CONTEXTUALLY PRIVATE MECHANISMS

ANDREAS HAUPT[*]

ZOË HITZIG[†]

[Click here for most recent version.]

Abstract. Consider a mechanism designer who employs a dynamic protocol to implement a choice rule. A protocol is *contextually private* if the designer only learns the private information necessary for computing the outcome. We characterize choice rules that have a contextually private protocol. When the designer is restricted to protocols that sequentially query agents one at a time, we show that few commonly studied choice rules have contextually private protocols. The first-price auction rule is a notable exception—the descending protocol is contextually private. We then partially characterize the class of *maximally contextually private protocols* for choice rules defined on ordered type spaces: protocols in this class pose to each agent a monotonic sequence of queries asking whether the agent's type exceeds a threshold. For the second-price auction rule, we present a protocol that improves on the contextual privacy of the ascending protocol.

1. INTRODUCTION

In mechanism design, agents weigh the expected benefits of revealing their private information against potential costs. Typically, the designer's role is to create the right allocative incentives for agents to voluntarily and truthfully reveal their private information. However, it has long been recognized that *even if* the designer provides the right allocative incentives, agents may *still* be reluctant to reveal their private information (Vickrey, 1961). Agents may worry that their private information will be used against them in subsequent interactions with the designer or third parties (Rothkopf et al., 1990, Ausubel, 2004).

To illustrate these theoretical concerns in practice, consider the following allegation from a recent lawsuit against Google, which takes aim at the company's second-price auctions for advertising:

[*] MIT, Computer Science and Artificial Intelligence Laboratory, haupt@mit.edu.

[†] Harvard, Society of Fellows, zhitzig@g.harvard.edu.

Date: October 20, 2023. We thank especially Eric Maskin, Shengwu Li and Ben Golub. We also thank Yannai Gonczarowski, Jerry Green, Chang Liu, Dirk Bergemann, Stephen Morris, Mohammad Akbarpour, Clayton Thomas, Oghuzan Celebi, David Laibson, Modibo Camara, Katrina Ligett, Benji Niswonger and Max Simchowitz. This work was supported by a Microsoft Research PhD Fellowship (Hitzig).

> Google induced advertisers to bid their true value, only to override pre-set AdX floors and use advertisers' true value bids against them... Google abused advertisers' trust and secretly... generated unique and custom per-buyer floors depending on what a buyer had bid in the past.
>
> State of Texas et al. v. Google, 2022[1]

In short, the plaintiffs alleged that Google used advertisers' losing bids in prior auctions to set personalized reserve prices in future auctions, thereby using information against the participants who supplied it.

The allegation highlights a class of situations that may be particularly worrisome from a privacy perspective: those in which agents are asked to reveal private information that the designer does not actually need in order to carry out its stated plan. A low bidder in a strategyproof auction reveals her exact willingness to pay even though the result of the auction would have been the same if the designer only knew that her type fell below a threshold. In a matching context, a student with a low priority score at desirable schools reveals her entire preference ranking even though she would have ended up at a low priority school irrespective of her preference ranking. In such settings, the designer learns more than is needed—and therefore some agents seem to give up their privacy for no reason.

In this paper, we study a privacy criterion for mechanism design which formalizes the idea that the designer should only learn what is needed. In our set up, when a designer commits to a choice rule, they also commit to a dynamic protocol for computing the choice rule. These dynamic protocols allow the designer to learn agents' private information in a minimal way, ruling out type profiles until the designer knows exactly what is needed to compute the outcome, and nothing more. We call such protocols *contextually private*—the designer's apprehension of agents' private information is justified by the market context.

Formally, the protocols we define are composed of queries which gradually reveal subsets of agents' private information. Each query presents a partition of the type space to agents, and asks the agents to identify in which cell of the partition the true type profile lies. These protocols allow us to accommodate a broad range of social and technological environments. The details of the environment may dictate the format of queries the designer is able to ask—we call the set of available queries the *elicitation technology*.

[1] Civil Action No. 1:21-cv-06841-PKC. Third Amended Complaint, January 14, 2022. Link.

An elicitation technology represents *how* the designer can learn about the subset of the type space in which the true type profile lies. One possible elicitation technology is a "trusted third party." If there is a trusted third party, the designer can delegate information retrieval to the third party, and ask the third party to report back only what is needed to compute the outcome. So, under this trusted third party elicitation technology, all choice rules trivially have a contextually private protocol. Cryptographic techniques like secure multi-party computation and zero-knowledge proofs are elicitation technologies that similarly trivialize contextual privacy.[2]

But there are many environments in which the designer has access neither to a trusted third party nor to a full suite of cryptographic tools. Advanced cryptography may be excessively costly in terms of time, money or computational power.[3] In addition, sophisticated cryptographic mechanisms require sophisticated participants—if participants do not fully understand how their information is kept private, their privacy concerns may not be alleviated.[4]

Motivated by the potential mistrust on the part of agents, we focus on protocols founded on minimal assumptions regarding trust and comprehension. Specifically, for much of the paper we restrict attention to *sequential elicitation* technologies. In a sequential elicitation protocol, the designer is limited to queries directed at individual agents, sequentially asking agents questions about their type. Sequential elicitation protocols neatly expose the privacy properties of a given protocol: for an agent to understand what the designer has learned about her, she merely has to recall her responses to the designer's questions. In essence, her reports to the designer perfectly mirror the information that the designer acquired.

This minimal assumption stands in contrast to more complex assumptions we could make about the set of available elicitation technologies. For example, if the designer's elicitation technology allows them to count the number of agents whose type satisfies a

[2] For a survey of cryptographic protocols for sealed-bid auctions, see Alvarez and Nojoumian (2020).

[3] Even if possible, some sophisticated solutions may be wasteful—in one of the earliest large-scale uses of secure multi-party computation, a double auction with sugar beet farmers in Denmark, designers wondered "if the full power of multiparty computation was actually needed," or if a simpler implementation guided by a weaker privacy criterion may have sufficed (Bogetoft et al., 2009).

[4] A recent survey shows that only 61% of WhatsApp's users believe the company's claim that their messages are end-to-end encrypted (Alawadhi, 2021).

certain property without learning *whose* type satisfies that property, the agents must trust the designer's use of anonymization techniques. If the designer's elicitation technology enables the secret sharing behind secure multiparty computation, agents must grasp the idiosyncratic guarantees and computational assumptions of the particular secure multiparty computation protocol in use.

An additional benefit of sequential elicitation protocols is that they induce a straightforward extensive-form game. Thus, we are able to connect to—and draw on—the growing literature on dynamic mechanism design, especially that on obvious strategyproofness (Li, 2017) and credibility (Akbarpour and Li, 2020). It is a common thread in this literature that the dynamic one-at-a-time implementation of a choice rule more clearly exposes its properties—obvious strategyproofness, defined in such environments, is a form of strategyproofness that is easier to understand. In a similar vein, contextual privacy can be understood as a form of privacy that is straightforward to understand.

To illustrate the key definitions of the paper, we turn to an introductory example. The example considers a simple choice rule and shows that it does not have a contextually private sequential elicitation protocol.

1.1. *Introductory Example*

Suppose there are two agents, 1 and 2, and two objects, A and B. The designer chooses a choice rule and a communication protocol to allocate the objects to the agents. An agent's private information is her preferred object, A or B, which we call her "type". At the start of the protocol, the designer knows that the true type lies in the type space $\{A, B\}^2$. For this type space, we visualize a protocol as a directed rooted tree, labelled by the sets of type profiles that the designer has not yet ruled out. We can visualize the nodes of this tree as 2×2 boxes (see Figure 1). In each box, the rows represent possible types for agent 1 and the columns represent possible types for agent 2. The shaded boxes in Figure 1 represent type profiles that the designer has ruled out with its queries.

An example of a query, shown in Figure 1 is: "Is agent 1's type A?" With this query, the designer learns either that agent 1's type is A or that it is B. These two possibilities correspond to the left and right child, respectively. The left arrow points to a refined type

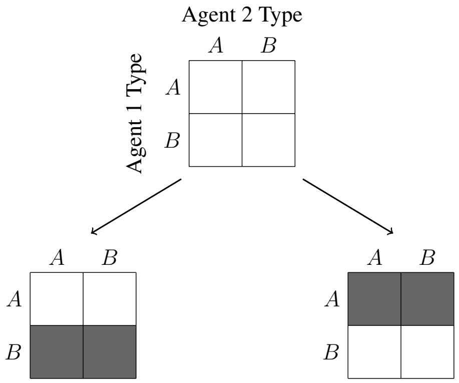


FIGURE 1.—Example: Communication protocol with two agents and two types

space in which the bottom row is ruled out, while the right arrow points to a refined type space in which the top row is ruled out.

A protocol is *contextually private* for a choice rule ϕ if each piece of information revealed through the protocol is necessary for determining the outcome of the choice rule. More precisely, a protocol is contextually private if the designer can only distinguish between possible types of agents in the event that the two types lead to different outcomes, holding other agent types fixed. A choice rule is contextually privately implementable if there is a contextually private protocol that implements it.

To see how choice rules fail to be contextually private, suppose the designer wants to implement a simple efficient choice rule ϕ^{fair}: give each agent their preferred object and break one of the possible ties in agent 1's favor, and the other tie in agent 2's favor. More formally, let (i, x) represent an allocation in which agent $i \in \{1, 2\}$ is allocated object $x \in \{A, B\}$. The choice rule ϕ^{fair} is then

$$\phi^{\text{fair}}(\boldsymbol{\theta}) = \begin{cases} (1, A), (2, B) & \text{if } \boldsymbol{\theta} \in \{(A, B), (A, A), (B, B)\} \\ (1, B), (2, A) & \text{if } \boldsymbol{\theta} = (B, A). \end{cases}$$

The designer cannot implement ϕ^{fair} with a contextually private protocol. To see this, consider again the query that asks "Is agent 1's type A?" (as shown in Figure 1). If the answer is "yes" (left arrow), then the designer knows that the true type profile $\boldsymbol{\theta}$ is either (A, A) or (A, B), and for either of these types, the outcome under ϕ^{fair} is $(1, A)$ and $(2, B)$.

However, if the answer is "no" (right arrow) then the designer learns that the true type is either (B, A) or (B, B). In this event, the designer does not have enough information to compute the choice rule, because the two possible type profiles—(B, A) and (B, B)—result in different outcomes under ϕ^{fair}.

So, in order to compute the choice rule, the designer must pose another query: "Is agent 2's type A?" This second query, combined with the first, allows the designer to distinguish between type profiles (B, A) and (B, B) and therefore compute the choice rule. But, this choice rule violates contextual privacy for agent 2. To see this, suppose the true type profile is (A, B). In this case, agent 1 gets object A and agent 2 gets object B, and the designer knows that agent 2's type is B and not A. But, notice that holding agent 1's type fixed at A, it did not matter whether agent 2 had type A or B, the social outcome would have been $(1, A)$ and $(2, B)$ regardless. The designer did not *need to know* that agent 2 had type A and not B. So, ϕ^{fair} is not contextually private. As we will see, contextual privacy is a demanding criterion, at least if only sequential elicitation protocols are admitted.

1.2. *Overview*

We begin by articulating our formal framework and key definitions in Section 2. We work within a standard mechanism design set up. We depart from the standard set up only to define dynamic protocols. Formally, protocols are finite directed rooted trees in which each node of the tree is associated with a subset of the type space. The subset of the type space associated with the node represents the designer's information at that node.

Our first main result, Theorem 1 in Section 3, is a characterization of contextually private choice rules under generic elicitation technologies. Whether a choice rule admits a contextually private protocol depends on local properties of the choice rule—that is, how the choice rule behaves on small subsets and projections of the type space. We show that this characterization is useful by deriving a simple necessary condition for contextual privacy under a *counting* elicitation technology, in which the designer is only able to ask questions of the form "Is the number of agents with a type θ equal to k," where k is a natural number less than the number of agents.

Next, we restrict attention to sequential elicitation protocols, in which the designer can only query agents one at a time, with a yes-or-no question about their type. In Section 4, after noting that the first-price auction and the serial dictatorship are contextually private, we focus on the limits of contextual privacy under sequential elicitation protocols in environments with and without transfers. We derive a powerful corollary of Theorem 1 called the Corners Lemma. The Corners Lemma is a simple necessary condition for a choice rule to have a contextually private protocol. We use the Corners Lemma to show all of the following negative results: there is no individually rational and efficient rule for the house assignment problem that is also contextually private (Proposition 1); there is no contextually private stable matching rule in matching with priorities (Theorem 2); the second-price auction rule does not have a contextually private protocol (Proposition 2); there is no efficient double auction rule that is contextually private (Proposition 3); and finally that the generalized median voting rule is not contextually private (Proposition 4).

Having seen the limits of contextual privacy under sequential elicitation, we next consider in Section 5 how to design for privacy when a chosen choice rule is not contextually private. Given a protocol, a choice rule, and a type profile, we define the *set of contextual privacy violations* to be the set of type profiles that produce a contextual privacy violation. We define two inclusion orders based on this set. We first consider the *agent-based contextual privacy order* which says that a protocol P is more contextually private than P' if all set of agents who have a violation in P is a subset of the agents who have a violation in P'. Our second major characterization result, Theorem 3, considers choice rules defined on ordered type spaces and shows that every maximally contextually private protocol for such rules is equivalent to a *bimonotonic* protocol. A bimonotonic protocol consists of *threshold queries* which, for each agent, are monotonically increasing or decreasing in the threshold. The commonly studied and used ascending protocol for the second-price choice rule is bimonotonic and a maximal element in the agent-based order, as is the more exotic over-descending protocol defined in Harstad (2018).

Next, we turn to another order, the *type-based contextual privacy order* which says that a protocol P is more contextually private than P' if the set of violations in P is a subset of the violations in P'. We find a necessary condition for protocols that are maximal in this order—they must be *purged* (Proposition 7). This result shows us that the ascending protocol, which is a maximal element in the agent-based order, is not a maximal element in the type-based order. A purged protocol that is more contextually private than the ascending protocol in the type-based order is the *ascending join* protocol,

Sequential elicitation protocols induce a well-defined extensive-form game. In Section 6, we consider the incentive properties of the central protocols discussed in the previous two sections, showing which protocols are dominant-strategy, and obviously-dominant strategy incentive compatible. Here we mostly rely on prior results, using results from Li (2017) to show that the ascending and ascending-join protocols have an equilibrium in obviously dominant strategies.

In Section 7, we explore two modifications of contextual privacy: individual contextual privacy and group contextual privacy. These extensions highlight connections to other concepts such as non-bossiness (Satterthwaite and Sonnenschein, 1981, Pycia and Raghavan, 2022) and (strong) obvious strategyproofness (Li, 2017, Pycia and Troyan, 2023). Finally, we discuss related literature in Section 8.

Proofs omitted from the main text are in Appendix A.

2. MODEL

Consider a set $N = \{1, 2, \dots, n\}$ of agents with private types $\theta_i \sim F$, $F \in \Delta(\Theta)$ that lie in a finite type space Θ. We denote by $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_n) \in \Theta^n = \boldsymbol{\Theta}$ a profile of agents' types. Agents have utility functions u over outcomes in X which depend on their private types, with $u_i : \Theta \times X \to \mathbb{R}$. The designer has a social choice rule $\phi\colon \boldsymbol{\Theta} \to X$ that assigns outcomes in X based on type profiles $\boldsymbol{\theta} \in \boldsymbol{\Theta}$.[5] All primitives of the model besides the realized type profile $\boldsymbol{\theta}$ are common knowledge.

The designer elicits information needed to compute a social choice function by evaluating a series of queries about the type profile. The designer has access to an *elicitation technology*, which specifies the types of queries they can ask about the type space. For instance, the designer may only be able to query a single agent at a time, asking a yes-or-no question about her type. Or, the designer may only be able to form existential queries, ask-

[5] We consider only deterministic choice rules. When we discuss common choice rules, such as the first-price auction choice rule, we will assume that there is deterministic (usually lexicographic) tie-breaking.

ing whether there exists an agent whose type has a certain property. Formally, an elicitation technology $\mathcal{S}$ is a set of partitions of the set of type profiles $\boldsymbol{\Theta}$.

DEFINITION—Protocol: A *protocol* $P = (V, E, r, Z)$ with elicitation technology $\mathcal{S} \subseteq 2^{\boldsymbol{\Theta}}$ is a directed tree with nodes V, edges E, root $r \in V$ and a set of terminal nodes Z. Each non-terminal node u is labelled with a query $s_u \colon \boldsymbol{\Theta} \to \mathrm{children}(u)$, such that $S_v := s_u^{-1}(v) \in \mathcal{S}$ for all $v \in \mathrm{children}(u)$.

Define the *designer's information* at node w to be

$$I_w := \bigcap_{(u,v) \in \mathrm{path}(w)} s_u^{-1}(v),$$

where $\mathrm{path}(w)$ is the set of edges from the root note r to the node w.

We refer to a protocol with elicitation technology $\mathcal{S}$ as an $\mathcal{S}$-protocol, and denote by $\mathcal{P}_{\mathcal{S}}$ the set of protocols with elicitation technology $\mathcal{S}$. Table I compiles notation.

As the protocol progresses from the root node r to the terminal nodes $z \in Z$, the designer learns about the true type profile $\boldsymbol{\theta}$. At the root node, all types are indistinguishable. As the protocol progresses, the designer is able to distinguish some type profiles form others, i.e. they are able to rule out possible type profiles. W

DEFINITION—Distinction: A protocol $P = (V, E, r)$ *distinguishes a type profile* $\boldsymbol{\theta}$ *from* $\boldsymbol{\theta}'$ *at node* $v \in V$ if there are children w, w', with $(v, w), (v, w') \in E$ such that $\boldsymbol{\theta} \in I_w$ and $\boldsymbol{\theta}' \in I_{w'}$.

A protocol must allow the designer to distinguish enough type profiles to compute the choice rule. We say that a protocol P *measures* a social choice rule ϕ if P yields sufficient information to compute the choice rule: For any terminal node z, any distinct type profiles that remain possible at terminal node z lead to the same outcome under ϕ. More formally, P is a protocol ϕ if for all $z \in Z$, for any $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \boldsymbol{\Theta}_z$, it is the case that

$$\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}').$$

In other words, a protocol P is a protocol for choice rule ϕ if ϕ is measurable with respect to the partition $(I_z)_{z \in Z}$. When P is a protocol for ϕ, we sometimes say P is a protocol *for* ϕ.

TABLE I

NOTATION FOR PROTOCOLS AND RELATIONS

Name	Sets	Representative Element
Agents	$\{1, \ldots, n\}$	i
Agent types	Θ	θ
Type profiles	$\boldsymbol{\Theta} = \Theta^n$	$\boldsymbol{\theta}$
Protocols		$P = (V, E)$
Nodes	V	v, w
Edges	E	$e = (v, w)$
Terminal nodes of protocol P	Z	z
Type profiles possible at node v	$\boldsymbol{\Theta}_v$	$\boldsymbol{\theta}_v$
Projection of $\boldsymbol{\Theta}$ onto component i	Θ_i	
	Relations	
Node v precedes v' in protocol P		$v \succ_P v'$
Protocol P' is more contextually private than P for ϕ		$P' \succ_\phi P$
Protocol P' is more agent-contextually private than P for ϕ		$P' \succ^{\text{agent}} \phi P$
Type θ is succeeded by θ' in the type space		$\theta' = \text{succ}(\theta)$ or $\theta = \text{pred}(\theta')$

2.1. *Defining Contextual Privacy.*

Now we present the main definition of the paper: a general privacy criterion that captures the idea that it may be desirable for the designer to employ protocols in which they only learn information that is needed to determine the outcome.

DEFINITION: We say that a protocol P is *contextually private at type profile* $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i})$ *for agent* i if for all θ_i' such that P distinguishes $\boldsymbol{\theta}$ and $(\theta_i', \boldsymbol{\theta}_{-i})$, $\phi(\boldsymbol{\theta}) \neq \phi(\theta_i', \boldsymbol{\theta}_{-i})$. We say that P is *contextually private* if it is contextually private for all agents i at all type profiles $\boldsymbol{\theta} \in \boldsymbol{\Theta}$.

That is, a protocol is contextually private at a given type profile $\boldsymbol{\theta}$ if, for all type profiles that are distinguished from it and differ only in one agent's type, the outcome is different under ϕ. In other words, the fact that agent i has type θ_i and not type θ_i' is information that is necessary for determining the outcome—holding all other types $\boldsymbol{\theta}_{-i}$ fixed, the designer needed to distinguish θ_i from θ'_i to compute ϕ.

This criterion captures the idea that there must be a reason that the designer needed to know whether agent i had type θ_i and not θ'_i, holding other agents' types fixed at $\boldsymbol{\theta}_{-i}$. That reason, in particular, is that without distinguishing θ_i from θ'_i, the designer could not have determined the overall allocation. This definition is illustrated on the right in Figure 2.

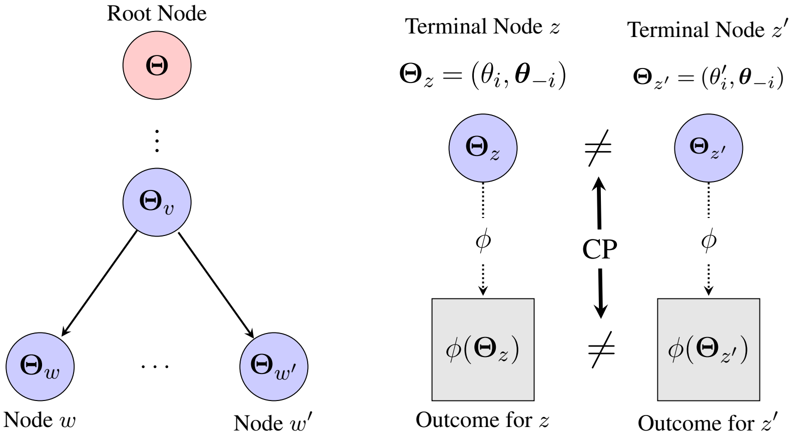

Root Node
$\boldsymbol{\Theta}$
$\boldsymbol{\Theta}_v$
$\boldsymbol{\Theta}_w$
$\boldsymbol{\Theta}_{w'}$
Node w
Node w'
Terminal Node z
$\boldsymbol{\Theta}_z = (\theta_i, \boldsymbol{\theta}_{-i})$
Terminal Node z'
$\boldsymbol{\Theta}_{z'} = (\theta'_i, \boldsymbol{\theta}_{-i})$
$\boldsymbol{\Theta}_z$
$\boldsymbol{\Theta}_{z'}$
$\neq$
CP
$\neq$
ϕ
ϕ
$\phi(\boldsymbol{\Theta}_z)$
$\phi(\boldsymbol{\Theta}_{z'})$
Outcome for z
Outcome for z'

FIGURE 2.—Definitions: Generic protocol (left), contextually private protocol (right)

In many cases, we may be interested in understanding whether there exists a contextually private protocol for a given choice rule. In other words, we might want to know whether a given choice rule has a contextually private protocol. Note that this requires quantifying over a set of admissible protocols—so whether there exists a contextually private protocol of a given choice rule depends on the elicitation technology.

DEFINITION—Contextual Privacy: A choice rule ϕ is *contextually private with elicitation technology* $\mathcal{S}$ if there exists an $\mathcal{S}$-protocol for ϕ that is contextually private.

Contextual privacy is thus a property both of protocols and choice rules—when context does not clearly signal whether we are speaking of the contextual privacy of a choice rule or a protocol, assume that we are talking about a particular protocol. Note that the contextual privacy of a protocol does not depend on the elicitation technology, while the contextual privacy of a choice rule does.

Absent any discussion of strategic considerations, the privacy guarantees cannot be interpreted as being privacy *about agents' private information*. Instead, they are interpreted as privacy with respect to agents' *messages*. We discuss incentives in subsection 2.2 and more deeply in Section 6.

Note that with permissive elictation technologies, all choice rules may be (trivially) contextually private. Contextual privacy only has bite when the designer's elicitation technology is restricted.

In some cases, especially when considering particularly restrictive elicitation technologies S, it will be hard to find contextually private protocols for desired choice rules. In such cases, it might still be possible to make existing protocols more private, and characterize the *most* contextually private protocol for a given choice rule ϕ and elicitation technology S. So, the next set of definitions build toward a contextual privacy order, which allows us to compare the contextual privacy of different protocols.

We will think of privacy improvement via the set of agents who have a *privacy violation* at any type profile $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. This is the set of agents for which there is an alternative type that is distinguished by P but leads to the same outcome under ϕ.

DEFINITION—Contextual Privacy Violations: A contextual privacy violation at $P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}'$ is given if the type profiles $\boldsymbol{\theta}, \boldsymbol{\theta}'$ are adjacent, they yield the same outcome under ϕ, and they are distinguished under P.

$$\Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \iff (\boldsymbol{\theta}_{-i} = \boldsymbol{\theta}'_{-i} \text{ and } \phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}') \text{ and } P \text{ distinguishes } \boldsymbol{\theta}, \boldsymbol{\theta}').$$

DEFINITION—Contextual Privacy Order: We say that a protocol P' is *more contextually private than protocol* P if for all

$$\Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \implies \Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \tag{1}$$

We say that it's more contextually private than protocol P' *if only violated agents matter* if

$$\bigvee_{\boldsymbol{\theta}' \in \boldsymbol{\Theta}} \Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \implies \bigvee_{\boldsymbol{\theta}' \in \boldsymbol{\Theta}} \Gamma(P', \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \tag{2}$$

If (1) holds, we write $P \succ_{\phi} P'$. If (2) holds, we write $P \succ_{\phi}^{\text{agent}} P'$. Both of these introduce partial orders on the set $(\mathcal{P}_{\mathcal{S}}, \preceq_{\phi})$.

We sometimes call the order defined by (1) the *type-based contextual privacy order*, and we call the order defined by (2) the *agent-based contextual privacy order*.[6] For some choice rules, the contextual privacy partial order may have many incomparable objects. As part of our analysis in Section 5, we show that this incomparability amounts to tradeoffs between the privacy violations imposed on different agents at different type profiles.

The privacy orders given by $\prec_{\phi}$ and $\prec_{\phi}^{\text{agent}}$ capture different notions of privacy protection. In the former, the principal would like to minimize the number of privacy violations. In the latter, the principal only cares about whether the privacy is violated for some agent.

2.2. *Incentives.*

We will also consider whether protocols are able to implement social choice functions. We deal only with "direct mechanisms" in the sense that each mechanism we study is one in which agents are queried about their private information ("types"). However, the standard mechanism design framework assumes that in a direct mechanism, agents send a single message to the designer with a report of their type. Meanwhile, in a dynamic protocol, the agent may communicate with the designer multiple times, giving partial reports about her type in response to the designer's queries. The agents' incentives to give truthful partial reports depend on many details of the environment, including the designer's elicitation technology and the information that is revealed to each agent about other agents' sequence of reports.

At the level of generality at which we presented our model, there is no straightforward way to write down agents' incentives to partially report their types truthfully. Their incentives will depend on the elicitation technology and details of the resulting information environment. For example, do the agents know exactly what the designer knows about the

[6] Segal (2007) and Mackenzie and Zhou (2022) consider a related order, the *relative informativeness*, on the information revealed by different extensive-form implementations of choice rules. In Mackenzie and Zhou (2022), the relative informativeness order is invoked to illustrate that menu mechanisms can be less informative than direct revelation mechanisms, while in Segal (2007), the order is employed to identify the communication costs of choice rules.

type profile? Or do the agents know only what the designer has asked them? What does the designer do if deviations by an agent lead to a contradiction, i.e. $I_v = \emptyset$ is reached?[7]

We discuss incentives in the environment studied in Section 4 and Section 5. In this environment, the elicitation technology features only queries asked to one agent at a time, and the agents' responses are public in the sense that each agents' information at node v is I_v, the same as the designers. This specific elicitation technology, together with public information, leads to an extensive-form implementation notion.

DEFINITION: An elicitation technology has *individual queries* if

$$\mathcal{S} \subseteq \{\{\boldsymbol{\theta} \in \boldsymbol{\Theta} : \theta_i \subseteq \tilde{\Theta}_i\} : i \in \mathcal{N}, \theta_i \in \tilde{\Theta}_i\}.$$

For an individual query v we denote by $i(v)$ the agent queried at node v.

Note that for individual queries, I_v consists of cylinder sets, $I_v = \bigtimes_{i \in N} \tilde{\Theta}_i$ where $\tilde{\Theta}_i \subseteq \Theta_i$. This allows us to write $I_{v,i}$ for the projection of the principal's information at node v. Also note that for a query v such that $i(v) = i$, $s_v(\boldsymbol{\theta})$ only depends on θ_i. For brevity, we hence may write $s_v(\theta_i)$.

For individual queries, we can define two notions of incentive compatibility. We will assume that agents observe the full game state, i.e. there is public information. The first is dominant-strategy incentive compatibility.

DEFINITION: We say that a protocol P for ϕ induces *dominant-strategy incentive compatibility under public information* if for all nodes v, $i = i(v)$, and all $\theta_i \in I_{v,i}$,

$$u_i\left(\phi(\boldsymbol{\theta}); \theta_i\right) \geq u_i\left(\phi(\boldsymbol{\theta}'); \theta_i\right)$$

where $\boldsymbol{\theta} \in I_{s_v(\theta_i)}$ and $\boldsymbol{\theta}' \in I_{\text{children}(v) \setminus s_v(\theta_i)}$.

[7] For example, an agent i could report that their type is below x at one point in the protocol and that their type is above x at another point in the protocol. The set of possible type profiles that satisfy both responses is empty.

We will also consider a stronger incentive-compatibility criterion introduced by Li (2017) which requires that the worst outcome following a truthful report is better than the best outcome of a non-truthful report.

DEFINITION: We say that a protocol P for ϕ induces *obviously dominant-strategy incentive compatibility* if for all nodes v, $i = i(v)$, and all $\theta_i \in I_{v,i}$,

$$\inf_{\substack{\boldsymbol{\theta}\in I_w\\ w=s_v(\theta_i)}} u_i\left(\phi(\boldsymbol{\theta});\theta_i\right) \geq \sup_{\substack{\boldsymbol{\theta}'\in I_w\\ w\in\text{children}(v)\setminus\{s_v(\theta_i)\}}} u_i\left(\phi(\boldsymbol{\theta}');\theta_i\right).$$

In this definition, it has to be the case that the worst outcome that can result from telling the truth, i.e. reporting θ_i, is better than the best outcome that can result from misreporting, regardless of what other agents are doing. Notice that when the agent tells the truth, the child node of v is $w = s_v(\theta_i)$. When the agent does not tell the truth, the child node of v is some $w \in \text{children}(v) \setminus s_v(\theta_i)$.

In Section 6, we show that many of the specific protocols showcased in Section 4 and Section 5 also have incentives for truth-telling.

3. CONTEXTUAL PRIVACY UNDER GENERIC ELICITATION TECHNOLOGIES

We first show that contextual privacy allows for concise disproof of contextual privacy, and present necessary conditions that will allow us to analyse the failure modes of contextual privacy in common social choice functions in Section 4.

Determining whether a social choice function is contextually private might be very challenging. In the following characterization, however, we see that showing a choice rule is *not* contextually private may be simpler—we can show that a choice rule is not contextually private by looking at what the choice rule does on small subsets of the type space.

THEOREM 1: *There exists a contextually private protocol for ϕ with elicitation technology $\mathcal{S}$ if and only if there does not exist a subset of type profiles $\hat{\boldsymbol{\Theta}} \subseteq \boldsymbol{\Theta}$ such that*

(i) *$\phi|_{\hat{\boldsymbol{\Theta}}}$ is non-constant, and*

(ii) *for every query s_v with $|\text{range}\, s_v| \geq 2$, there is a type profile $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i})$ and a type profile $\boldsymbol{\theta}' = (\theta_i', \boldsymbol{\theta}_{-i})$ with $s_v(\boldsymbol{\theta}) \neq s_v(\boldsymbol{\theta})$ and $\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}')$.*

PROOF: We first show that contextual privacy implies that no such set $\hat{\boldsymbol{\Theta}}$ exists. We prove this statement in the contrapositive. As $\phi|_{\hat{\boldsymbol{\Theta}}}$ is non-constant (by hypothesis (i)) and P is a protocol for ϕ, the set of nodes that distinguish two type profiles in $\hat{\boldsymbol{\Theta}}$ is non-empty. Let v be any earliest (i.e. minimal in precedence order) node that distinguishes two type profiles in $\hat{\boldsymbol{\Theta}}$. The query at node v, s_v, must have $|\operatorname{range} s_v| \geq 2$ as it distinguishes types in $\hat{\boldsymbol{\Theta}}$. As v is an earliest node that distinguishes type profiles from $\hat{\boldsymbol{\Theta}}$, it must be the case that $\boldsymbol{\Theta}_v \supseteq \hat{\boldsymbol{\Theta}}$. Hence, by the hypothesis (ii) in the statement, there exist $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i})$ and $\boldsymbol{\theta'} = (\theta'_i, \boldsymbol{\theta}_{-i})$ with $s_v(\boldsymbol{\theta}) \neq s_v(\boldsymbol{\theta})$ and $\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta'})$. These constitute a contextual privacy violation.

We next show that the existence of a contextually private protocol implies that for any $\hat{\boldsymbol{\Theta}}$ either (i) or (ii) does not hold. We show this by showing that (i) $\implies \neg$ (ii). That is, either $\phi|_{\hat{\boldsymbol{\Theta}}}$ is non-constant (i), or there exists a non-trivial query that separates two types but does not lead to a contextual privacy violation. Let P be a contextually private protocol. For any non-constant $\hat{\boldsymbol{\Theta}}$ such that $\phi|_{\hat{\boldsymbol{\Theta}}}$ is non-constant, as P is a protocol for ϕ, there must be a query v that separates two type profiles from $\hat{\boldsymbol{\Theta}}$. This query cannot lead to a contextual privacy violation. Hence, $\hat{\boldsymbol{\Theta}}$ cannot satisfy property (ii). *Q.E.D.*

This characterization reduces the characterization of contextual privacy violations to the search for local properties of a social choice function. As the space of protocols is potentially vast (trees of depth up to $|\boldsymbol{\Theta}|$), this is a significant simplification, and allows us to gain insights into local failure modes of contextual privacy. We consider an elicitation technology that is able to count the number of agents who have a given type to make this point.

EXAMPLE—Counting Queries: Consider the following elicitation technology, composed of queries in which the designer asks questions of the form “How many agents' types are in $\tilde{\Theta} \in \Theta$?” Each of these queries has up to $n+1$ children. (The answer to each query could be 0, 1, 2, $\ldots$, n.)

DEFINITION—Counting Queries: The counting elicitation technology S_{count} is

$$\mathcal{S}_{\text{count}} = \bigcup_{\substack{\tilde{\Theta}\subseteq\Theta \\ k\in N}} \{\{\boldsymbol{\theta}\in\boldsymbol{\Theta}\colon |\{\theta_i\in\tilde{\Theta}\}| = k\}\}$$

Suppose a designer only has access to counting queries. Then the characterization theorem immediately gives us a useful necessary condition related to local properties of the choice rule. We can use this necessary condition to more easily search for counterexamples that tell us that a given choice rule does not have a contextually private protocol.

COROLLARY 1—Necessary Condition for Contextual Privacy Under Counting Queries: *Assume that there are mutually distinct* $\theta_i, \theta_i' \in \Theta_i$ *and a partial type profile* $\boldsymbol{\theta}_{-i} \in \boldsymbol{\Theta}_{-i}$ *such that*

$$\phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}). \tag{3}$$

Then, there is no contextually private $\mathcal{S}_{\text{count}}$*-protocol for* ϕ*.*

This means that for all type profiles the outcome needs to be different for any two outcomes.

PROOF: Assume that (3) does not hold. We claim that then $\tilde{\boldsymbol{\Theta}} = \{(\theta_i, \boldsymbol{\theta}_{-i}), (\theta_i', \boldsymbol{\theta}_{-i})\}$ satisfies the requirements in Theorem 1. Clearly, ϕ is not constant on $\tilde{\boldsymbol{\Theta}}$.

In addition, note that any query for which $\operatorname{range} s_v|_{\hat{\boldsymbol{\Theta}}}$ has at least two elements separates the type profiles in $\hat{\boldsymbol{\Theta}}$. As these are different but lead to the same outcome, this set also satisfies the same type profile. *Q.E.D.*

In order for agents to trust that their information is private, they must trust the designer's "anonymization" technology that can de-identify the participants. Such technologies do exist, and are trusted, in both communication and access settings. In access settings, a range of trusted de-identification software programs are used to mask the identities of participants while gathering summary statistics about the sample of the population to which they belong. An example of an everyday anonymous sample technology in a communication setting is a secret ballot. In most elections in the U.S., voters have the right to a secret ballot, which means their vote is not learned by the government, other voters, or third parties.

There may be settings where this is a natural assumption to make. For example, many voting systems use anonymous ballots. But also there may settings where agents only trust themselves.

4. CONTEXTUAL PRIVACY UNDER SEQUENTIAL ELICITATION PROTOCOLS

A sequential elicitation protocol is a protocol in which the elicitation technology allows the designer to learn something about only one agent at a time. Formally, in a sequential elicitation query, the designer can ask whether an agent i's private information θ_i belongs to any subset of the type space $\tilde{\Theta} \subseteq \Theta$.

DEFINITION—Sequential Elicitation Protocol: A protocol $P = (V, E, r)$ is a *sequential elicitation protocol* if for all $v \in V$, $s_v \in \mathcal{S}_{\text{SE}}$, where

$$\mathcal{S}_{\text{SE}} = \{\{\boldsymbol{\theta} \in \boldsymbol{\Theta} \colon \theta_i \in \tilde{\Theta}\} \text{ for some } i \in N, \tilde{\Theta} \in \Theta\}.$$

In other words, the designer can only learn about the i-th component of the type profile at each step.

Although the characterization in Theorem 1 tells us what is contextually private for any elicitation technology, the search for counterexamples can be simplified in the case of sequential elicitation protocols. In particular, as we will see, we can restrict the set of potential "counterexamples" to cylinder sets, and we can use a simplified notion of separability to check for the second condition of Theorem 1.

Instrumental to this analysis will be a notion of *type inseparability.*[8] Roughly, two types for an agent i are *inseparable* if the designer cannot distinguish between them without violating contextual privacy.

[8] Our concept of *inseparability* parallels the concept of *forbidden matrices* used in the work on decentralized computation Chor and Kushilevitz (1989), Chor et al. (1994). Our statement is more general in that it captures agents more than 2 agents.

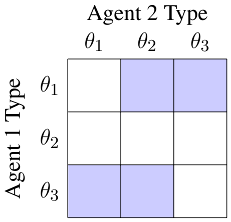


FIGURE 3.—Illustration of Inseparable Types with $n = 2, \boldsymbol{\Theta}' = \{\theta_1, \theta_2, \theta_3\}^2$. Shaded regions represent outcome x under ϕ.For agent 1, $\theta_3 \sim_{1,\phi,\boldsymbol{\Theta}'} \theta_1$. For agent 2, $\theta_1 \sim_{2,\phi,\boldsymbol{\Theta}'} \theta_2 \sim_{2,\phi,\boldsymbol{\Theta}'} \theta_3$.

DEFINITION—Inseparable Types: For a social choice function ϕ, call two types θ_i, θ_i' for an agent i *directly inseparable* on $\boldsymbol{\Theta}'$, denoted $\theta_i \sim'_{i,\phi,\boldsymbol{\Theta}'} \theta_i'$ if there exists $\boldsymbol{\theta}_{-i}$ such that $(\theta_i, \boldsymbol{\theta}_{-i}), (\theta_i', \boldsymbol{\theta}_{-i}) \in \boldsymbol{\Theta}'$, and

$$\phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}).$$

Denote the transitive closure of $\sim'_{i,\phi,\boldsymbol{\Theta}'}$ by $\sim_{i,\phi,\boldsymbol{\Theta}'}$. If $\theta_i \sim_i \theta_i'$, call θ_i and θ_i' *inseparable* for i. We denote equivalence classes under $\sim_{i,\phi,\boldsymbol{\Theta}'}$ by $[\theta]_{i,\phi,\boldsymbol{\Theta}'}$.

We can view inseparability as a necessary condition for contextual privacy when ϕ is evaluated on a subset of type profiles $\boldsymbol{\Theta}'$. Assume that the designer arrives at an interior node v such that $\boldsymbol{\Theta}_v = \boldsymbol{\Theta}'$. Then, a query to agent i that separates θ_i and θ_i' leads to a violation of contextual privacy, as $\phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}$. When the designer learns something that "separates" inseparable types, it learns something that it didn't need to know.

As inseparability will be important for the central characterization of the paper, we build further intuition for this definition. See Figure 3 for an illustration of inseparable types. The 3 × 3 grid represents a subset of the type space in a setting where there are two agents $(n = 2)$. The shaded regions of the grid represent type profiles for which the outcome under ϕ is a particular outcome $x \in X$. Regions of the grid that are not shaded in lead to arbitrary outcomes under ϕ. On $\boldsymbol{\Theta}'$, all of agent 2's types are inseparable. To see this, note that θ_1 and θ_2 are directly inseparable—they lead to the same outcome x when agent 1's type is fixed at θ_3. Furthermore, for agent 2, θ_2 and θ_3 are directly inseparable, since they lead to the same outcome x when agent 1's type is fixed at θ_1. So, since inspeparability is transitive, θ_1, θ_2 and θ_3 are all inseparable for agent 2.

In short, when a choice rule ϕ requires separating inseparable types, contextual privacy is violated. Note that a particular protocol for ϕ may not in fact arrive at an interior node such that $\mathbf{\Theta}_v = \mathbf{\Theta}'$. However, the fact that some product set $\mathbf{\Theta}'$ exists where all contained types are inseperable and ϕ is nonconstant already implies a violation of contextual privacy. The following characterization specializes Theorem 1 to the case of sequential elicitation protocols.

COROLLARY 2—Characterization of Contextually Private Choice Functions Under Sequential Elicitation Protocols: *A choice function ϕ is contextually private if and only if there is no product set $\mathbf{\Theta}'$ such that $\phi|_{\mathbf{\Theta}'}$ is non-constant and for all agents i and all $\theta_i, \theta_i' \in \mathbf{\Theta}_i'$, θ_i and θ_i' are inseparable.*

The full characterization is often not necessary to disprove contextual privacy. A more minimal necessary condition, which is a corollary, dramatically simplifies impossibility results.

COROLLARY 3—Corners Lemma (compare Chor and Kushilevitz (1989)): *Let ϕ be contextually private under sequential elicitation protocols. Then, for any fixed $\boldsymbol{\theta}_{-ij} \in \mathbf{\Theta}_{-ij}$, for all types $\theta_i, \theta_i', \theta_j, \theta_j' \in \Theta$,*

$$\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}) = \phi(\theta_i', \theta_j, \boldsymbol{\theta}_{-ij}) = \phi(\theta_i, \theta_j', \boldsymbol{\theta}_{-ij}) = x \implies \phi(\theta_i', \theta_j', \boldsymbol{\theta}_{-ij}) = x. \tag{4}$$

PROOF: Consider the product set $\mathbf{\Theta}' = \{\theta_i, \theta_i'\} \times \{\theta_j, \theta_j'\} \times \mathbf{\Theta}_{-ij}$. By assumption, θ_i and θ_i', resp. θ_j and θ_j' are directly inseparable, and ϕ is non-constant on $\mathbf{\Theta}'$. Hence, by Corollary 2, ϕ is not contextually private. *Q.E.D.*

The Corners Lemma can be seen as a minimal (in terms of number of agents and types) counterexample to contextual privacy. It is best understood graphically. Hold fixed the types of all agents who are neither agent i nor agent j (i.e. hold $\boldsymbol{\theta}_{-ij}$ fixed). Look only at the $|\Theta| \times |\Theta|$ square that represents the types of any two agents i and j. A 2×2 sub-square is shown in Figure 1. If a choice rule ϕ is contextually private under sequential elicitation protocols, then it must be the case that, when three of the four quadrants in any such square result in outcome x, the fourth quadrant also results in x.

With the Corners Lemma, we prove a number of negative results in different environments. First, we look at assignment domains, then auctions, then voting rules.

4.1. *Assignment.*

In the assignment domain, we fix a set $\mathcal{C}$ of objects. The set of outcomes is $\mathcal{X} = 2^{N\times\mathcal{C}}$, i.e. outcomes are matchings between agents in N and objects in $\mathcal{C}$.

In the standard object assignment setting, agents may receive at most one object, and agents have ordinal preferences over objects, which are private information. So agents' types $\boldsymbol{\theta} \in \Theta$ are preference orders of $\mathcal{C}$ where $\succ_i$ refers to agent i's preference ordering.

In what follows, we use the Corners Lemma to rule out contextual privacy of choice rules, and to illuminate why contextual privacy fails in conjunction with other desiderata. Although most of the results in this section are negative, we do show in Appendix B that the serial dictatorship (with deterministic lexicographic tie-breaking) is contextually private. To see why this is the case, notice that serial dictatorships play nicely with sequential elicitation protocols—every time an agent is asked a question about her type in a serial dictatorship, her assignment is determined.[9]

Now we turn to our applications of the Corners Lemma in two different assignment environments. Consider first the house assignment problem Shapley and Scarf (1974). All agents are initially endowed with an object from $\mathcal{C}$. Denote the initial assignment by an injective function $e\colon N \to \mathcal{C}$, where $e(i) \in \mathcal{C}$ refers to agent i's initial endowment. For our result it will be irrelevant whether the endowments are private information or known to the designer. We call a choice rule ϕ *individually rational* if for all $i \in N$

$$\phi_i(\boldsymbol{\theta}) \succeq_i e(i).$$

PROPOSITION 1: *Assume agents have initial endowments $e(i)$. Then there is no individually rational, efficient and contextually private choice rule under sequential elicitation.*

[9] In fact, this feature of serial dictatorships implies that they are not just contextually private but in fact *individually* contextually private, as discussed in Section 7.

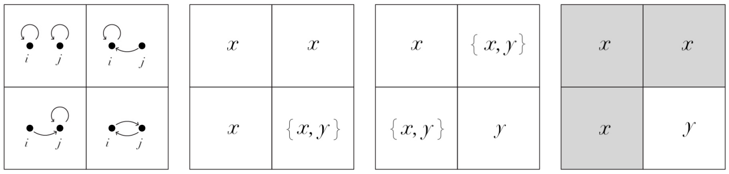


FIGURE 4.—Applying the Corners Lemma for house assignment. Type profiles $\{\theta_i, \theta_i'\} \times \{\theta_j, \theta_j'\}$ used in the proof (left, arrows denote whether agent prefers own or other's endowed object); required outcomes for each type profile under individual rationality (mid-left), under efficiency (mid-right), and under both efficiency and individual rationality (right).

PROOF: The proof uses the Corners Lemma. Consider two agents i and j and two possible preference profiles for each agent. For agent i, consider a type θ_i which contains $e(i) \succ_i e(j)$, and a type θ_i' which contains $e(j) \succ_i e(i)$. For agent j, consider θ_j which contains $e(j) \succ_j e(i)$, and a type θ_j' which contains $e(i) \succ_j e(i)$. Hold fixed all other types $\boldsymbol{\theta}_{-i,-j}$, to be such that they prefer their own endowment to all other objects, i.e. $\boldsymbol{\theta}_{-ij} = \{\boldsymbol{\theta} \in \Theta^{n-2} \colon e(k) \succeq_k c \text{ for all } c \in \mathcal{C} \setminus \{e(k)\}\}$.

When the type profile is $(\theta_i, \theta_j, \boldsymbol{\theta}_{-i,-j})$, the agents both prefer their own endowment to the other's. When the profile is $(\theta_i', \theta_j, \boldsymbol{\theta}_{-i,-j})$, or $(\theta_i, \theta_j', \boldsymbol{\theta}_{-i,-j})$, they both prefer i's endowment and j's endowment, respectively. When $(\theta_i', \theta_j', \boldsymbol{\theta}_{-i,-j})$, they each prefer the other's endowment to their own. Let x be the outcome in which both agents retain their endowment, i.e. $x = (i, e(i)), (j, e(j))$. Let y be the outcome in which each agent gets each other's endowment $y = (i, e(j)), (j, e(i))$. Then, individual rationality makes the requirements shown on the mid-left in Figure 4: $\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-i,-j}) = (\theta_i, \theta_j', \boldsymbol{\theta}_{-i,-j}) = (\theta_i', \theta_j, \boldsymbol{\theta}_{-i,-j}) = x$. Meanwhile, efficiency requires $\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-i,-j}) = y$ (shown on the mid-right in Figure 4). Hence, by the Corners Lemma, no individually rational and efficient choice rule is contextually private under sequential elicitation protocols. *Q.E.D.*

The failure of contextual privacy in the house assignment problem is illuminating. In particular, problems arise because, in an efficient rule, there may be competition for a particular house. In such cases, the designer must elicit information from multiple participants that may not be used.

In two-sided matching, we see a similar failure mode for contextual privacy and stable outcomes under sequential elicitation. In two-sided matching, every agent is matched to at most one object, and at most $\kappa(c)$ agents are matched to school c, for every $c \in \mathcal{C}$, for some *capacities* $\kappa(c)$. That is, the set of outcomes is

$$\mathcal{X} = \{\mu \subseteq N \times \mathcal{C} \colon \forall i \in N \colon |\{c \in \mathcal{C} | (i,c) \in \mu\}| \leq 1 \text{ and } \forall c \in \mathcal{C} |\{i \in N \colon (i,c) \in \mu\}| \leq \kappa(c)\}$$

We say there is *no oversupply* if the aggregate capacity equals the number of agents, $\sum_{c \in \mathcal{C}} \kappa(c) = n$.

We assume that objects have preferences over agents, which are given by *priority scores*. We assume the scores for different objects are private information of the agents. This matches the *college assignment problem* with standardized test scores (Balinski and Sönmez, 1999, Sönmez and Ünver, 2010). We assume that each agent (student) has a vector of *scores* s_c, representing their score at each object (school) $c \in \mathcal{C}$. Objects prefer agents with higher scores. Agent i has private information $\theta_i = (\prec_i, s_i)$, where $\prec_i$ is i's preference ranking over schools, and $s_i \colon \mathcal{C} \to \mathbb{R}$ maps objects to scores.

In such school choice settings, a desirable property of choice rules is *stability* (or *no justified envy*). A choice rule ϕ is *stable* or *induces no justified envy* if there is no blocking pair (i,c), $i \in N$, $c \in \mathcal{C}$ such that $c \succ_i \phi_i(\boldsymbol{\theta})$ and $s_i(c) > s_i(\phi_i(\boldsymbol{\theta}))$.

THEOREM 2: *Assume there are $n \geq 2$ agents, $|C| \geq 2$ objects and no oversupply. Then there is no stable and contextually private choice rule under sequential elicitation.*

The reason why stability conflicts with contextual privacy is relatively easy to see. A stable protocol must "check" for blocking pairs. However, in so doing, the protocol will necessarily check for a blocking pair even when there is none. Consider for example the deferred acceptance protocol, which produces a stable outcome. Consider some tentative assignment in which (i,c) form a pair. Later in the protocol, the designer must elicit information from i to check whether (i,c) forms a blocking pair, and if they do not, then (i,c) becomes a final assignment and the designer has learned something they did not need to know.

4.2. *Auctions.*

We next consider single-item auctions and double auctions. Our results that pertain to the first-price and second-price auctions parallel prior results in a literature on decentralized computation, Brandt and Sandholm (2008).

Consider a standard private values auction environment in which a single indivisible item is to be allocated to one agent. Agents $i \in \mathcal{N}$ with types $\theta_i \in \Theta \subseteq [0,1]$, $|\Theta| < \infty$. The outcomes are given by (q_i, t_i), $q_i \in \{0,1\}$, $t_i \in \mathbb{R}$, where q_i is agent i's allocation, and t_i is their payment. Preferences are defined by

$$u_i((q,t),\theta_i) = \mathbf{1}_{q_i=1} v_i(\theta_i) - t_i, \tag{5}$$

where $v_i : \Theta \to \mathbb{R}$. We call an auction *standard* if there is at most one agent $i \in \mathcal{N}$ such that $t_i \neq 0$ and $q_i = 1$. We call an auction *efficient* if

$$\phi(\boldsymbol{\theta}) \in \operatorname{argmax}_{(q(\boldsymbol{\theta}),t(\boldsymbol{\theta}))} \sum_{i \in N} u_i((q(\boldsymbol{\theta}),t(\boldsymbol{\theta})),\theta_i)$$

for all $\boldsymbol{\theta} \in \boldsymbol{\Theta}$.

The two most widely studied standard auction rules are the first-price and the second-price auction. As this article considers deterministic mechanisms, we consider these rules with deterministic tie-breaking which we without loss assume to be lexicographic. The *first-price auction* is a choice rule $\phi^{\text{FP}}(\boldsymbol{\theta}) = (\phi_1^{\text{FP}}(\boldsymbol{\theta}), \ldots, \phi_n^{\text{FP}}(\boldsymbol{\theta})) = ((q_1,t_1), \ldots, (q_n,t_n))(\boldsymbol{\theta})$, where

$$\phi_i^{\text{FP}}(\boldsymbol{\theta}) = \begin{cases} (1,\theta_i) & \text{if } \theta_i = \min \operatorname{argmax}_{j \in N} \theta_j \\ (0,0) & \text{otherwise.} \end{cases}$$

The *second-price auction* is a choice rule $\phi^{\text{SP}}(\boldsymbol{\theta})$, where

$$\phi_i^{\text{SP}}(\boldsymbol{\theta}) = \begin{cases} (1,\theta_{[2]}) & \text{if } i = \min \operatorname{argmax}_{j \in N} \theta_j \\ (0,0) & \text{otherwise.} \end{cases}$$

Both of these auction rules can be computed via a number of different protocols. Commonly studied protocols for the first-price and second-price rules, respectively, include the descending ("Dutch") protocol and the ascending ("English") protocol. We formally define these (classes of) protocols in Appendix E.

Although the focus of this section is on negative results for contextual privacy via the Corners Lemma, we note first that the first-price auction is contextually private with a descending protocol. We present and discuss this result in Appendix C. The intuition for this result is very similar to the intuition behind the result that the serial dictatorship is contextually private. The designer begins at the top of the type space and asks questions of the form "Is your type above $\tilde{\theta}$?" As soon as one agent answers in the affirmative at some $\tilde{\theta}$, the protocol ends and assigns the object to the agent who responded affirmatively and the price is set at $t = \tilde{\theta}$. The designer thus only elicits information that directly changes the outcome.

Now we turn to negative results for the second-price auction, uniform kth price auctions, and later, uniform price efficient double auctions. The ascending protocol for the second-price is celebrated because it is not only efficient but also strategyproof (Vickrey, 1961, Wilson, 1989), obviously strategyproof (Li, 2017) and credible (Akbarpour and Li, 2020). Paralleling the result (Brandt and Sandholm, 2005, Theorem 4.9) for decentralized protocols, the second-price choice rule does not admit a contextually private protocol. In particular, this result implies that the ascending auction protocol is not contextually private. In fact, the result holds for any uniform kth price auction.

PROPOSITION 2: *Assume $n \geq 3$ agents and $|\theta| \geq 3$. Under sequential elicitation, the second-price choice rule ϕ^{SP} is not contextually private. If $n \geq k+2$, the uniform kth price auction is not contextually private.*

PROOF: The proof uses the Corners Lemma. Consider a type profile with $\theta_{[1]} > \theta_{[4]}$ (or no constraint if $n = 3$), and consider agents i, j that have types in $\underline{\theta}, \bar{\theta}$ such that $\theta_{[4]} < \underline{\theta} < \bar{\theta} < \theta_{[1]}$. Consider the product set $\{\underline{\theta}, \bar{\theta}\} \times \{\underline{\theta}, \bar{\theta}\} \times \bigtimes_{k \in \mathcal{N} \setminus \{i,j\}} \theta_k$. This corresponds to a square depicted in Figure 5.

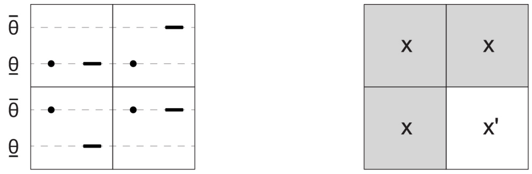


FIGURE 5.—Applying the Corners Lemma to the second-price auction. Type profiles for agent i and agent j (left, agent j's type is represented by a dot, and agent i's type is represented by a dash); required outcome under the second-price auction rule ϕ^{SP}.

Let x be the outcome in which the highest type wins ($q_i = 1$ for $\theta_i = \theta_{[}1]$, $q_i = 0$ otherwise) and pays the price $t_i = \underline{\theta}$. Let x' be the outcome under which the highest type wins and pays the price $t_i = \bar{\theta}$. Then, $\phi^{\mathrm{SP}}(\underline{\theta}, \underline{\theta}, \boldsymbol{\theta}_{-i,-j}) = \phi^{\mathrm{SP}}(\underline{\theta}, \bar{\theta}, \boldsymbol{\theta}_{-i,-j}) = \phi^{\mathrm{SP}}(\bar{\theta}, \underline{\theta}, \boldsymbol{\theta}_{-i,-j}) = x$. But, under ϕ^{SP}, it must be the case that $\phi(\bar{\theta}, \bar{\theta}, \boldsymbol{\theta}_{-i,-j}) = x'$. Since $x \neq x'$, the Corners Lemma is violated, and thus the second-price choice rule is not contextually private under sequential elicitation.

An analogous construction is possible for the k-th price auction by considering agents i, j with types $\underline{\theta}, \overline{\theta} \in (\theta_{[k-1]}, \theta_{[k+2]})$. *Q.E.D.*

We show in Subsection A.3 that this impossibility holds even if ties are ruled out.

We conclude this section with a similar impossibility for standard double auction price rules. Suppose m agents are buyers and m agents are sellers, and $n = 2m$. The m sellers are each endowed with one homogeneous, indivisible object. The buyers have unit demand for objects. Formally, agents have initial endowments $e(i) \in \{0, 1\}$ where $e(i) = 0$ for buyers and $e(i) = 1$ for sellers. The preferences are

$$u_i((q,t),\theta) = -e(i)v_i(\theta_i)q_i + (1 - e(i))v_i(\theta_i)q_i + t_i.$$

A double auction price rule seeks to find a price t that maximizes $\sum_{i \in \mathcal{N}} u_i((q,t),\theta_i)$ if buyers with types $\theta_i \geq t$ buy a good at price t, and sellers with value $\theta_i \leq t$ sell their good at price t. Agents with $\theta_i = t$ sell or buy in order to match supply to demand.

PROPOSITION 3: *Assume there are $n > 3$ agents. There is no efficient, uniform-price contextually private double auction price rule under sequential elicitation protocols.*

The main observation for this statement is that efficient price rules set prices that are medians of the empirical type distribution, $\phi(\boldsymbol{\theta}) \in [\theta_{[m]}, \theta_{[m+1]}]$. We prove that medians may not be computed in a contextually private way under sequential elicitation. To see why this is the case, note that if there are two or more agents at the median value, then, when either agent changes her report, the outcome doesn't change—i.e. a single deviation cannot change the outcome. However, a double deviation, where both agents change their report, the outcome does change. So under sequential elicitation, the designer must ask both agents about their types in order to compute the rule. But then, in so doing, the designer may learn something they did not need to know.

4.3. *Voting.*

We next turn to a voting environment. There is an ordered set of social outcomes X. Preferences are single-peaked $\prec_i$ with peaks $\theta_i \in X$.[10]

We consider a commonly studied class of voting rules. Namely, we study *generalized median voting rules*. As shown in Moulin (1980), this class is the class of all anonymous, strategy-proof and Pareto-efficient voting rules, where anonymity means that the outcome cannot depend on the identity of any agent. A generalized median voting rule takes as input submitted peaks of agents' preferences $\theta_1, \theta_2, \ldots \theta_n$ as well as *phantom ballots* $k_1, k_2, \ldots, k_{n-1} \in X \cup \{-\infty, \infty\}$. The output is the median of the submitted votes and phantom votes, i.e.

$$\phi_{(k_1,k_2,\ldots,k_{n-1})}(\boldsymbol{\theta}) = \operatorname{median}(\theta_1, \theta_2, \ldots, \theta_n, k_1, k_2, \ldots, k_{n-1}).$$

PROPOSITION 4: *For any $k_1, k_2, \ldots, k_{n-1} \in X$ such that neither $\sup(k_1, k_2, \ldots, k_{n-1}) \neq -\infty$ nor $\inf(k_1, k_2, \ldots, k_{n-1}) \neq \infty$, the generalized median voting rule $\phi_{(k_1,k_2,\ldots,k_{n-1})}$ is not contextually private under sequential elicitation.*

The case where $\sup(k_1, k_2, \ldots, k_{n-1}) = -\infty$ and $\inf(k_1, k_2, \ldots, k_{n-1}) = \infty$ makes the choice rule $\inf(\theta_1, \theta_2, \ldots, \theta_n)$ resp. $\sup(\theta_1, \theta_2, \ldots, \theta_n)$, which indeed have a contextually

[10] A preference order $\prec_i$ is single-peaked with peak θ if $x \prec_i \theta \prec_i \theta'$ for $x < \theta < \theta'$ and $x \succ_i \theta \succ_i \theta'$ if $x > \theta > \theta'$.

private protocol. For any other cases, we use the Corners Lemma. Note that this result connects closely to Proposition 3—the negative result for the double auction. In the course of proving the result for the double auction, we showed that there is no contextually private protocol for computing a median of an even number of bids. For general median voting rules, an argument for an odd number of bids is needed.

To summarize, this section first presented a corollary of Theorem 1: a characterization of choice rules that have a contextually private protocol when the designer is restricted to sequential elicitation protocols. Then, we presented a useful corollary of Theorem 1, the Corners Lemma, and used it to prove that many commonly studied choice rules are not contextually private under sequential elicitation protocols. We considered assignment, auction, and voting domains. There are two notable choice rules that *are* contextually private under sequential elicitation: the serial dictatorship and the first-price auction. In Appendix B, we show that the serial dictatorship is contextually private. In Appendix C we show that the first-price auction is contextually private with a “descending" or “Dutch" protocol.

5. BEYOND CONTEXTUAL PRIVACY

We will now consider the design for privacy if contextual privacy is unattainable, e.g., if counterexamples like in the Corners Lemma exist. We first consider the design of mechanisms under $\prec_{\phi}^{\text{agent}}$, showing that if we are searching for maximal protocols with respect to this order, it is without loss to restrict to *bimonotonic* protocols. We show that common protocols for social choice functions studied in section 4.2 and 4.3 are bimonotonic. We also show that they cannot be improved under this $\prec_{\phi}^{\text{agent}}$. We go on to show that the same does not hold for $\prec_{\phi}$: The ascending auction, for example, is bimonotonic, but permits a privacy improvement in the order $\prec_{\phi}$. The privacy improvement leads to the *ascending-join protocol*.

Throughout the section, we consider ordered type spaces $(\Theta_i, \leq)$. This captures all settings from Subsection 4.2 and 4.3. Compatibility of the order of queries with the order of type spaces will be the main tool of analysis.

5.1. *Only the Existence of Privacy Violations Matters.*

Under the equivalence relation induced by the order $\prec_\phi^{\text{agent}}$, $\sim_\phi^{\text{agent}}$, all equivalence classes of *interval pivotal* social choice functions have special representatives. We first define interval pivotality.

DEFINITION—Interval Pivotality: A social choice function ϕ exhibits *interval pivotality* if for all $\boldsymbol{\theta}_{-i} \in \boldsymbol{\Theta}_{-i}$ there are elements $\underline{\theta} \in \Theta$ and $\overline{\theta} \in \Theta$ such that

$$\phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}) \iff (\theta_i, \theta_i' \leq \underline{\theta} \text{ or } \theta_i, \theta_i' \geq \overline{\theta}).$$

This property of choice rules states that, holding fixed some $\boldsymbol{\theta}_{-i}$, the choice rule is constant in θ_i if and only if θ_i is outside some interval $[\underline{\theta}, \overline{\theta}] \in \Theta$. So, inside the interval, i is "pivotal" in that her report changes the outcome. Note in this definition that the interval defined by $[\underline{\theta}, \overline{\theta}]$ depends on the profile of other agents' types, $\boldsymbol{\theta}_{-i}$.

Many known social choice rules are interval pivotal: Any k-th price auction, the Walrasian auction, and all generalized median voting rules. That is, all social choice functions considered in Subsections 4.2 and 4.3.

To define the set of representatives from equivalence classes, we define the elicitation technology of *threshold queries*. Denote the set $S_{\text{thresh}} = \{\{\theta_i \leq \tilde{\theta}\} | \tilde{\theta}_i \in \Theta_i\} \cup \{\{\theta_i > \tilde{\theta}\} | \tilde{\theta}_i \in \Theta_i\}$. For an S_{thresh}-protocol

DEFINITION—Bimonotonic Protocol: A $\mathcal{S}_{\text{thresh}}$-protocol for ordered type spaces $(\Theta_i, \leq)$, $i = 1, 2, \ldots, n$ is *bimonotonic* if for any path in P, the sequence of thresholds to agent i form an increasing or decreasing interval in Θ_i.

The following result reduces the search for protocols, and means that, after an initial query, all following queries need to be ascending or descending, depending on the response to the initial query.

THEOREM 3: *Let P be a protocol for ϕ, where ϕ exhibits interval pivotality. There is a bimonotonic protocol P' that is exactly as contextually private as P.*

The proof of Theorem 3 proceeds by considering any protocol and applying a set of modifications that preserve the set of contextual privacy violations. Up to a technical operation which we call *anchoring*, a first operation allows "fill in" any wholes between two separated agent types in queries to a same agent. A "deduplication" operation allows to only keep monotonic threshold queries.

Several common protocols are bimonotonic and presented in Appendix E: The ascending protocol for the second-price auction, the descending protocol for the first-price auction, and the overdescending protocol for the second-price auction. While the ascending protocol for the second-price choice rule and descending protocol for the first-price choice rule are both standard (corresponding to open "English" auctions and "Dutch" auctions respectively), the overdescending protocol for the second-price choice rule is more exotic. To our knowledge, the overdescending protocol for the second-price choice rule is discussed only in Harstad (2018): it is the same as a descending protocol, except when the highest bid is reached, it continues to descend until it reaches the second-highest bid, which sets the price. We will discuss these protocols in depth in Section 6, when we consider their incentive properties.

For voting rules, one-sided protocols are bimonotonic: First, query all agents whether they are below the right-most alternative, then query the next-to-rightmost alternative, *etc.* These protocols, as the auction protocols, fall into the class of bimonotonic protocols.

Having a structure like bimonotonicity in place shows that the choices that a designer makes around privacy are around the initial queries (ascending, descending, or from an intermediate type) and the order in which questions are asked.

Although the class of maximally contextually private protocols includes protocols resemble those used in practice and studied elsewhere, there are also many other maximally contextually private protocols which do not resemble anything (to our knowledge) used in practice or studied in theory. For example, there are maximally contextually private protocols based on the principle of *guessing*.

EXAMPLE—Guessing Protocols: The *guessing protocol* for the second-price auction can be seen as guessing the correct price $p \in \Theta$ and verifying that it is the second-highest bid. The protocol starts with a query $1_{\theta_i \geq \tilde{\theta}}$ for $\tilde{\theta} = \mathrm{succ}_{\Theta}(p)$ for all agents. If a single agent is higher than the query, no ascending questions are asked anymore.

We claim that for a type profile in which the second-highest bid is p, that is, the guess is correct, only a single agent's privacy is violated. Denote this type profile $\boldsymbol{\theta}$ and the terminal node following it by z. Then,

$$I_z = \{\boldsymbol{\theta} \mid \exists i, j \in N : \theta_i \geq p, \theta_j = p, \boldsymbol{\theta}_{-ij} \leq p\}.$$

That is, the guessing auction learns one agent's exact type, but otherwise only what's needed to compute the outcome. Compare this to the privacy violations of the ascending auction at $\boldsymbol{\theta}$: In this case the privacy of all agents except the winner i is violated. For the overdescending protocol, the privacy of i and j are violated. Hence, at this type profile, both the ascending protocol and the overdescending protocol have more agent violations than the guessing protocol.

On the other hand, if the guess is wrong, the guessing protocol might lead to more violations than either the ascending or overdescending protocols. If all types are smaller than p, the protocol will be an over-descending protocol on the restricted type spaces $\{\theta \in \Theta \mid \theta \leq p\}$, which will violate the privacy of the winner and second-highest agent. At such type profiles, the set of violations are incomparable with the ascending auction. If all types are above p, the protocol amounts to an ascending protocol on the restricted type space $\{\theta \in \Theta \mid \theta \geq p\}$, violating the privacy of all the losers, which is incompatible with the privacy violations of the overdescending auction.

This example shows that maximal contextual privacy with respect to the order on agent violations highlights tradeoffs among privacy violations for different agents.

We next consider a finer order based on the set of contextual privacy violations.

5.2. *Exact Privacy Violations Matter.*

One reason why the ascending protocol might be considered not maximally private is that it contains apparently superfluous queries: After asking an agent whether they are at least type $\tilde{\theta} \in \Theta$, why are the other agents asked as well? We first capture the deletion of superfluous queries, and show that this leads to a maximally contextually private protocol for the second-price auction.

DEFINITION—Superfluous Query: Let P be a protocol for ϕ. We say that a query $v \in V(P)$ is ***superfluous*** if there is a child $w \in \text{children}(v)$ such that the protocol $\text{Purge}_v(P)$ that attaches $\text{subtree}(w)$ to $\text{parent}(v)$ is a protocol for ϕ.

Purging weakly reduces the set of contextual privacy violations, as a purged protocol does not distinguish any type profiles that were not already distinguished in the original protocol.

PROPOSITION 5: *Let $P' \in \text{Purge}_v(P)$ for a superfluous query $v \in V(P)$. Then, $P' \prec_\phi P$.*

We will build toward an inductive purging process. To do so, we first note that as long as a non-superfluous query v' comes before a superfluous query v in the protocol tree, then purging v cannot make v' superfluous. We capture this observation in the following proposition.

PROPOSITION 6: *Let P be a protocol for ϕ. If $v \in V(P)$ is superfluous, a non-superfluous query $v' \succ_P v$ in P cannot be superfluous in $\text{Purge}_v(P)$.*

PROOF: We will denote I_v^P the principal's knowledge at node v in protocol P. Note that for any $I_{v'}^{\text{Purge}_v(P)} \subseteq I_{v'}^P$, and all queries further after v' are the same. By definition of superfluity, a query cannot become superfluous in $\text{Purge}_v(P)$. *Q.E.D.*

We hence may inductively purge in reverse topological order[11] to obtain a purged protocol P.

DEFINITION—Purging: Let S be any elicitation technology. $\text{Purge}\colon \mathcal{P}_S \rightrightarrows \mathcal{P}_S, P \mapsto \text{Purge}(P)$ is the union of the outcomes of any purging for any reverse topological order of P.

[11] Recall that for a directed acyclic graph G, a reverse topological order $\prec_{P,\text{top.}}$ is a total order such that $v \succ_P v'$ implies $v \prec_{P,\text{top.}} v'$.

This correspondence has the property that purging a purged protocol yields the same protocol, i.e.

$$\mathrm{Purge}(\mathrm{Purge}(P)) = \mathrm{Purge}(P).$$

It is hence meaningful to consider the *set of purged protocols* $\mathrm{Purged}_{\mathcal{S}} \coloneqq \{P \in \mathcal{P}_{\mathcal{S}} : P = \mathrm{Purge}(P)\}$.

PROPOSITION 7: *Let P be a maximally contextually private for ϕ that does not contain superfluous queries (i.e. queries s_v with $|s_v(I_v)| = 1$). Then P is purged for ϕ. The reverse implication does not hold.*

PROOF: First, we show that the reverse statement does not hold. Consider an ascending protocol for the first-price auction with lexicographic tiebreaking ϕ^{FP}. Agents are queried in order $1, 2, \ldots, n$ and queries for each $\tilde{\theta}$ in increasing order whether $\{\theta_i \geq \tilde{\theta}\}$. After a negative answer, they are not asked again. Note that this protocol violates privacy with its initial query: It is not necessary to learn that the type is not the lowest type for the lowest agent. As the descending protocol for the first-price auction is contextually private (as shown in Appendix C), and hence has no contextual privacy violations, the ascending protocol for ϕ^{FP} is not maximally contextually private.

However, the ascending protocol for ϕ^{FP} is purged. To see this, consider any node v with query $\mathbf{1}_{\{\theta_i \geq \tilde{\theta}\}}$ with to an agent i, and consider the protocol in which each of the children w, w' are attached to $\mathrm{parent}(v)$.

Case 1 (Affirmative answer). In this case, the protocol proceeds *as if* the answer had been affirmative. Consider a type profile in which the "correct" answer is negative. Consider a type profile in which all of the following queries are answered negatively. The protocol terminates, but it is not clear whether $\theta_i = \mathrm{pred}_{\Theta}(\tilde{\theta})$ or $\theta_i = \tilde{\theta}_i$. The price at which the good is sold is $\mathrm{pred}_{\Theta}(\tilde{\theta})$ in the former case, and $\tilde{\theta}_i$ in the latter case, hence meaning that the purging is not possible.

Case 2 (Negative answer). In this case, the protocol proceeds *as if* the answer had been negative. Assume it was positive and that the queried agent has the highest type. In this case, there is no way to know the highest type, and hecne the price at which the good is sold.

Together, Case 1 and Case 2 show that the ascending protocol for the first-price auction is purged.

Now we prove the main statement: If P is maximally contextually private for ϕ and does not contain superfluous queries, then it is purged for ϕ. We prove the contrapositive. Let v be a purgeable query to agent i and $(\theta_i, \boldsymbol{\theta}_{-i})$, $(\theta_i', \boldsymbol{\theta}_{-i})$ be two type profiles that are distinguished at v and not later. Such queries must exist as there are no superfluous queries in the protocol. As the protocol is purgeable, the distinction of $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ cannot have been necessary for computing ϕ. Hence, $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ must constitute a privacy violation that $\mathrm{Purge}_v(P)$ does not have. *Q.E.D.*

While in general, purging does not lead to maximally contextually private protocols, this holds for the ascending protocol.

PROPOSITION 8: *The ascending protocol with lexicographic tiebreaking is not purged for $n \geq 3$, and hence not contextually private. Algorithm 1 is a purging of the ascending protocol.*

Algorithm 1: Ascending Join Protocol

```
Input: Ordered list $n$ of agents , each with willingness to pay $\theta_i \in \Theta$
Output: Agent $i$ and price $\tilde{\theta}$
Data: $\tilde{\theta} \leftarrow \min(\Theta)$
Data: $\mathrm{active} \leftarrow$ first two agents from $N$
while true do
    foreach agent $i$ in $\mathrm{active}$ do
        if $\theta_i < \tilde{\theta}$ then
            Remove $i$ from $\mathrm{active}$;
            if There is agent $i$ that never has been active then
                $\mathrm{active} \leftarrow \mathrm{active} \cup \{i\}$;
            else
                return Remaining agent $i$ from active, $\mathrm{pred}_\Theta(\tilde{\theta})$
```

PROOF: To see that the ascending protocol is purgable, consider the third agent that is asked for their lowest type. Purging this query with the subtree following an affirmative answer will still allow to compute the second-price auction choice rule. The ascending join auction is a purging of the ascending auction as it deletes all queries for the same type $\tilde{\theta}$ after the first two agents have been queried.

Q.E.D.

The join protocol maintains a set of two agents that are *active*. The threshold price is raised and the two active agents are queried for whether they are willing to pay the price. If one agent does not answer in the affirmative, they are removed from the active agents. If only one agent remains, the price and winner are determined.

6. CONTEXTUAL PRIVACY AND INCENTIVES

So far, we have focused on whether choice rules can be computed through a contextually private protocol, without considering the strategic aspects of the induced messaging game. As noted in subsection 2.2, for generic elicitation technologies, it is not straightforward to write down the strategic game induced by a particular protocol. With generic elicitation technologies, multiple agents may be queried at a time—the strategic properties of such protocols will depend on details of the environment, like whether agents can communicate with each other before submitting their responses.

However, the extensive-form game induced by a sequential elicitation protocol is standard, assuming that at every node in the protocol the designer's information is public. We can thus study the conjunction of contextual privacy and implementation, i.e. whether there are (maximally) contextually private protocols that are also dominant strategy or Bayesian incentive compatible. In fact, given that the extensive-form games induced by sequential elicitation protocols have been studied elsewhere, we can rely largely on existing results to show that the (maximally) contextually private protocols characterized by Theorem 3 in fact *implement* the corresponding choice rule.

Throughout this section, we make a standard assumption about agents' private information: their types are independent and identically distributed according to a prior f with cumulative distribution function F. The designer and the agents share a common prior. We call this information environment *independent private values*.

6.1. *The Extensive-Form Game Induced by Sequential Elicitation.*

In order to study implementation, we must first formally define the game environment. The game defined by the sequential elicitation protocol is direct in the sense that the agents' messages to the designer consist of subsets of the type space. At every non-terminal node $v \in V$, an agent can choose among $\mathrm{children}(v)$. We call the report $s_v(\theta_i) \in \mathrm{children}(v)$ the *truthful report*. Formally, the agents' actions are reports about in which cell of the partition defined by s_v their type profile lies. To simplify the analysis, we will assume that the protocol does not contain *redundant queries* which have only one possible answer under truthtelling, $|s_v(I_v)| = 1$.

An agent's *interim strategy* $\sigma_i(\cdot)$ selects a report $w \in \mathrm{children}(v)$. Let Σ_i denote agent i's interim strategies. An *ex-ante* strategy is a function that maps types to interim strategy profiles $\boldsymbol{\sigma}_i \colon \Theta \to \Sigma_i$.

Note that if a protocol does not contain redundant queries, agents' ex-ante strategies cannot generate contradictions. At every node, there are answers that lead to a non-empty set of principal knowledge I_v.

6.2. *Descending Protocols: Incentive-Compatibility via Uninformativeness.*

We consider the over-descending protocol, which is a maximally contextually private protocol for the first-price auction. As a reminder, the over-descending protocol begins at the top of the type space and asks each agent whether her type is above some $\tilde{\theta}$—when one agent responds in the affirmative, the protocol continues to descend until another agent responds in the affirmative, at which point the protocol stops. We show that the over-descending protocol has an equilibrium ϕ^{SP} in obviously dominant strategies.

PROPOSITION 9: *The over-descending protocol for ϕ^{SP} has an equilibrium in dominant strategies, and is a maximally contextually private protocol for ϕ^{SP}.*

The over-descending protocol, introduced in Harstad (2018), has not been widely studied.[12] The main reason why it is dominant-strategy incentive compatible is straightforward, but there is some subtlety due to the publicity of information. Consider two phases of the protocol: phase 1 consists of the queries up to when the highest type agent answers in the affirmative, and phase 2 consists of the queries that follow to search for the second-highest type. In phase 1, it is straightforward to see that agents would like to answer affirmatively at their true type to maximize their probability of winning—since they do not set their own price, and they learn nothing about other agents' types other than the fact that they are below the going threshold, the incentives to report truthfully in this phase are the same as the incentives in the static second-price auction. Then, in phase 2, the agents who remain in the protocol know that they will not get the good, so they are indifferent between a truthful report and all other reports.[13]

6.3. *Ascending Protocols: Strong Incentive Guarantees.*

We begin by showing that the ascending protocol obviously dominant-strategy implements the second-price auction choice rule. Then we show that the ascending-join protocol, which improves the privacy of the ascending protocol, also has an implementation in obviously dominant strategies.

PROPOSITION 10: *The ascending and ascending-join protocols for ϕ^{SP} have equilibria in obviously dominant strategies and are maximally contextually private for ϕ^{SP}.*

To show that this holds, we show that the ascending protocol we have defined is equivalent to a *personal-clock auction*. Therefore, we can rely on the characterization in Li (2017, Theorem 3), which states that for every personal-clock auction there is an allocation rule that it has an equilibrium in in obviously dominant strategies.

[12] While Harstad (2018) notes that the overdescending protocol is incentive-compatible, it does not offer a formal treatment. Furthermore, the information environment is different, in Harstad (2018) it is assumed that bids are not public.

[13] In practice, it may be desirable to not announce publicly when the highest bid has been found—in which case the second-highest bidder strictly prefers to reveal her true type instead of weakly preferring to do so. Indeed, this is the assumption in Harstad (2018).

Note that not all maximally contextually private protocols for the second-price auction have such strong incentives for truth-telling. Above, we noted that the over-descending auction, which is also maximally contextually private for the second-price auction, has an equilibrium in which at all nodes, it is a dominant strategy to respond to the designer's queries truthfully, but this truthtelling is not *obviously* dominant. Furthermore, there are maximally contextually private protocols for the second-price auction that do not have dominant strategy truth-telling equilibria.

7. VARIATIONS

In this section, we consider two concepts that strengthen contextual privacy. We explore these stronger concepts for both theoretical and practical reasons. On the practical side, these extensions may have desirable properties in some settings. On the theoretical side, these criteria help to illuminate connections to other desiderata in mechanism design, and illustrate which of our results are robust to alternative formulations of contextual privacy. In this section, we revert to the assumption maintained throughout Section 4 that the designer has access only to sequential elicitation protocols.

7.1. *Individual Contextual Privacy*

The first extension, *individual contextual privacy*, requires that if two types are distinguishable for agent i, these types must lead to different outcomes under ϕ *for agent* i. This definition thus only applies in domains where the outcome space $\mathcal{X}$, specifies an allocation for each agent $i \in N$.[14] Let $\phi_i(\boldsymbol{\theta})$ denote the projection of the outcome vector $\phi(\boldsymbol{\theta})$ onto the ith component.

DEFINITION—Individually Contextually Private Protocols: A protocol $P = (V, E)$ for a social choice function ϕ is *individually contextually private* if for all terminal nodes $z, z' \in Z$ and all type profiles $(\theta_i, \boldsymbol{\theta}_{-i}), (\theta_i', \boldsymbol{\theta}_{-i})$ that are distinguished by P,

$$\phi_i(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi_i(\theta_i', \boldsymbol{\theta}_{-i}). \tag{6}$$

[14] The domains that we focus on in this paper—auction domains and assignment domains—both satisfy this property. Any domain with transfers also satisfies this property. However, voting rules, for example, do not have this property—there is a single social outcome, without individualized allocations.

Notice that individual contextual privacy is stronger than contextual privacy—any choice rule that is contextually private is also individually contextually private. If there were an agent i for whom contextual privacy were violated, then individually contextual privacy would automatically be violated.

As a normative criterion, individual contextual privacy requires that if the designer can distinguish between two types for agent i, then it *should* be the case that agent i's outcome is changed. This criterion captures a notion of legitimacy—agent i may view participation in the mechanism as involving an inherent tradeoff between information revelation and allocation. We can imagine a speech from agent i along the following lines: "The designer can learn that I have type θ_i and not θ_i' as long as the designer's knowledge of this makes a difference to my allocation."

Individual contextual privacy is closely related to *non-bossiness*, introduced by Satterthwaite and Sonnenschein (1981). A choice rule ϕ is *non-bossy* if for all $\theta_i \in \Theta$,

$$\phi_i(\theta_i, \boldsymbol{\theta}_{-i}) = \phi_i(\theta_i', \boldsymbol{\theta}_{-i}) \implies \phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}).$$

Non-bossiness says that if agent i changes her report from θ_i to θ_i' and her allocation is unchanged, then no other agent j's allocation changes either. The idea is that if agent i could unilaterally change her report and affect a change in some agent j's allocation without changing her own allocation, agent i would be "bossy."[15] We show that non-bossiness is a necessary condition for individual contextual privacy, and individual contextual privacy lies at the intersection of contextual privacy and non-bossiness.

PROPOSITION 11: *A choice rule $\phi : \boldsymbol{\Theta} \to X^n$ is individually contextually private if and only if it is contextually private and non-bossy.*

PROOF: Suppose for contradiction that protocol P is individually contextually private for ϕ, but ϕ is bossy. Since ϕ is bossy, there exists a $j \in N \setminus \{i\}$ and type profiles $(\theta_i, \boldsymbol{\theta}_{-i})$, $(\theta_i', \boldsymbol{\theta}_{-i})$ such that $\phi_i(\theta_i, \boldsymbol{\theta}_{-i}) = \phi_i(\theta_i', \boldsymbol{\theta}_{-i})$ but $\phi_j(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi_j(\theta_i', \boldsymbol{\theta}_{-i})$. Since $\phi_j(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi_j(\theta_i', \boldsymbol{\theta}_{-i})$, any protocol P for ϕ, including P, must have $(\theta_i, \boldsymbol{\theta}_{-i})$ and

[15] See Thompson (2014) for a discussion of the normative content of non-bossiness.

$(\theta'_i, \boldsymbol{\theta}_{-i})$ in distinct terminal nodes z, z' (otherwise P could not compute ϕ). But if $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta'_i, \boldsymbol{\theta}_{-i})$ belong to distinct terminal nodes in P and P is a contextually private protocol for ϕ, it must be that $\phi_i(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi_i(\theta'_i, \boldsymbol{\theta}_{-i})$, which contradicts the assumption that ϕ is bossy.

Next, assume that ϕ is non-bossy and contextually private. Consider $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta'_i, \boldsymbol{\theta}_{-i})$ in distinct terminal nodes of protocol P. By contextual privacy, $\phi(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi(\theta'_i, \boldsymbol{\theta}_{-i})$. By non-bossiness, $\phi_i(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi_i(\theta'_i, \boldsymbol{\theta}_{-i})$ follows. Thus, P is contextually private. *Q.E.D.*

This characterization results in unique characterizations for the first-price auction and the serial dictatorship.

PROPOSITION 12: *Serial dictatorships are the unique individually contextually private, efficient and strategyproof object assignment rules.*

PROOF: The serial dictatorship contextually private and non-bossy, hence indivudually contextually private by Proposition 11. It also is strategyproof.

Conversely, if ϕ is individually contextually private, then it is also non-bossy by Proposition 11. It is known that the only efficient, strategyproof and non-bossy object assignment mechanisms are serial dictatorships (see Hatfield (2009) for a proof, and compare Satterthwaite and Sonnenschein (1981)). *Q.E.D.*

PROPOSITION 13: *The first-price auction is the unique individually contextually private, efficient, and individually rational auction rule.*

PROOF: The first-price auction is individually contextually private. Indeed, the protocol outlined in Appendix C is individually contextually private. It is well known that the first-price auction is efficient and individually rational.

Let ϕ be any individually contextually private, efficient and individually rational auction rule. By Pycia and Raghavan (2022, Theorem 1), this means that it is (up to a zero set) a protocol for the first-price auction. *Q.E.D.*

7.2. *Group Contextual Privacy*

Another extension is *group* contextual privacy. Group contextual privacy requires that if two type *profiles* are distinguishable at the end of the protocol, then they must lead to different outcomes. This notion strengthens contextual privacy, which requires only that if a single agent's types are distinguishable, then they lead to different outcomes.

DEFINITION—Group-Contextual Privacy: A protocol $P = (V, E)$ for a social choice function ϕ is *contextually private* if for all type profiles $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \boldsymbol{\Theta}$ that are distinguished by P,

$$\phi(\boldsymbol{\theta}) \neq \phi(\boldsymbol{\theta}').$$

Notice again that this definition strengthens contextual privacy. Here, it is because it strengthens the underlying notion of distinguishability—two type profiles $\boldsymbol{\theta}, \boldsymbol{\theta}'$ are distinguishable if they belong to different terminal nodes. Regular contextual privacy's notion of distinguishability is on the agent-level—two types θ_i, θ_i' are distinguishable if they belong to different terminal nodes, *holding all other agent's types fixed* at $\boldsymbol{\Theta}_{-i}$.

We characterize the set of group contextually private protocols next. First, recall that we denote the set of outcomes reachable from node v, labelled with I_v by $\mathcal{X}_v := \phi(I_v)$.

THEOREM 4: *A protocol $P = (V, E)$ is group contextually private under a class of protocols $\mathcal{P}$ if and only if for any query, $\bigcup_{(v,w)\in E} \mathcal{X}_w = \mathcal{X}_v$ is a disjoint union.*

PROOF: First assume that P is group contextually private, and assume for contradiction that v is a query such that $(v, w), (v, w') \in E$ and $\mathcal{X}_w \cap \mathcal{X}_{w'} \neq 0$. Hence, there are $\boldsymbol{\theta} \in \boldsymbol{\Theta}_w$ and $\boldsymbol{\theta}' \in \boldsymbol{\Theta}_{w'}$ such that $\phi(w) = \phi(w')$, which contradicts group contextual privacy.

Next assume that reachable outcomes are disjoint at each query. Let $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ be distinguished at v. As outcomes are disjoint, it must be that $\phi(\boldsymbol{\theta}) \neq \phi(\boldsymbol{\theta}')$. Hence the protocol is group contextually private. *Q.E.D.*

This general characterization lends more practical insight when specialized to group contextual privacy under only sequential elicitation protocols.

COROLLARY 4: *A social choice function is group contextually private under sequential elicitation protocols if and only if it can be represented by a protocol in which, at every node, the agent's choice rules out a subset of the outcomes.*

This characterization, in particular, implies that the serial dictatorship is group contextually private. In a live sequential protocol of a serial dictatorship, whenever an agent is called to play, they obtain their favorite object among those that remain. So, their choice rules out the outcomes in which a different agent gets their favorite object that remains.

Group contextual privacy, under sequential elicitation protocols, is thus reminiscent of other extensive-form properties related to simplicity that the serial dictatorship satisfies. In particular, the serial dictatorship is obviously strategyproof Li (2017) and strongly obviously strategyproof Pycia and Troyan (2023).

Is there are containment relationship between protocols that are group contextually private and obviously strategyproof? It turns out that the answer is no: there are mechanisms that are group contextually private and not obviously strategyproof, and vice versa. The ascending auction is obviously strategyproof Li (2017), but not group contextually private with respect to sequential elicitation, as it is a protocol for the second-price auction, which is not contextually private. A class of "non-clinching rules", on the other hand, are strategyproof and group contextually private, but fail to have an obviously strategyproof implementation. Appendix D offers an example of a non-clinching rule, and shows that it is group contextually private but not obviously strategyproof.

8. RELATED LITERATURE.

This paper brings privacy considerations into extensive-form mechanism design.

Our restriction to sequential elicitation protocols coincides with the extensive-form messaging game used to define and study obvious strategyproofness (Li, 2017) and credibility (Akbarpour and Li, 2020). Credibility shares a motivation with contextual privacy—both criteria have to do with the potential for the designer to somehow abuse the information it receives. Credibility, which requires incentive compatibility for the auctioneer, sometimes coincides with the diagnoses of contextual privacy and sometimes not.[16] Under sequential elicitation protocols, contextual privacy can be understood as a form of privacy that is easier for participants to understand, just as obvious strategyproofness is a form of strategyproofness that is easier for participants to understand. Many papers study variants of obvious strategyproofness and their compatibility with other axioms and computational properties (Bade and Gonczarowski, 2016, Ashlagi and Gonczarowski, 2018, Mackenzie, 2020, Golowich and Li, 2021, Mackenzie, 2020, Pycia and Troyan, 2023).

Other considerations related to privacy and trust have been incorporated into mechanism design and market design, in both static and dynamic models. Though it is not the focus of their paper, Mackenzie and Zhou (2022) discuss how the dynamic *menu mechanisms* they define (in which at each history the agent chooses from a menu of possible outcomes) protect privacy compared to direct mechanisms. Grigoryan and Möller (2023) and Woodward (2020) define two different but related notions of the *auditability* of different mechanisms, based on the amount of information that would be required to determine whether the outcome of a mechanism had been correctly computed. Others study aftermarkets, and how the disclosure of past trades affects future trades (Dworczak, 2020, Ollar et al., 2021). Canetti et al. (2023) considers the privacy of the designer as opposed to our focus on the privacy of agents, and investigates the use of zero-knowledge proofs to prove properties of the mechanism without revealing the designer's objectives. Several papers have incorporated measures of "privacy loss" as constraints on mechanism design (Eilat et al., 2021, Liu and Bagh, 2020), where privacy loss is defined as some measure (e.g. Shannon entropy, Kullback-Leibler divergence) of information revelation. These measure-based criteria treat all datum as equal. Contextual privacy, unlike these measure-based criteria, is not about *how much* information is revealed, and is also not just about *whether* information is revealed, but rather it is about *how* the information that is revealed is *used*.[17]

There are two important precursors to contextual privacy that draw heavily on the mathematics of decentralized computation. The notion of *perfect implementation* (Izmalkov et al.,

[16] The descending or "Dutch" protocol of the first-price auction is both contextually private and credible, but the ascending protocol of the second-price auction is credible but not contextually private.

[17] For a survey of the literature on privacy in economics more broadly, i.e. beyond mechanism design, see Acquisti et al. (2016).

2005, 2011) seeks implementations that do not rely on trusted mediators, but rather rely on simple technologies that enable verification of what was learned—like sealed envelopes. Even more closely, contextual privacy (with sequential elicitation) exactly parallels the concept of *unconditional full privacy* for decentralized protocols (Brandt and Sandholm, 2005, 2008). Unconditional full privacy requires that the only information revealed through a decentralized protocol is the information contained in the outcome. In its standard formulation, it is not amenable to a mechanism design framework in which a principal chooses an allocation based on participants' information. Unconditional full privacy has been applied to an auction domain (Brandt and Sandholm, 2008), and a voting domain (Brandt and Sandholm, 2005). Our definition of contextual privacy brings unconditional full privacy into a framework amenable to economic analysis and extends it in several ways: we discuss assignment domains, we add count queries, and we discuss extensions to group and individual contextual privacy.[18] Furthermore, under general elicitation technologies, contextual privacy has no immediate analogue in decentralized computing.

Milgrom and Segal (2020)'s concept of *unconditional winner privacy* is similar to contextual privacy in that it brings unconditional full privacy into centralized mechanism design: unconditional winner privacy is unconditional full privacy in a centralized mechanism, for the winner only. Contextual privacy differs from unconditional winner privacy in three ways: (i) we require privacy for all players while Milgrom and Segal require privacy for just the winner, (ii) we define the set of outcomes to be allocations and prices (whereas Milgrom and Segal define outcomes to be allocations alone), (iii) we define contextual privacy in a range of domains while Milgrom and Segal consider only the auction domain.

Beyond unconditional full privacy, the most closely related concept in computer science and cryptography, lies an extensive literature on privacy preserving protocols for auctions and allocation. The literature on cryptographic protocols for auctions, going back to Nurmi and Salomaa (1993) and Franklin and Reiter (1996) is too vast to summarize here—the main point is that there are many cryptographic protocols that do not reveal *any* private information to a designer. Such protocols allow participants to jointly compute the outcome

[18] In addition, we strengthen the impossibility result Brandt and Sandholm (2008, Theorem 4.9) paralleling our Proposition 2 to cases where no ties are allowed (see the proof in subsection A.3).

without relying on any trusted third party. To this literature we bring analysis of the impact of the social and technological environments in which many designers operate: when arbitrary cryptographic protocols are not available, we need some other privacy desideratum to guide design. Thus, we align with the tradition of *contextual integrity* (Nissenbaum, 2004) which contrasts with traditions that view cryptography as a go-to solution for all privacy problems (Benthall et al., 2017).

An influential privacy desideratum is differential privacy (Dwork et al., 2006). Contextual privacy sharply diverges from interpretations of differential privacy in mechanism design contexts.[19] Differential privacy, as adapted for mechanism design contexts, says that the report of a single agent should have a negligible effect on the outcome. (This idea also has a precedent in the concept of "informational smallness" studied in Gul and Postlewaite (1992) and McLean and Postlewaite (2002).) To illustrate the sharp contrast between differential privacy and contextual privacy, suppose some bit of information is revealed through the mechanism. Differential privacy says that this bit can be revealed if it does not have an effect (or has a negligible effect) on the outcome. Contextual privacy says that this bit can be revealed if it *does* have an effect on the outcome—it can be revealed if the designer needed to know it. Whether contextual or differential privacy is a more appropriate notion of privacy will depend on context.[20]

BIBLIOGRAPHY

ACQUISTI, ALESSANDRO, CURTIS TAYLOR, AND LIAD WAGMAN (2016): "The economics of privacy," *Journal of economic Literature*, 54, 442–492. [43]

AKBARPOUR, MOHAMMAD AND SHENGWU LI (2020): "Credible auctions: A trilemma," *Econometrica*, 88, 425–467. [4, 25, 42]

ALAWADHI, NEHA (2021): "Only 1 in 250 understand role of encryption in securing messaging: Study," *Business Standard*, March 21. [3]

[19] Differential privacy was originally proposed as a tool for database management. For a survey of its incorporation into mechanism design, see Pai and Roth (2013).

[20] Differential privacy has also been microfounded with privacy concerns in agent utility functions, (Ghosh and Roth, 2015, Nissim et al., 2012, Roth and Schoenebeck, 2012, Ligett and Roth, 2012), and has been shown to be compatible with truthfulness (McSherry and Talwar, 2007, Xiao, 2013). In addition, information design for differential privacy (Schmutte and Yoder, 2022).

Alvarez, Ramiro and Mehrdad Nojoumian (2020): "Comprehensive survey on privacy-preserving protocols for sealed-bid auctions," *Computers & Security*, 88, 101502. [3]

Ashlagi, Itai and Yannai A Gonczarowski (2018): "Stable matching mechanisms are not obviously strategy-proof," *Journal of Economic Theory*, 177, 405–425. [43]

Ausubel, Lawrence M (2004): "An efficient ascending-bid auction for multiple objects," *American Economic Review*, 94, 1452–1475. [1]

Bade, Sophie and Yannai A Gonczarowski (2016): "Gibbard-Satterthwaite success stories and obvious strategyproofness," *arXiv preprint arXiv:1610.04873*. [43]

Balinski, Michel and Tayfun Sönmez (1999): "A tale of two mechanisms: student placement," *Journal of Economic theory*, 84, 73–94. [23]

Benthall, Sebastian, Seda Gürses, Helen Nissenbaum, et al. (2017): *Contextual integrity through the lens of computer science*, Now Publishers. [45]

Bogetoft, Peter, Dan Lund Christensen, Ivan Damgård, Martin Geisler, Thomas Jakobsen, Mikkel Krøigaard, Janus Dam Nielsen, Jesper Buus Nielsen, Kurt Nielsen, Jakob Pagter, et al. (2009): "Secure multiparty computation goes live," in *International Conference on Financial Cryptography and Data Security*, Springer, 325–343. [3]

Brandt, Felix and Tuomas Sandholm (2005): "Unconditional privacy in social choice," in *Theorectical Aspects of Rationality and Knowledge*, 207–218. [25, 44]

——— (2008): "On the existence of unconditionally privacy-preserving auction protocols," *ACM Transactions on Information and System Security (TISSEC)*, 11, 1–21. [24, 44, 51]

Canetti, Ran, Amos Fiat, and Yannai A Gonczarowski (2023): "Zero-Knowledge Mechanisms," *arXiv preprint arXiv:2302.05590*. [43]

Chor, Benny, Mihály Geréb-Graus, and Eyal Kushilevitz (1994): "On the structure of the privacy hierarchy," *Journal of Cryptology*, 7, 53–60. [18]

Chor, Benny and E. Kushilevitz (1989): "A Zero-One Law for Boolean Privacy," in *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, New York, NY, USA: Association for Computing Machinery, STOC '89, 62–72. [18, 20]

Dworczak, Piotr (2020): "Mechanism design with aftermarkets: Cutoff mechanisms," *Econometrica*, 88, 2629–2661. [43]

Dwork, Cynthia, Frank McSherry, Kobbi Nissim, and Adam Smith (2006): "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*, Springer, 265–284. [45]

Eilat, Ran, Kfir Eliaz, and Xiaosheng Mu (2021): "Bayesian privacy," *Theoretical Economics*, 16, 1557–1603. [43]

Franklin, Matthew K and Michael K Reiter (1996): "The design and implementation of a secure auction service," *IEEE Transactions on Software Engineering*, 22, 302–312. [44]

Ghosh, Arpita and Aaron Roth (2015): "Selling privacy at auction," *Games and Economic Behavior*, 91, 334–346. [45]

GOLOWICH, LOUIS AND SHENGWU LI (2021): "On the computational properties of obviously strategy-proof mechanisms," *arXiv preprint arXiv:2101.05149*. [43]

GRIGORYAN, ARAM AND MARKUS MÖLLER (2023): "A theory of auditability for allocation and social choice mechanisms," *arXiv preprint arXiv:2305.09314*. [43]

GUL, FARUK AND ANDREW POSTLEWAITE (1992): "Asymptotic efficiency in large exchange economies with asymmetric information," *Econometrica: Journal of the Econometric Society*, 1273–1292. [45]

HARSTAD, RONALD (2018): "Systems and methods for advanced auction management," *US Patent No. 10,726,476 B2*. [7, 30, 37]

HATFIELD, JOHN WILLIAM (2009): "Strategy-proof, efficient, and nonbossy quota allocations," *Social Choice and Welfare*, 33, 505–515. [40]

IZMALKOV, SERGEI, MATT LEPINSKI, AND SILVIO MICALI (2011): "Perfect implementation," *Games and Economic Behavior*, 71, 121–140. [44]

IZMALKOV, SERGEI, SILVIO MICALI, AND MATT LEPINSKI (2005): "Rational secure computation and ideal mechanism design," in *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, IEEE, 585–594. [43]

LI, SHENGWU (2017): "Obviously strategy-proof mechanisms," *American Economic Review*, 107, 3257–87. [4, 8, 15, 25, 37, 42]

LIGETT, KATRINA AND AARON ROTH (2012): "Take it or leave it: Running a survey when privacy comes at a cost," in *International workshop on internet and network economics*, Springer, 378–391. [45]

LIU, DE AND ADIB BAGH (2020): "Preserving bidder privacy in assignment auctions: design and measurement," *Management Science*, 66, 3162–3182. [43]

MACKENZIE, ANDREW (2020): "A revelation principle for obviously strategy-proof implementation," *Games and Economic Behavior*, 124, 512–533. [43]

MACKENZIE, ANDREW AND YU ZHOU (2022): "Menu mechanisms," *Journal of Economic Theory*, 204, 105511. [13, 43]

MCLEAN, RICHARD AND ANDREW POSTLEWAITE (2002): "Informational size and incentive compatibility," *Econometrica*, 70, 2421–2453. [45]

MCSHERRY, FRANK AND KUNAL TALWAR (2007): "Mechanism design via differential privacy," in *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, IEEE, 94–103. [45]

MILGROM, PAUL AND ILYA SEGAL (2020): "Clock auctions and radio spectrum reallocation," *Journal of Political Economy*, 128, 1–31. [44]

MOULIN, H. (1980): "On Strategy-Proofness and Single Peakedness," *Public Choice*, 35, 437–455. [27]

NISSENBAUM, HELEN (2004): "Privacy as contextual integrity," *Washington Law Review*, 79, 119. [45]

NISSIM, KOBBI, CLAUDIO ORLANDI, AND RANN SMORODINSKY (2012): "Privacy-aware mechanism design," in *Proceedings of the 13th ACM Conference on Electronic Commerce*, 774–789. [45]

NURMI, HANNU AND ARTO SALOMAA (1993): "Cryptographic protocols for Vickrey auctions," *Group Decision and Negotiation*, 2, 363–373. [44]

OLLAR, MARIANN, MARZENA J ROSTEK, AND JI HEE YOON (2021): "Privacy in markets," *Available at SSRN 3071374*. [43]

PAI, MALLESH M AND AARON ROTH (2013): "Privacy and mechanism design," *ACM SIGecom Exchanges*, 12, 8–29. [45]

PYCIA, MAREK AND MADHAV RAGHAVAN (2022): "Non-bossiness and First-Price Auctions," Tech. rep., CEPR Discussion Paper No. DP16874. [8, 40]

PYCIA, MAREK AND PETER TROYAN (2023): "A theory of simplicity in games and mechanism design," *Econometrica*, 91, 1495–1526. [8, 42, 43]

ROTH, AARON AND GRANT SCHOENEBECK (2012): "Conducting truthful surveys, cheaply," in *Proceedings of the 13th ACM Conference on Electronic Commerce*, 826–843. [45]

ROTHKOPF, MICHAEL H, THOMAS J TEISBERG, AND EDWARD P KAHN (1990): "Why are Vickrey auctions rare?" *Journal of Political Economy*, 98, 94–109. [1]

SATTERTHWAITE, MARK A AND HUGO SONNENSCHEIN (1981): "Strategy-proof allocation mechanisms at differentiable points," *The Review of Economic Studies*, 48, 587–597. [8, 39, 40]

SCHMUTTE, IAN M AND NATHAN YODER (2022): "Information Design for Differential Privacy," in *Proceedings of the 23rd ACM Conference on Economics and Computation*, 1142–1143. [45]

SEGAL, ILYA (2007): "The communication requirements of social choice rules and supporting budget sets," *Journal of Economic Theory*, 136, 341–378. [13]

SHAPLEY, LLOYD AND HERBERT SCARF (1974): "On cores and indivisibility," *Journal of mathematical economics*, 1, 23–37. [21]

SÖNMEZ, TAYFUN AND M UTKU ÜNVER (2010): "Course bidding at business schools," *International Economic Review*, 51, 99–123. [23]

THOMPSON, WILLIAM (2014): "Non-bossiness," *Rochester Center for Economic Research, Working Paper Number 586*. [39]

VICKREY, WILLIAM (1961): "Counterspeculation, auctions, and competitive sealed tenders," *The Journal of finance*, 16, 8–37. [1, 25]

WILSON, ROBERT (1989): "Game Theoretic Analysis of Trading," in *Advances in Economic Theory: Fifth World Congress*, CUP Archive, 12, 33. [25]

WOODWARD, KYLE (2020): "Self-Auditable Auctions," Tech. rep., Working paper. [43]

XIAO, DAVID (2013): "Is privacy compatible with truthfulness?" in *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, 67–86. [45]

APPENDIX A: PROOFS

A.1. *Proof of Corollary 2.*

PROOF OF COROLLARY 2: We first consider necessity. Assume for contradiction that there is a contextually private protocol P for the choice function ϕ and that there is a product set $\mathbf{\Theta}'$ such that all types are inseparable under $\mathbf{\Theta}'$ and ϕ is non-constant on this set.

As ϕ is nonconstant on $\mathbf{\Theta}'$, the protocol must make a query separating type profiles $(\theta, \boldsymbol{\theta}_{-i})$ and $(\theta', \boldsymbol{\theta}_{-i})$ for $\theta \not\sim_{i,\phi,\mathbf{\Theta}'} \theta'$ for some agent i. Consider the earliest such query in the precedence order on P.

By the choice of v and $\theta \sim_{i,\theta,\tilde{\Theta}} \theta'$, there must be a chain $\theta^1, \theta^2, \ldots, \theta^k$ such that $\theta^1 = \theta$ and $\theta^k = \theta'$ and

$$\theta^1 \sim'_{i,\phi,\mathbf{\Theta}'} \theta^2 \sim'_{i,\phi,\mathbf{\Theta}'} \cdots \sim'_{i,\phi,\mathbf{\Theta}'} \theta^k.$$

That is, there is a chain of direct inseparability from θ to θ'. As θ and θ' are separated at v, there must be $l = 1, 2, \ldots, k-1$ such that θ^l is separated from θ^{l+1} at v. As a property of sequential elicitation protocols, for any $\boldsymbol{\theta}_{-i}$ such that $(\theta^l, \boldsymbol{\theta}_{-i}), (\theta^{l+1}, \boldsymbol{\theta}_{-i}) \in \mathbf{\Theta}' \subseteq \mathbf{\Theta}_v$, $(\theta^{l+1}, \boldsymbol{\theta}_{-i})$ and $(\theta^{l+1}, \boldsymbol{\theta}_{-i})$ lead to distinct terminal nodes. By direct inseparability, there is $\boldsymbol{\theta}_{-i}$ such that $(\theta^l, \boldsymbol{\theta}_{-i}), (\theta^{l+1}, \boldsymbol{\theta}_{-i}) \in \mathbf{\Theta}'$ and $\phi(\theta^l, \boldsymbol{\theta}_{-i}) = \phi(\theta^{l+1}, \boldsymbol{\theta}_{-i})$. Together, these two observations yield a contradiction to contextual privacy of P.

Now consider sufficiency. We define a contextually private protocol inductively. Throughout the induction, the following holds:

$$\begin{gathered}\text{For any terminal nodes } w, w' \text{ whose earliest point of departure in } P \text{ is } v \text{, there} \\ \text{are no } (\theta_i, \boldsymbol{\theta}_{-i}) \in \Theta_{v'} \text{ and } (\theta'_i, \boldsymbol{\theta}_{-i}) \in \Theta_{v''} \text{ such that } \phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta'_i, \boldsymbol{\theta}_{-i}).\end{gathered} \tag{7}$$

Note that a protocol that satisfies (7) at all internal nodes is contextually private. We prove the statement by induction over the tree P.

Assume a protocol has been constructed until query v associated to type set $\mathbf{\Theta}_v \subseteq \mathbf{\Theta}$. If ϕ is constant on the remaining set, the node is terminal, and the outcome of the social choice function can be determined.

Otherwise, because there is no restriction to a product set $\mathbf{\Theta}'$ under which all types are inseparable, there are types θ, θ' that are separable under $\mathbf{\Theta}_v$ for agent i. Consider the binary query that separates the equivalence class of θ, $[\theta]_{i,\phi,\mathbf{\Theta}_v}$, from its complement $\Theta_{v,i} \setminus [\theta]_{i,\phi,\mathbf{\Theta}_v}$, which is non-empty as it contains at least θ'. By definition of $\sim$, for any $\boldsymbol{\theta}_{-i}$,

$$\phi(\tilde{\theta}_i, \boldsymbol{\theta}_{-i}) \neq \phi(\tilde{\theta}'_i, \boldsymbol{\theta}_{-i}), \tag{8}$$

implying that (7) continues to hold. By induction, it holds at all internal nodes. *Q.E.D.*

A.2. *Proof of Theorem 2*

PROOF OF THEOREM 2: The proof uses the Corners Lemma. We first choose the type profile for $n-2$ agents that are not labelled i or j. Then we construct a square representing possible types for agents i and j.

Let $s_1 > s_2 > s_3 > s_4$. Fix the type profile of all $n-2$ agents that are not i or j to be $\boldsymbol{\theta}_{-ij}$ where each agent has a score greater than s_1 for their top choice object, and their top choice object has capacity to accommodate them. Denote $\mathcal{C}$ there is no oversupply, i.e. $\sum_{c \in \mathcal{C}} \kappa(c) = n$, the number of remaining spots is $n - (n-2) = 2$. Assume without loss of generality that the remaining spots are for different objects. Label these objects with remaining spots a and b.

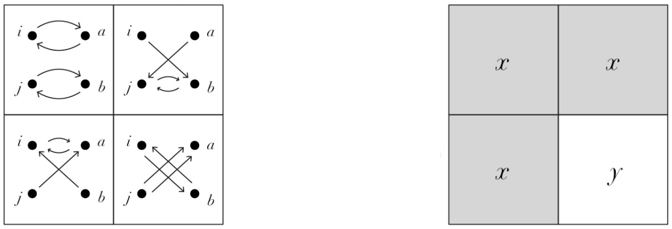
Consider the final two agents $i, j \in \mathcal{N}$. Two possible (partial) types for agent i, j are:

$$\theta_i = (a \succ_i b, s_i(a) = s_1, s_j(b) = s_4), \qquad \theta'_i = (b \succ_i a, s_i(a) = s_3, s_j(b) = s_2)$$
$$\theta_j = (b \succ_j a, s_j(a) = s_4, s_j(b) = s_1), \qquad \theta'_j = (a \succ_j b, s_j(a) = s_2, s_j(b) = s_3).$$

The parts of agent i and j's types for other schools can be arbitrary.

Let x be the outcome in which agent i is matched to school a and j is matched to b. Let y be the outcome in which agent i is matched to b and j is matched to a. In both x and y, all agents not i or j are assigned to their top choice object at which they have a high score.

Stability requires that $\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-i,-j}) = (\theta_i, \theta'_j, \boldsymbol{\theta}_{-i,-j}) = (\theta'_i, \theta_j, \theta_{-i,-j}) = x$ while $\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}) = y$. We have chosen a particular $\boldsymbol{\theta}_{-ij} \in \Theta^{n-2}$, and particular $\theta_i, \theta'_i, \theta_j, \theta'_j \in$



x	x
x	y

FIGURE A.1.—Applying the Corners Lemma to college assignment. Agent types $\theta_i, \theta_i', \theta_j, \theta_j'$ (left, arrows from agents denote favored object, arrows from objects denote high score); outcomes under any stable choice rule (right, where $x = ((i,a),(j,b))$ and $y = ((j,a),(i,b))$).

Θ such that the condition (4) does not hold for any stable choice rule. So, by the Corners Lemma, no stable choice rule is contextually private under sequential elicitation. *Q.E.D.*

A.3. *Alternate Proof of Proposition 2.*

We include the following alternative proof of Proposition 2 which does not rely on ties. While Brandt and Sandholm (2008) includes a proof analogous to the one in the main text using the Corners Lemma, they do not present a more general proof as the one below. Note that in order to guarantee that there are no ties, we assume that the type space has at least 9 values. As auctions are often studied in continuous type spaces, this is restriction is not too consequential.

PROPOSITION: Assume $n \geq 3$ agents and $|\theta| \geq 3$. Under sequential elicitation, the second-price choice rule ϕ^{SP} is not contextually private. If $n \geq k+2$, the uniform k-th price auction is not contextually private.

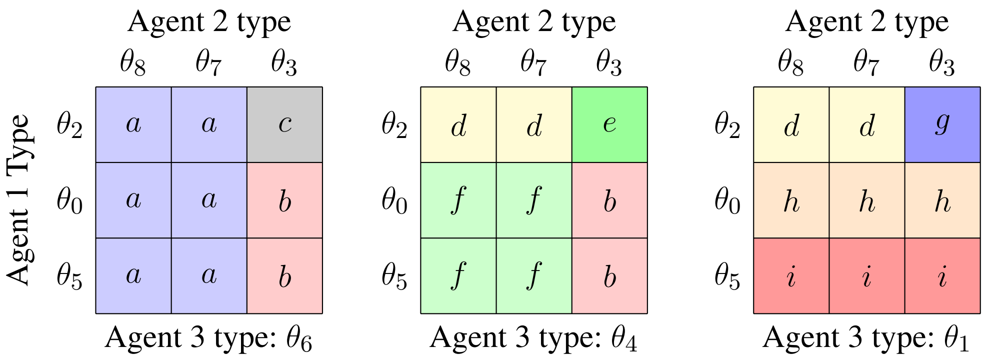
ALTERNATE PROOF OF PROPOSITION 2 (NO TIES): This proof proceeds in two steps. First, we construct a direct contradiction of Corollary 2 in a case with $n = 3$ and $|\Theta| = 9$. Then we argue that for any auction with $n \geq 3$ and $|\Theta| \geq 9$, this counterexample cannot be ruled out.

Construction of a minimal counterexample. Let $n = 3$ and let

$$\Theta = \{\theta_0, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8\}.$$

Consider the product set $\boldsymbol{\Theta}' = \{\theta_5, \theta_0, \theta_2\} \times \{8, 7, 3\} \times \{6, 4, 1\}$. In this product set, the first factor represents types of agent 1, the second represents possible types of agent 2, and the third represents possible types of agent 3. We will show that when ϕ^{SPA} is evaluated on this restricted product set, it is non-constant and all types in the product set are inseparable.

To see this, we construct the tensor of outcomes for the product set. This tensor is represented in Figure A.2. We represent agent 1's type on the up-down axis, agent 2's type on the left-right axis, and agent 3's type is constant for each box. The outcomes under ϕ^{SPA} are represented by letters and colors. For example, the upper left corner in the left-most box signifies $\phi^{\text{SPA}}(\theta_2, \theta_8, \theta_6) = a$, where a is the outcome under which agent 2 wins the object and pays a price θ_6.



Agent 1 Type

Agent 2 type

	θ_8	θ_7	θ_3
θ_2	a	a	c
θ_0	a	a	b
θ_5	a	a	b

Agent 3 type: θ_6

Agent 2 type

	θ_8	θ_7	θ_3
θ_2	d	d	e
θ_0	f	f	b
θ_5	f	f	b

Agent 3 type: θ_4

Agent 2 type

	θ_8	θ_7	θ_3
θ_2	d	d	g
θ_0	h	h	h
θ_5	i	i	i

Agent 3 type: θ_1

FIGURE A.2.—Counterexample ϕ^{SPA} with $n = 3$ and $|\Theta| = 9$.

To see that this constitutes a violation of contextual privacy, we show that: (i) ϕ is non-constant on $\boldsymbol{\Theta}'$, and (ii) for all agents i, and all $\theta_i, \theta_i' \in \boldsymbol{\Theta}'$, θ_i and θ_i' are inseparable. As for (i), we can observe immediately that $\phi|_{\boldsymbol{\Theta}'}$ is non-constant. To see (ii) that all types are inseparable, we go through each agent in turn.

- Agent 1: Outcome a is the same for $\boldsymbol{\theta}_{-1} = (\theta_7, \theta_6)$, hence all agent 1 types are inseparable.
- Agent 2: Outcome i for $\boldsymbol{\theta}_{-2} = (\theta_5, \theta_1)$ show that all agent 2 types are inseparable.
- Agent 3: θ_6 and θ_4 are inseparable because they both yield outcome b for $\boldsymbol{\theta}_{-3} = (\theta_0, \theta_3)$. θ_1 and θ_4 are inseparable because they both yield outcome d for $\boldsymbol{\theta}_{-3} = (\theta_2, \theta_7)$.

Now that we have constructed a counter-example, we argue that for any settings with $n \geq 3$ and $|\Theta| \geq 9$, this situation cannot be ruled out. Consider a restriction $\boldsymbol{\Theta}'' = \boldsymbol{\Theta}' \times \bigtimes_{i\in\{4,\dots,n\}} \Theta_i$ where each Θ_i for agents $i \in \{4, \dots, n\}$ contains only types below θ_9 and $\boldsymbol{\Theta}'$ is as defined in step 1. Then, $\phi|_{\boldsymbol{\Theta}''} = \phi|_{\boldsymbol{\Theta}'}$. As shown in step 1, $\phi|_{\boldsymbol{\Theta}'}$ is non-constant and all types are inseparable.

Q.E.D.

A.4. *Proof of Proposition 3*

PROPOSITION: Assume there are $n > 3$ agents. There is no efficient, uniform-price contextually private double auction price rule under sequential elicitation protocols.

PROOF OF PROPOSITION 3. Since every efficient uniform-price choice rule is a Walrasian choice rule, it suffices to show that no Walrasian choice rule is contextually private.

We use the Corners Lemma. Consider four agents i, j, k, and ℓ who have the types

$$\{\theta_{[m-1]}, \theta_{[m]}, \theta_{[m+1]}, \theta_{[m+2]}\}.$$

For ease of exposition (and without loss of generality), suppose these middle four types are belong to the set $\{\underline{\theta}, \bar{\theta}\}^4$ with $\underline{\theta} < \bar{\theta}$. Consider arbitrary endowments.

We construct two squares: a square that holds agents k and ℓ's types fixed at $(\theta_k, \theta_\ell) = (\underline{\theta}, \underline{\theta})$ and considers all possible combinations in $\{\underline{\theta}, \bar{\theta}\}^2$ for agents i and j; a square that holds i and j's types fixed at $(\theta_k, \theta_\ell) = (\bar{\theta}, \bar{\theta})$ and varies k and ℓ in the same manner. See Figure A.3.

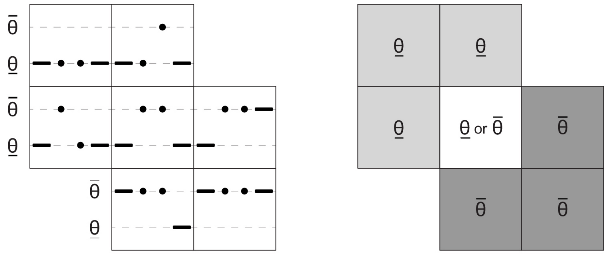


FIGURE A.3.—Applying the Corners Lemma to the double auction. Combinations of types for agents i, j, k, ℓ (left); required prices t in an efficient choice rule.

Let x be the outcome in which the market clearing price is $t = \underline{\theta}$ and let x' be the outcome in which the market clearing price is $t = \bar{\theta}$. Consider first the top square which holds the types of agents k and ℓ fixed and varies the types of agents i and j. Efficiency requires $\phi(\theta_i, \theta_j, \underline{\theta}, \underline{\theta}) = \phi(\theta'_i, \theta_j, \underline{\theta}, \underline{\theta}) = \phi(\theta_i, \theta'_j, \underline{\theta}, \underline{\theta}) = x$. Efficiency also requires that $\phi(\theta'_i, \theta'_j, \underline{\theta}, \underline{\theta}) \in \{x, x'\}$.

Now consider the bottom square which holds the types of agents i and j fixed and varies the types of agents k and ℓ. Efficiency requires $\phi(\bar{\theta}, \bar{\theta}, \theta_k, \theta_\ell) \in \{x, x'\}$. It also requires that $\phi(\bar{\theta}, \bar{\theta}, \theta'_k, \theta_\ell) = \phi(\bar{\theta}, \bar{\theta}, \theta_k, \theta'_\ell) = \phi(\bar{\theta}, \bar{\theta}, \theta'_k, \theta'_\ell) = x'$.

The outcome under the type profile in the box that conjoins the two squares $(\bar{\theta}, \bar{\theta}, \underline{\theta}, \underline{\theta})$ must be *either* x or x' (it cannot be both). If it is x, then the Corners Lemma is violated in the bottom $(k - \ell)$ square. If it is x', then the Corners Lemma is violated in the top $(i - j)$ square. So, there must be a violation of the Corners Lemma, and any efficient uniform-price rule is not contextually private under sequential elicitation protocols. *Q.E.D.*

A.5. *Proof of Proposition 4*

PROOF: The proof is related to the one forProposition 3, however for a median with an odd number of bids. Consider two adjacent types $\theta, \theta' \in \Theta$ and $\boldsymbol{\theta}_{-ij} \in \boldsymbol{\Theta}_{-ij}$ such that

$$\theta = \text{median}(\boldsymbol{\theta}_{-ij}, k_1, k_2, \ldots, k_{n-1}).$$

By assumption that rules out extreme phantom ballots such types and a partial type profile exists. The set $\tilde{\Theta} = \{theta, \theta'\} \times \{\theta, \theta'\} \times \boldsymbol{\Theta}_{-ij}$ produces a corner. By the Corners Lemma, generalized median voting rules are not contextually private. *Q.E.D.*

A.6. *Proof of Theorem 3*

THEOREM: Let P be a protocol for ϕ, where ϕ exhibits interval pivotality. There is a bimonotonic protocol P' that is exactly as contextually private as P.

PROOF: The definition of bimonotonic protocols only depends on a single agent, hence we consider an arbitrary but fixed agent $i \in N$.

We will show that there are ways to modify to protocols for interval pivotal social choice functions that produces a protocol that is as contextually private as the original. We first discuss *injecting* then *filling-in* and then.... [].

The first operation we define is an *injection* to a protocol.

DEFINITION—v-before-v' injected protocol: Let $P = (V, E)$ be a protocol. Let (v, s_v), $s_v\colon \mathbf{\Theta} \to \mathrm{children}(v)$ be a query and let

- $v' \in V$ be a non-root node
- $u = \mathrm{parent}(v)$
- $\mathrm{subtree}(v)$ be the sub-tree following v.

Define the v-before-v'-injected protocol $P_{v,v'}$ as the protocol where

- $\mathrm{child}(u) = v'$,
- all children of v' are followed by $\mathrm{subtree}(v)$.

We denote $i(v)$ as the agent who receives a query at node v.

DEFINITION: Let $(\Theta_i, \leq)$ be a finite ordered type space and $\mathrm{succ}(\theta)$ resp. $\mathrm{pred}(\theta)$ be the next-highest resp. lowest type, if existent. Also, let s be a query for node v. We call $\overline{\theta}_{v,i}, \underline{\theta}_{v,i}$ the highest resp. lowest *separator* if

$$\overline{\theta}_v = \min\{\theta_i \in \Theta_{v,i} \mid \exists \boldsymbol{\theta}_{-i} : s(\mathrm{succ}(\theta_i), \boldsymbol{\theta}_{-i}) \neq s(\theta_i, \boldsymbol{\theta}_{-i})\}$$

$$\overline{\theta}_v = \max\{\theta_i \in \Theta_{v,i} \mid \exists \boldsymbol{\theta}_{-i} : s(\mathrm{pred}(\theta_i), \boldsymbol{\theta}_{-i}) \neq s(\theta_i, \boldsymbol{\theta}_{-i})\}$$

(If these sets are empty, we will set $\overline{\theta}_v < \theta$ for all $\theta \in \Theta$ and $\underline{\theta}_v > \theta$ for all $\theta \in \Theta$.) That is, the lowest separator at node v is the lowest element in the type space that ends up in a different child.

PROPOSITION 14—Protocol Injection: *Let ϕ be interval pivotal, $i(v) = i(v')$, and $v' \in \mathrm{subtree}(v)$. Define*

$$\overline{\theta}_{v,v'} = \max\{\overline{\theta}_v, \overline{\theta}_{v'}\} \tag{9}$$

$$\underline{\theta}_{v,v'} = \min\{\underline{\theta}_v, \underline{\theta}_{v'}\} \tag{10}$$

where $\overline{\theta}_v, \underline{\theta}_v$ are the highest and lowest separators at v as defined above (and similarly for v'). Then, for any $\tilde{\theta}_i \in [\underline{\theta}_{v,v'}, \overline{\theta}_{v,v'}]$, a threshold query *$v''$ with $s_{v''}(\boldsymbol{\theta}) = \mathbf{1}_{\{\boldsymbol{\theta} \in \boldsymbol{\Theta} | \theta_i \leq \tilde{\theta}_i\}}$ satisfies*

$$P \sim_{\phi} P_{v'',v'}$$

for any $\boldsymbol{\theta} \in \boldsymbol{\Theta}$.

In other words, consider an agent that is asked two queries v and v' one after the other on one path through the protocol. If one inserts a threshold query *in between* v and v', where the threshold $\tilde{\theta}_i$ lies *between* the highest and lowest types that v and v' can distinguish, this insertion will result in no change to the set of contextual privacy violations.

PROOF: Consider the newly introduced node v'' as introduced in the statement. Assume $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ produce a contextual privacy violation for agent i at node v'',[21] i.e. they are distinguished at v'' while

$$\phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}).$$

Assume without loss that $\theta_i \leq \tilde{\theta}_i < \theta_i'$, where $\tilde{\theta}_i$ is the threshold of the threshold query asked at v''.

We will now show that $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ are contextual privacy violations at v or v' in P. By interval pivotality, there are two cases:

(a) $\phi(\hat{\theta}_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i})$ for all types $\hat{\theta}_i \leq \theta_i'$, or

(b) $\phi(\hat{\theta}_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i, \boldsymbol{\theta}_{-i})$ for all types $\hat{\theta}_i > \theta_i$.

For the first case (a), assume without loss that v attains the minimum in (9). By definition of $\underline{\theta}_{v,v'}$, we have that there must be two (adjacent) types $\underline{\theta}_v, \mathrm{succ}(\underline{\theta}_v) \leq \theta_i'$ that are distinguished at v. There are two cases within case (a), assuming without loss that v attains the minimum in (9):

[21] Note that if a contextual privacy violation is produced at a non-terminal node, then, by the definition of protocols (particularly that the type spaces associated to the children of a node v form a partition of the type space associated with the parent) the same contextual privacy violation will persist at the terminal node.

- $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta'_i, \boldsymbol{\theta}_{-i})$ are distinguished at v. In this case, both type profiles produce contextual privacy violations for agent i at v.
- $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta'_i, \boldsymbol{\theta}_{-i})$ are *not* distinguished at v, but this means that $\{\theta_i, \theta'_i\}$ are distinguished from $\underline{\theta}_v$ or $\mathrm{succ}(\underline{\theta}_v)$ (or both). In this case as well, these type profiles are contextual privacy violations.

For the second case (b), we can follow similar reasoning, but with flipped inequiuality signs. In this case, one proceeds by showing that a contextual privacy violation happened at the query maximizing (10).

Hence, in both cases (a) and (b), we showed that inserting a threshold query between v and v' does not add new contextual privacy violations, i.e. if there is a contextual privacy violation in $P_{v'',v'}$ then that contextual privacy violation also occurs in P. Formally,

$$P \succeq_\phi P_{v'',v'}. \tag{11}$$

Observe also that any type profiles that are separated by P are also separated by $P_{v'',v'}$, which means that

$$P \preceq_\phi P_{v'',v'}$$

and hence $P \sim_\phi P_{v'',v'}$. *Q.E.D.*

We next introduce another operation that does not affect the set of CP violations. A filled-in protocol is one in which, for each agent, the threshold of every threshold query is adjacent in the type space to the threshold of the previously asked threshold query.

DEFINITION: We call a protocol P *filled-in* if (a) all queries are threshold queries i.e. for all $v \in V(P)$, $s_v(\boldsymbol{\theta}) = \mathbf{1}_{\{\boldsymbol{\theta} \in \boldsymbol{\Theta} | \theta_i \leq \tilde{\theta}\}}$ and (b) for any v, v' such that $i(v) = i(v')$ and there is no v'' such that $i(v'') = i(v)$ and $v \prec_P v'' \prec_P v'$, it must hold that

$$\mathrm{thresh}(v) = \mathrm{succ}_{\Theta_i} \mathrm{thresh}(v') \text{ or } \mathrm{thresh}(v) = \mathrm{pred}_{\Theta_i} \mathrm{thresh}(v').$$

That is, in a filled-in protocol, every query is a threshold query and the threshold for every query adjacent in the protocol is also adjacent in the type space.

LEMMA 1: *Let ϕ be an interval pivotal choice rule. Then, for any protocol P, there is a filled-in protocol P' such that*

$$P \sim_\phi P'$$

PROOF: We prove this lemma in three steps.

Anchoring. We first add trivial queries $s_i(\boldsymbol{\theta}) = v_i$ to all agents in the beginning, where v_i is a new child to the queries. These allow us to perform protocol injection on initial queries to agents.

Inserting trivial queries affects neither measurability nor contextual privacy violations. We then introduce threshold queries at the highest ($\overline{\theta}_{v,v'}$) and lowest ($\underline{\theta}_{v,v'}$) separators of pairs of queries to the same agent, that is, before the later in P of v, v', $i(v) = i(v')$. This leads to the introduction of at most $|V|^2$ many new queries and does not affect measurability or contextual privacy violations.

Inserting. Let v, v' be the set of queries such that $i(v) = i(v')$ and there is $\tilde{\theta}$ such that $\mathrm{thresh}(v) \prec_\Theta \tilde{\theta}\, \mathrm{pred}_\Theta\, \mathrm{thresh}_{v'}$ but there is no v'' such that $v \prec_P v'' \prec_P v'$ and and $\tilde{\theta} = \tilde{\theta}_v$. As threshold queries v'' continue to be inserted before v' with threshold $\tilde{\theta}$, the set of candidate insertions (triples (v, v', v'')) shrinks. As there are only $|V|^3$ many candidate insertions, this process terminates after finitely many rounds.

Deleting. After this process, for any non-threshold queries $v \in V(P)$, all thresholds between $\underline{\theta}_v$ and $\overline{\theta}_v$ are queried. This implies that non-threshold queries can be removed without affecting measurability. Similarly to the observation that insert operations do not affect contextual privacy violations. The resulting protocol is filled-in. *Q.E.D.*

The following observation helps finish the proof.

DEFINITION—Redundant query: Let P be a protocol. A query w is *redundant* if there is a query v such that

$$s_v = s_w.$$

We call w a *repetition of* v. If there is a query that w repeats, then we call it *repeated.*

DEFINITION—Protocol locally deduplicated at v: Let P be a protocol. Let $v \in V(P)$ be a repeated query. Then, there is a unique $w \in \text{children}(v)$ such that $\mathbf{\Theta}_w \neq \emptyset$. Denote by $\text{Dedup}_v(P)$ the protocol that attaches $\text{subtree}(w)$ to $\text{parent}(v)$ and deletes $\text{subtree}(w')$ for all $w' \in \text{children}(v) \setminus \{w\}$.

DEFINITION: We inductively define a deduplication through a nested induction. Starting with the earliest query u that has a repetition, we start with the earliest repetition v, and apply $\text{Dedup}_v(P)$. Continuing with the next-earliest repetition v', we apply $\text{Dedup}'_v(P)$. After all repetitions of u are treated in this way, we continue with the next node u' that has a repetition, and deduplicate in the same way. Each of these operations decreases $|V(P)|$ by at least one, and, as $V(P)$ initially is finite, will terminate after finitely many steps.

LEMMA 2: *For any filled-in protocol P for any social choice function ϕ the deduplicated protocol is bimonotonic.*

PROOF: Assume a deduplicated, filled-in protocol were not bimonotonic. In this case, there must be a path and three threshold queries v, v', v'' such that $\text{thresh}(v) = \text{thresh}(v'')$ and $\text{thresh}(v') \neq \text{thresh}(v), \text{thresh}(v')$. This, however, would mean that the the protocol is not fully deduplicated. *Q.E.D.*

Q.E.D.

A.7. *Proof of Proposition 9.*

PROPOSITION: The over-descending protocol for ϕ^{SP} has a dominant strategy equilibrium and is maximally contextually private for ϕ^{SP}.

A.8. *Proof of Proposition 10.*

PROPOSITION: The ascending and ascending-join protocols for ϕ^{SP} have an equilibrium in obviously dominant strategies and are maximally contextually private for ϕ^{SP}.

APPENDIX B: SERIAL DICTATORSHIPS ARE CONTEXTUALLY PRIVATE UNDER SEQUENTIAL ELICITATION

In the assignment domain, we fix a set $\mathcal{C}$ of objects. The set of outcomes is $\mathcal{X} = 2^{N \times \mathcal{C}}$.

In the standard object assignment setting, agents may receive at most one object, and agents have ordinal preferences over objects, which are private information. So agents' types $\boldsymbol{\theta} \in \Theta$ are preference orders of $\mathcal{C}$ where $\succ_i$ reference to agent i's preference ordering. A choice rule ϕ is *efficient* if there is no outcome x such that $x \succsim_i \phi_i(\boldsymbol{\theta})$ for all agents i and $x \succ_j \phi_j(\boldsymbol{\theta})$ for some agent j.

Let $A \subseteq N \times \mathcal{C}$ be an outcome. A *partial assignment* $N(A)$ is the set of agents who have an assigned object in A, i.e. $N(A) = \{i \in N \colon \exists c \in \mathcal{C} : (i, c) \in A\} \subseteq N$. If $N(A) = N$, we call A *complete*. For a partial assignment A, denote $A(i)$ the (at most one) object assigned to agent i.

The *remaining objects* $R(A)$ are the objects that do not have an assigned agent in A, i.e. $R(A) \coloneqq \{c \in \mathcal{C} \colon \nexists i \in N : (i, c) \in A\}$.

We first study serial dictatorship mechanisms, in which agents are sequentially asked to choose one of the remaining objects. To define the serial dictatorship protocol in our notation, we characterize the nodes and edges of the rooted tree. Fix the permutation $\pi \colon N \to N$ of agents that defines the priority order of the serial dictatorship. The *serial dictatorship protocol* with respect to π has as nodes all partial assignments to agents in $N_i^\pi \coloneqq \{\pi(i') \colon 1 \leq i' \leq i\}$ for any $i \in N$. Edges are between partial assignments A, A' such that exactly agents N_i^π resp. N_{i+1}^π are assigned an object, $N(A_i) = N_i^\pi$ and $N(A_{i+1}) \in N_{i+1}^\pi$, and $\pi(1)$, $\pi(2)$, …, $\pi(i)$ are assigned the same objects. We define sets of type profiles associated to each node recursively. For an edge (A, A'),

$$\Theta^{A'} = \Theta^A \cap \left\{ \boldsymbol{\theta} \in \boldsymbol{\Theta} : \max_{\theta_{\pi(i)}} R(A) = A'(\pi(i+1)) \right\}.$$

Here, $R(i)$ is the set of remaining objects when it is agent i's turn in the partial order; $\max_{\theta_{\pi(i)}} R(i)$ is the most preferred element of $R(i)$ with respect to the strict order $\theta_{\pi(i)}$. If a node is reached that is a complete assignment, the protocol ends, and the complete assignment is computed.

PROPOSITION 15: *Serial dictatorships are contextually private under sequential elicitation.*

PROOF OF PROPOSITION 15: Consider $\theta_i, \theta_i' \in \Theta$ and a partial type profile for other agents $\boldsymbol{\theta}_{-i} \in \Theta^{n-1}$ such that $(\theta_i, \boldsymbol{\theta}_{-i})$ is separated from $(\theta_i', \boldsymbol{\theta}_{-i})$. We will show that $\phi(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi(\theta_i', \boldsymbol{\theta}_{-i})$.

Denote A the node of separation. By definition of sequential elicitation, this must be a query to agent i. By definition of serial dictatorship, the children of A are given by $\{\theta | \theta_{\pi(i)} \in \max_{\theta_{\pi(i)}} R(i) = c\}$ for some $c \in R(A)$. Hence, if $\phi(\theta_i, \boldsymbol{\theta}_{-i})$ and $\phi(\theta_i, \boldsymbol{\theta}_{-i})$ are separated from each other, agent i must get a different assignment under θ_i and θ_i', hence $\phi(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi(\theta_i, \boldsymbol{\theta}_{-i})$.

Q.E.D.

The above protocol for the serial dictatorship satisfies even stronger versions of contextual privacy. For any θ, θ' in distinct terminal nodes of the protocol, $\phi(\theta) = \phi(\theta')$. The reason for this is that at an earliest point of departure, the assignment to an agent is different, and any actions by later agents will lead to different outcomes. Such *group-contextually private* mechanism may be formulated as restricting the set of outcomes. Additionally, if θ_i, θ_i' are such that for some $\boldsymbol{\theta}_{-i}$, $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ are in distinct terminal nodes, it holds that $\phi_i(\theta_i, \boldsymbol{\theta}_{-i}) \neq \phi_i'(\theta_i, \boldsymbol{\theta}_{-i})$, serial dictatorships are *individually contextually private*. We discuss both of these strengthenings in Section 7.

In the case of only two agents, serial dictatorship is the unique contextually private and efficient mechanism, as the following example shows.

EXAMPLE—Contextual Privacy and Serial Dictatorships, $n = 2$: Consider again the example from the introduction in Figure 1. There are two agents $N = \{1, 2\}$, each of which is allocated an object A or B. The two possible outcomes are $x = \{(1, A), (2, B)\}$ and $x' = \{(1, B), (2, A)\}$.

Table B.I shows possible assignments under efficiency. In the upper right table cell, efficiency requires that the outcome is x. In the lower left cell, efficiency requires that the outcome is x'. In the top left and bottom right cell, where both agents have the same type, efficiency allows either x or x'.

	A	B
A	x or x'	x
B	x'	x or x'

	A	B
A	x	x
B	x'	x

	A	B
A	x	x'
B	x	x

	A	B
A	x	x
B	x'	x'

	A	B
A	x'	x
B	x'	x

TABLE B.I

OUTCOMES FOR EXAMPLE IN FIGURE 1 UNDER AN ARBITRARY EFFICIENT CHOICE RULE (LEFT); UNDER AN EFFICIENT CHOICE RULE WHICH BREAKS TIES LEXICOGRAPHICALLY (ϕ^{FAIR}) (MID-LEFT, MIDDLE); UNDER A SERIAL DICTATORSHIP ϕ^{sd} (MID-RIGHT, RIGHT)

Four different assignments remain. The first two assignments contain a Corner in the sense of Corollary 3, hence are not contextually private. The other two are Serial Dictatorships corresponding to the agent orderings $\pi(1) = 1, \pi(2) = 2$ resp. $\pi(1) = 2, \pi(2) = 1$.

APPENDIX C: FIRST-PRICE AUCTION IS CONTEXTUALLY PRIVATE UNDER SEQUENTIAL ELICITATION

While second-price auctions are incompatible with contextual privacy, there are contextually private protocols of the first-price auction. Such a protocol is given by a descending protocol. A descending protocol queries, for each element of the type space $\tilde{\theta}$, in decreasing order, each agent 1 to n on whether their type θ_i is above θ. Formalized as a protocol, this leads to a set of nodes $\mathcal{N} \times \Theta \times \{0,1\}$ and edges from $((i,\theta,0)$ to $(i+1,\theta,0))$, for $i \in \mathcal{N} \setminus \{n\}$ and $\theta \in \Theta$. There are edges from $(n,\theta,0)$ to $(1, \max_{\theta'<\theta} \theta', 0)$. Furthermore, there are edges $((i,\theta,0),(i,\theta,1))$ for all $i \in \mathcal{N}$ and $\theta \in \Theta$ corresponding to an agent stating that they have a type θ, which leads to them being allocated the good. Hence, the set of terminal nodes is $\Theta \times \mathcal{N} \times \{1\}$. The associated set of type profiles is recursively defined as

$$\Theta_{(i+1,\theta,0),i} \coloneqq \Theta_{(i,\theta,0),i} \setminus \{\theta\}, \quad \Theta_{(i,\theta,1),i} \coloneqq \{\theta\}, \quad \Theta_{(1,\theta,0),i} \coloneqq \Theta_{(n,\max_{\theta'<\theta}\theta',0),i} \setminus \{\theta\}.$$

The first rules out the type θ for type i when they claim that they are not type θ. The second identifies an agent's type exactly when they claim they are type θ. The last rules out the type θ for agent n when they claim they are not θ and leads to the protocol considering the next-lowest type $\max_{\theta'<\theta}$.

PROPOSITION 16: *The descending protocol for the first-price rule ϕ^{FP} is contextually private under sequential elicitation.*

PROOF OF PROPOSITION 16: It suffices to show that the descending protocol is contextually private. Let $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ be separated. By definition of sequential elicitation, this must happen when agent i is queried. Note that terminal nodes cannot separate type profiles. Hence, $(\theta_i, \boldsymbol{\theta}_{-i})$ and $(\theta_i', \boldsymbol{\theta}_{-i})$ are separated at a node of the form $(i, \tilde{\theta}, 0)$. By definition of the descending protocol, the children of the node $(i, \tilde{\theta}, 0)$ are associated to sets

$$\{\tilde{\theta}\} \text{ and } \{\tilde{\theta}' \in \Theta \colon \tilde{\theta}' < \tilde{\theta}\}.$$

Let, without loss, $\theta_i = \tilde{\theta}$ and $\theta_i' < \tilde{\theta}$. In the former case, the outcome is that agent i gets the good at price $\tilde{\theta}$. By definition of the descending protocol, in the latter case, it is that either agent i does not get the good, or they get it at a price $\tilde{\theta}' < \tilde{\theta}$.

Note that by construction of the protocol, the first query leading to a singleton possible type space must be a type θ_i attaining $\max_{i \in \mathcal{N}} \theta_i$. This implies that the descending protocol is a protocol for the first-price choice rule (with tie-breaking according to the order $1, 2, \ldots, n$). *Q.E.D.*

Hence, the first-price choice rule is contextually private under sequential elicitation. The next example gives insight into the other kinds of standard auction choice rules that have contextually private protocols under sequential elicitation protocols.

EXAMPLE: Consider a standard auction rule in which the agent with the highest type wins and pays a price t where $t \colon \boldsymbol{\Theta} \to \mathbb{R}$ is an injective function. That is, the payment t that the winner pays is different for every type profile $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. In this case, the outcome $\phi(\boldsymbol{\theta}) = (q(\boldsymbol{\theta}), t(\boldsymbol{\theta}))$ is different for every type profile. Hence, contextual privacy is trivial, as $\phi(\boldsymbol{\theta}) \neq \phi(\boldsymbol{\theta}')$ for any $\boldsymbol{\theta} \neq \boldsymbol{\theta}'$, $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \boldsymbol{\Theta}$. Hence, any protocol for this auction rule is contextually private.

The example above shows that if the winner's payment can depend in an arbitrary way on the profile of bids, many standard auction choice rules are contextually private. However, with an additional condition on how the payment depends on the bid distribution, the first-price choice rule is the unique contextually private choice rule. We say that payments in an auction *depend only on rank* if the payment is a function of an order statistic, $t(\boldsymbol{\theta}) = f(\theta_{[k]})$, $k \in \mathcal{N}$.

PROPOSITION 17: *Consider the class of choice rules Φ that consists only of standard auctions where the payment t depends only on rank. Under sequential elicitation protocols, the first-price choice rule ϕ^{FP} is the unique efficient and contextually private standard auction rule in Φ.*

PROOF OF PROPOSITION 17: A similar construction as in the proof of Proposition 2. The quantile that the price depends on is chosen by two types. The Corners Lemma can be similarly applied. *Q.E.D.*

APPENDIX D: GROUP CONTEXTUAL PRIVACY AND OBVIOUS STRATEGYPROOFNESS

The following example illustrates that there are rules that are group contextually private but not obviously strategyproof.

EXAMPLE—Non-Clinching Rule: In particular, there are strategyproof choice rules that are not obviously strategyprouf but group-contextually private. As an example, consider $n = 2$, $\Theta = \{\underline{\theta}, \overline{\theta}\}$ and $\mathcal{X} = \{x_1, x_2, x_3, x_4\}$. Assume that for agent 1,

$$x_1 \succ_{\underline{\theta}} x_3 \succ_{\underline{\theta}} x_2 \succ_{\underline{\theta}} x_4$$

$$x_1 \prec_{\underline{\theta}} x_3 \prec_{\underline{\theta}} x_2 \prec_{\underline{\theta}} x_4$$

and for agent 2

$$x_1 \succ_{\underline{\theta}} x_2 \succ_{\underline{\theta}} x_3 \succ_{\underline{\theta}} x_4$$

$$x_1 \prec_{\underline{\theta}} x_2 \prec_{\underline{\theta}} x_3 \prec_{\underline{\theta}} x_4.$$

Consider the social choice function

$$\phi(\underline{\theta}, \underline{\theta}) = x_1 \qquad \phi(\underline{\theta}, \overline{\theta}) = x_2 \qquad \phi(\overline{\theta}, \underline{\theta}) = x_3 \qquad \phi(\overline{\theta}, \overline{\theta}) = x_4.$$

As ϕ is injective, any protocol for ϕ is group contextually private. It is also tedious but straightforward to check that this rule is strategyproof. There is no obviously strategyproof implementation, however. Assume that agent 1 is asked to play first. They face a choice between outcomes $\{x_1, x_3\}$ and $\{x_2, x_4\}$, which, for both $\underline{\theta}$ and $\overline{\theta}$ types are are not ordered in the set order, and hence make no action obviously dominated. A similar observation for agent 2 shows that neither first action can be obviously dominant.

APPENDIX E: DEFINITIONS OF PROTOCOLS

Algorithm 2: Ascending Protocol

```
Input: $N$ agents, each with willingness to pay $\theta_i \in \Theta$
Output: The remaining agent in set $R$ and price $\tilde{\theta}$
Data: $\tilde{\theta} \leftarrow \min(\Theta)$
Data: Set of remaining agents $R \leftarrow N$
while $|R| > 1$ do
    $R_{\text{next}} \leftarrow \emptyset$;
    foreach agent $i$ in $R$ do
        if $\theta_i \geq \tilde{\theta}$ then
            Add agent $i$ to $R_{\text{next}}$;
    $R \leftarrow R_{\text{next}}$;
    $\tilde{\theta} \leftarrow \mathrm{succ}_\Theta \tilde{\theta}$;
```

Algorithm 3: Descending Protocol

```
Input: $N$ agents, each with willingness to pay $\theta_i \in \Theta$
Output: Agent winner and price $\tilde{\theta}$
Data: $\tilde{\theta} \leftarrow \max(\Theta)$
while true do
    foreach agent $i$ in $R$ do
        if $\theta_i \geq \tilde{\theta}$ then
            return winner $i$ and price $\tilde{\theta}$
        $\tilde{\theta} \leftarrow \mathrm{pred}_\Theta(\tilde{\theta})$;
```

Algorithm 4: Overdescending Protocol

```
Input: $N$ agents, each with willingness to pay $\theta_i \in \Theta$
Output: Agent winner and price $\tilde{\theta}$
Data: $\tilde{\theta} \leftarrow \max(\Theta)$
while true do
    $\mathrm{winnerFound} \leftarrow$ False;
    $\mathrm{winner} \leftarrow \emptyset$;
    foreach agent $i$ in $R \setminus$ winner do
        if $\mathrm{winnerFound}$ then
            if $\theta_i \geq \tilde{\theta}$ then
                $\mathrm{winnerFound} \leftarrow$ True;
                $\mathrm{winner} \leftarrow \{i\}$;
        else
            if $\theta_i \geq \tilde{\theta}$ then
                return Agent winner and price $\tilde{\theta}$;
    $\tilde{\theta} \leftarrow \mathrm{pred}_{\Theta}(\tilde{\theta})$;
```

Algorithm 5: Ascending Join Protocol

```
Input: Ordered list $N$ of agents , each with willingness to pay $\theta_i \in \Theta$
Output: Agent $i$ and price $\tilde{\theta}$
Data: $\tilde{\theta} \leftarrow \min(\Theta)$
Data: $\mathrm{active} \leftarrow$ first two agents from $N$
while true do
    foreach agent $i$ in $\mathrm{active}$ do
        if $\theta_i < \tilde{\theta}$ then
            Remove $i$ from $\mathrm{active}$;
            if There is agent $i$ that never has been active then
                $\mathrm{active} \leftarrow \mathrm{active} \cup \{i\}$;
            else
                return Remaining agent i from active, $\mathrm{pred}_{\Theta}(\tilde{\theta})$
```