail, or (2) a protocol based on α' that (λ, μ)-extracts the upper tail where $\lambda \cdot \mu \geq C_1/11$. We can therefore conclude that $C_2 \geq \min\{1/100, C_1/11\}$.

One minor issue left is that we might want to derandomize the upper tail extraction protocol α constructed above. We quickly sketch how this can be done. First observe that losing another factor of 2 in $\lambda \cdot \mu$, one can ensure that the protocol for upper tail extraction terminates iff the conditional expected total value among active agents is at least $\frac{\lambda}{2} \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t]$. Then, essentially the protocol only needs to maximize the expected "reward" it collects, where the reward is the conditional expected total value among active agents if this quantity is at least $\frac{\lambda}{2} \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v\|_1 \mid \|v\|_1 \geq t]$, and 0 otherwise. In particular, we know there is a protocol that produces a reward high enough, and at the same time, any protocol that produces a reward high enough must also be a good protocol for the upper tail extraction problem. This is a single objective optimization problem without troubling constraints, and to derandomize a randomized protocol, one can simply fix the random bits in a way that maximizes this objective. In other words, if the first claim holds for potentially randomized clock auction protocols, then the second claim holds for deterministic clock auction protocols. Now we proceed to the other direction of the proof, i.e., the second claim implies the first.

Direction $\Longleftarrow$ *:* Fix any welfare maximization instace $(n, \mathcal{V}, \mathcal{D}, \mathcal{F})$, where $\mathcal{M}(\mathcal{F}) = \{S_1, \ldots, S_k\}$. For each $i \in [k]$, let $\mathcal{D}_i$ be the marginal distribution of v_{S_i} when $v \sim \mathcal{D}$,

$$q_i = \Pr_{v\sim\mathcal{D}}[\|v_{S_i}\|_1 > \|v_{S_j}\|_1, \text{ for all } j \in [k] \setminus \{i\}],$$

and t_i be such that $\Pr_{v\sim\mathcal{D}}[\|v_{S_i}\|_1 \geq t_i] = q_i$. Then, by essentially the same argument as in the first part of the proof, we have

$$\frac{1}{5}\sum_{i\in[k]} q_i \cdot \mathbb{E}_{v\sim\mathcal{D}_i}[\|v\|_1 \mid \|v\|_1 \geq t_i] \leq \mathbb{E}_{v\sim\mathcal{D}}\left[\max_{i\in[k]} \|v_{S_i}\|_1\right] \leq \sum_{i\in[k]} q_i \cdot \mathbb{E}_{v\sim\mathcal{D}_i}[\|v\|_1 \mid \|v\|_1 \geq t_i].$$

Suppose the second claim holds with constant $C_2 > 0$. For each $i \in [k]$, let α_i be a clock auction protocol for the upper tail extraction instance $(|S_i|, \mathcal{D}_i, t_i)$ which (λ_i, μ_i)-extracts the upper tail defined by t_i where $\lambda_i \cdot \mu_i \geq C_2$. Without loss of generality, suppose $\Pr_{v\sim\mathcal{D}_i}[\alpha_i(v) \neq \emptyset] \leq q_i$ (otherwise, conditioned on $\alpha_i(v) \neq \emptyset$, one can randomly set $\alpha_i(v)$ to $\emptyset$ with a properly chosen probability without affecting $\lambda_i \cdot \mu_i$). We construct a clock auction protocol α for the welfare maximization instance which constant-approximates the first-best welfare based on $\{\alpha_i\}$.

The construction is again through a prohpet-style procedure: For $v \sim \mathcal{D}$, we process $S_1, \ldots, S_k$ sequentially. For each S_i, we run α_i on v_{S_i}. If $\alpha_i(v_{S_i}) \neq \emptyset$, then with probability $1/2$, we "accept" S_i (in fact, we accept $\alpha_i(v_{S_i})$ which is what is left in S_i), by raising the price for all other agents to infinity and terminating the auction. When this happens, $\alpha(v) = \alpha_i(v_{S_i})$. Otherwise, we move on to S_{i+1}. We allocate to no one (i.e., $\alpha(v) = \emptyset$) if none of the S_i is accepted in this procedure. The welfare guarantee of this protocol is the following:

$$\begin{aligned}
\mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha(v)}\|_1] &\geq \sum_{i\in[k]} \Pr_{v\sim\mathcal{D}}[S_i \text{ is accepted}] \cdot \mathbb{E}_{v\sim\mathcal{D}}[\|v_{\alpha_i(S_i)}\|_1 \mid S_i \text{ is accepted}] \\
&\geq \sum_{i\in[k]} \left(\sum_{j<i} \frac{1}{2} \Pr_{v\sim\mathcal{D}_j}[\alpha_j(v) \neq \emptyset]\right) \cdot \frac{1}{2} \Pr_{v\sim\mathcal{D}_i}[\alpha_i(v) \neq \emptyset] \cdot \mathbb{E}_{v\sim\mathcal{D}_i}[\|v_{\alpha_i(v)}\|_1 \mid \alpha_i(v) \neq \emptyset] \\
&\qquad \text{(union bound)} \\
&\geq \sum_{i\in[k]} \frac{1}{4} \Pr_{v\sim\mathcal{D}_i}[\alpha_i(v) \neq \emptyset] \cdot \mathbb{E}_{v\sim\mathcal{D}_i}[\|v_{\alpha_i(v)}\|_1 \mid \alpha_i(v) \neq \emptyset] \\
&\qquad (\Pr_{v\sim\mathcal{D}_j}[\alpha_j(v) \neq \emptyset] \leq q_j) \\
&\geq \sum_{i\in[k]} \frac{1}{4} \cdot C_2 \cdot \Pr_{v\sim\mathcal{D}_i}[\|v\|_1 \geq t_i] \cdot \mathbb{E}_{v\sim\mathcal{D}_i}[\|v\|_1 \mid \|v\|_1 \geq t_i] \qquad \text{(choice of } \alpha_i\text{)} \\
&\geq \frac{C_2}{4} \cdot \mathbb{E}_{v\sim\mathcal{D}}\left[\max_{i\in[k]} \|v_{S_i}\|_1\right]. \\
&\qquad \text{(upper bound on first-best welfare via upper tail extraction)}
\end{aligned}$$

In other words, if the second claim holds with constant $C_2 > 0$, then the first claim holds with constant $C_2/4 > 0$. Again, one can derandomize the protocol α constructed above by choosing the optimal random bits, since we are dealing with a single-objective optimization problem without troubling constraints. This finishes the entire proof. □