CONTEXTUALLY PRIVATE MECHANISMS

ANDREAS HAUPT[*]

ZOË HITZIG[†]

Abstract. Consider a mechanism designer who employs a dynamic protocol to implement a choice rule. A protocol *violates the contextual privacy* of an agent if the designer learns more of the agent's private information than is necessary for computing the outcome. Our first main result is a characterization of choice rules that can be implemented without producing any contextual privacy violations. We apply this result to show that many commonly studied and employed choice rules violate some agent's contextual privacy—the first-price auction and serial dictatorship rules are notable exceptions that can avoid violations altogether. Our second main result is a representation theorem for protocols that are *contextual privacy equivalent*. We use this result to derive a novel protocol for the second-price auction choice rule, the *ascending-join* protocol, which is more contextually private than the widespread ascending or "English" protocol.

1. INTRODUCTION

In standard mechanism design, a designer elicits agents' private information in order to determine the outcome of a social choice rule. Ex-post, in an incentive compatible mechanism, the designer learns *all* of agents' private information—typically more than is strictly necessary for computing the rule. For example, in a sealed-bid second price auction, the designer learns all losing bids exactly, even though it is only necessary to know that all losing bids fall below the second highest. The designer also learns the winner's bid exactly, even though it is only necessary to know that the winning bid is above the second highest.

At the same time, there are several reasons why it may be desirable that the designer not learn "too much" about agents. First, gaining excess knowledge of agents' private information could expose the designer to legal liabilities or political risk. For example, as recounted in McMillan (1994), a second-price auction for spectrum licenses in New Zealand had a

---

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

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

Date: January 26, 2024. 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, Navin Kartik, Ludvig Sinander, 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).

"political defect": "by revealing the high bidder's willingness to pay, the auction exposed the government to criticism, because after the auction everyone knew that the firm valued the license at more than it paid for it" (McMillan, 1994). The government would have benefited from a design that ensured it only learned what was strictly needed—with the idea that, as the adage goes, what they don't know can't hurt them. Second, agents may have intrinsic privacy concerns (Acquisti et al., 2016) or they may worry that their private information could be used against them in subsequent interactions with the designer or third parties (Rothkopf et al., 1990, Ausubel, 2004). If agents have privacy concerns, they may be reluctant to reveal their private information even if the appropriate allocative incentives are in place.[1]

In this paper, we study mechanisms that minimize the superfluous information learned by a designer. In our set up, when a designer commits to a social choice rule, they also choose a dynamic protocol for computing the rule. These dynamic protocols allow the designer to learn agents' private information in a minimal way, ruling out type profiles until they know enough to compute the outcome. A protocol produces a *contextual privacy violation* for a particular agent if the designer learns a piece of their private information that was not needed for computing the outcome. We call protocols that produce no violations for any agent at any type profile *contextually private* protocols.[2]

We define protocols in a way that allows us to accommodate a broad range of social and technological environments. A protocol is composed of queries. A query can be directed to one agent—for example, a designer may ask one agent "Is your type above x?" Or, a query can be directed to multiple agents—e.g. the designer could ask "How many agents have a type above x?" Formally, a 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. The details of

[1] For a recent example where superfluous information learned by the designer may have been used against participants, consider Google's second-price auctions for advertising. A 2022 lawsuit *State of Texas v. Google* alleged that "Google induced advertisers to bid their true value, only to override pre-set AdX floors and use advertisers' true value bids against them... generat[ing] unique and custom per-buyer floors depending on what a buyer had bid in the past."

[2] As we will discuss further when we review related literature (Section 8), contextual privacy is analogous, in some environments, to the concept of *unconditional privacy* in decentralized computing (Chor and Kushilevitz, 1989, Brandt and Sandholm, 2005) and generalizes the notion of *unconditional winner privacy* studied in an auction context in Milgrom and Segal (2020).

the environment may dictate the format of queries the designer is able to ask—we call the set of available queries the *elicitation technology*.

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 this 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.[3]

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.[4] In addition, sophisticated cryptographic mechanisms require sophisticated participants—if participants do not understand how their information is kept private, their privacy concerns may not be alleviated.[5]

We focus on protocols founded on minimal assumptions regarding trust and comprehension. Specifically, for the most design-relevant portions 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. The designer knows something if and only if an agent said it.

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

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

[4]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).

[5]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).

elicitation technology allows them to count the number of agents whose type satisfies a 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 central definition of the paper and to introduce a key theme of our results, we turn to a simple example.

1.1. *Introductory Example.*

Consider a setting with 4 agents and 2 identical objects. The agents' private information is their value for the object, which we will call the agent's "type." The agents have integer values for the object in the range from 0 to 10.

TABLE I

INTRODUCTORY EXAMPLE: COMPARING TWO PROTOCOLS FOR $\phi^{\text{EFFICIENT}}$.

	Ascending Protocol	Descending Protocol
Definition	1. For each i ask $\theta_i \geq 1$? 2. For each i ask $\theta_i \geq 2$? ⋮ k. For each i ask $\theta_i \geq k$?	1. For each i ask $\theta_i \leq 9$? 2. For each i ask $\theta_i \leq 8$? ⋮ k. For each i ask $\theta_i \leq 10 - k$?
CP?	no	no
Agent Violations	violates losers' privacy	violates winners' privacy

The designer aims to compute a choice rule ϕ, and learns about the type profile through a dynamic protocol. A protocol is *contextually private* for a choice rule ϕ if, at all possible type profiles, 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 for all agents and all possible types, the designer can only distinguish between possible types of agents in the event that the two types lead to different outcomes under ϕ, holding other agent types fixed.

To see how a protocol can fail to be contextually private, we consider a specific choice rule. Suppose a designer wants to allocate the two objects efficiently, and there are no transfers. We call this choice rule $\phi^{\text{efficient}}$.[6] The designer chooses a protocol to learn about the type space, proceeding with queries that refine their knowledge of the type profile until they have found the agents with the 2 highest values.

We will consider two natural protocols that the designer could use to compute the choice rule $\phi^{\text{efficient}}$: the *ascending* and *descending* protocols.[7] First, consider the ascending protocol, in which the designer asks all agents if their value for the good is above 1, then asks all agents if their value for the good is above 2, and so forth. The designer stops asking questions when they have enough information to compute the efficient allocation.

The ascending protocol for the efficient choice rule is not contextually private. To see this, suppose the type profile is $(1, 2, 5, 9)$. Following the ascending protocol, the designer learns two agents' types exactly, and learns that the other two agents have types greater than or equal to 3, i.e. they learn that the type profile is contained in the subset of type profiles $\{1\} \times \{2\} \times [3, 10] \times [3, 10]$. Note that, holding all other agents' types fixed, if agent 1 had a value of 2 instead of 1, it would not make a difference to the outcome. Similarly, holding

[6] Formally, let agent i's type be θ_i, and let the profile of agent types be $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3, \theta_4)$. The rule the designer wants to implement is $\phi^{\text{efficient}}(\boldsymbol{\theta}) = (\phi_1(\theta_1), \dots, \phi_4(\theta_4))$ with

$$\phi_i(\theta_i) = \begin{cases} 1 & \text{if } \theta_i \geq \boldsymbol{\theta}_{[2]} \\ 0 & \text{otherwise,} \end{cases}$$

where $\boldsymbol{\theta}_{[2]}$ is the 2nd order statistic of the profile of agent types.

[7] For simplicity, this example abstracts away from agent strategies, assuming that agents truthfully respond to all of the designer's queries. Our formal framework, however, will accommodate indirect messaging protocols with generic messaging strategies.

all other agents' types fixed, if agent 2 had a value of 1 instead of 2, it would not make a difference to the outcome. Because contextual privacy requires that there are no violations at *any* possible type profile, we have immediately shown that the ascending protocol is not contextually private: For the losers (agents 1 and 2), the designer *only* needed to know that they had values below 3—the designer did not need to learn their exact values.

Next we turn to the descending protocol, which is also not contextually private. Following the descending protocol, the designer learns the winners' types exactly and that all other agents have types below 5. That is, the designer learns that the true type profile is contained in the subset of type profiles $[0,4] \times [0,4] \times \{5\} \times \{9\}$. This protocol is not contextually private by similar logic to that of the ascending protocol, except here, it is the winners' types that did not need to be learned exactly.

This introductory example, summarized in Table I, illustrates two key themes of the paper. First, neither the ascending nor the descending protocol for $\phi^{\text{efficient}}$ is contextually private—we will show in Section 4 that under our minimal trust assumption, it is hard to find contextually private protocols for many choice rules. Second, the fact that the ascending protocol and descending protocol produce disjoint contextual privacy violations sharpens our understanding of the privacy tradeoffs of different design choices. In Section 5, we discuss in depth how different design choices lead to different contextual privacy violations for different agents.

1.2. *Overview.*

We now present a brief overview of the paper. In Section 2, we articulate our formal framework and present the key definitions of the paper.

In Section 3, we discuss which choice rules have protocols that produce no contextual privacy violations for any agent. Our first main result, Theorem 1, is a characterization of contextually private choice rules under a fixed arbitrary set of 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 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 under sequential elicitation technologies.

Next, in Section 4, we apply the necessary condition from Theorem 1 to understand which choice rules are contextually private under a minimal assumption on the designer's elicitation technology. After noting that the first-price auction and the serial dictatorship choice rules are contextually private, we focus on the limits of contextual privacy under sequential elicitation protocols in assignment, auction and voting domains. 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 2); there is no contextually private stable matching rule in matching with priorities (Proposition 3); the second-price auction rule does not have a contextually private protocol (Proposition 4); there is no efficient double auction rule that is contextually private (Proposition 5); and finally that the generalized median voting rule is not contextually private (Proposition 6).

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. We consider contextual privacy equivalence: a protocol is *contextual privacy equivalent* to another protocol if the two protocols produce contextual privacy violations for the same set of agents. Our second main result, Theorem 2, considers choice rules defined on ordered type spaces and shows that every choice rule is equivalent to a *bimonotonic* protocol, so it is without loss for the designer to consider only bimonotonic protocols. A bimonotonic protocol consists of *threshold queries* which, for each agent, are monotonically increasing or decreasing in the threshold.

We also consider how a designer might select among protocols in a particular contextual privacy equivalence class. We say that a protocol *contextual privacy improves* on another protocol if, taking into account the type profiles that produce violations, the set of contextual privacy violations can be reduced. We observe that the ascending protocol for the second-price auction can be contextual privacy improved: what we call *ascending-join protocols* are contextual privacy improvements on the ascending protocol (Proposition 7).

Every sequential elicitation protocol induces a well-defined extensive-form game. In Section 6, we "check" that three protocols we found to have good contextual privacy properties in Section 4 and Section 5 also allow the designer to "implement" the choice rule in obviously dominant strategies or in perfect Bayesian strategies. This section relies mostly on prior results (from e.g. Li (2017)), and is intended to be an initial step toward a more complete understanding of how contextual privacy interacts with incentives.

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 and conclude in Section 9.

Proofs omitted from the main text are in Appendix A.

2. MODEL

Consider a set $N = \{1, 2, \ldots, n\}$ of agents with private information (or "types") θ_i distributed according to $F_i \in \Delta(\Theta)$, where Θ is a finite type space. We denote by $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n) \in \Theta^n = \boldsymbol{\Theta}$ a profile of agents' types. Agents have utility functions u over outcomes in X which depend on their private types, $u_i : \Theta \times X \to \mathbb{R}$. All primitives of the model besides the realized type profile $\boldsymbol{\theta}$ are common knowledge.

The designer wishes to implement a social choice function $\phi\colon \boldsymbol{\Theta} \to X$ through a dynamic protocol.[8] In a protocol, agents repeatedly send *messages* $m_i \in M$, where a profile of agent messages is $\mathbf{m} = (m_1, \ldots, m_n) \in \mathbf{M}$.

2.1. *Elicitation Technologies.*

We will formally define protocols in the following subsection. For now it is enough to think of them as a series of questions that the designer can ask about submitted message profiles at each round. An *elicitation technology* captures the partitions of the space of message profiles that are indistinguishable to the designer. Formally, hence, an elicitation

[8]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, and unless other noted otherwise, lexicographic, tie-breaking.

technology is a collection of partitions of the space of message profiles, i.e. a set of partitions $\mathfrak{S} \subseteq 2^{(2^{\mathbf{M}})}$.

At each stage of the protocol, the designer chooses a partition $\mathcal{S} \in \mathfrak{S}$ and learns which of the partition cells the submitted message profile lies in. At any point, the protocol may stop and choose an outcome $x \in X$.

We next give two examples of particular elicitation technologies which will feature in our analysis. The first elicitation technology, the sequential elicitation technology, was already introduced informally in Section 1 and corresponds to a minimal assumption about the agent's trust.

EXAMPLE—Sequential Elicitation Technology: The *sequential elicitation* technology $\mathfrak{S}_{\mathrm{SE}}$ is a collection of partitions of the message space based on a single agent's message. This elicitation technology contains n elements, $\mathcal{S}_i \in 2^{(2^{\mathbf{M}})}$ for $i = 1, \ldots, n$. Each element $\mathcal{S}_i$ is a partition of the message space, i.e.

$$\mathcal{S}_i = \{\{\mathbf{m} \in \mathbf{M} : m_i = m\} : m \in M\}. \tag{1}$$

The interpretation of the sequential elicitation technology is that the designer can ask one agent at a time to send a message about the cell of a given partition in which their message lies. This is a result of the fact that $(m_i, \mathbf{m}_{-i})$ and $(m_i, \mathbf{m}_{-i})$ are indistinguishable for the designer for any $i = 1, 2, \ldots, n$, $m_i \in M$ and $\mathbf{m}_{-i} \in \mathbf{M}_{-i}$. The sequential elicitation technology corresponds to a minimal assumption on trust because agents are queried one at a time, and thus immediately identified with their messages: there is no mediator or mediating technology that could shield the agent's message from their identity.

Another example of an elicitation technology is the *count elicitation technology,* under which a designer can observe the number of messages, without learning the identities of agents who send this message. This elicitation technology corresponds to the common practice of anonymized elicitation (for example, many political elections use a secret ballot and online auctions permit anonymous bidding).

EXAMPLE—Count Elicitation Technology: The *count elicitation technology* $\mathfrak{S}_{\text{Count}}$ is

$$\mathfrak{S}_{\text{Count}} = \{\{\mathbf{m} \in \mathbf{M} \colon |\{m_i = m\}| = k\} : k = 1, 2, \dots, n, m \in M\}.$$

The count elicitation technology requires more trust than the sequential elicitation technology in the sense that it requires "mediation". To see this, suppose the designer uses the count elicitation technology to ask "How many agents have a message above x?" Then, in order for it to be the case that the designer *only* learns the number, and not the identity of the agents, there must be a technology that anonymizes agents messages when they are sent. In practice, count elicitation technologies could be a ballot box, a third-party mediator, or another trusted anonymization technique.

2.2. *Protocols and Messaging Strategies.*

We represent protocols, i.e. dynamic elicitation processes, as directed rooted trees. All nodes encode histories of questions based on an elicitation technology. The root node r corresponds to the initial, empry history. Terminal nodes $z \in Z$ are the final queries at which the outcome $x \in X$ is determined.

DEFINITION—Protocol: A *protocol* $P = (V, E, r, Z)$ with elicitation technology $\mathfrak{S}$ is a directed tree with nodes V, edges E, root $r \in V$ and a set of terminal nodes Z. Each non-terminal node v is labelled with a query $s_v \colon \mathbf{M} \to \text{children}(v)$, such that the partition induced by the preimages of s_v lies in $\mathfrak{S}$, that is

$$\{s_v^{-1}(w) \subseteq \mathbf{M} : w \in \text{children}(v)\} \in \mathfrak{S}.$$

As a slight abuse of notation, we also write $s_v \in \mathfrak{S}$ in this case.

We refer to a protocol with elicitation technology $\mathfrak{S}$ as an $\mathfrak{S}$-protocol, and denote by $\mathcal{P}_{\mathfrak{S}}$ the set of protocols with elicitation technology $\mathfrak{S}$. Table II compiles notation. When it is not clear from the context which protocol a given query belongs to, we will mark queries s_v by a superscript for the relevant protocol, s_v^P.

It will be helpful to capture what the designer knows about the type profile at node $y \in V$. Denote the *information available to the designer at node* y as

$$\mathbf{\Theta}_y := \bigcap_{(v,w)\in \mathrm{path}(y)} \boldsymbol{\sigma}_v^{-1}(s_v^{-1}(w)),$$

where, $\mathrm{path}(y)$ is the set of edges from the root node r to the node w. So $\mathbf{\Theta}_y$ is the set of type profiles that are possible under the protocol P when node y is reached.

Agents submit messages according to deterministic strategies σ_i, which map non-terminal nodes and type spaces into messages, i.e. $\sigma_i\colon (V \setminus Z) \times \Theta \to M$. That is, depending on their type θ_i, an agent chooses a message to send at each non-terminal node V. The state of the protocol is public: The agents know the exact history of previously asked queries, and the outcomes that were observed by the designer, so that the information on which agent i's strategy depends at node v is her own type θ_i in conjunction with the information known at node v i.e. $\mathbf{\Theta}_v$. Depending on the elicitation technology used, the messages of many agents at a particular node may be inconsequential. For example, sequential elicitation queries only take into account a single agent's message at each node v.

We define the terminal node reached in protocol P from strategy profile σ inductively. At a non-terminal node $v \in V \setminus Z$, the successor node is determined as

$$s_v(\sigma_1(\theta_1, v), \sigma_2(\theta_2, v), \ldots, \sigma_n(\theta_n, v)).$$

This defines a path, and a terminal node $z \in Z$ for every type profile $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n) \in \mathbf{\Theta}$, which we denote by $P(\boldsymbol{\sigma}(\boldsymbol{\theta}))$. We also say that $\boldsymbol{\theta}$ *leads to* terminal node $z = P(\boldsymbol{\sigma}(\boldsymbol{\theta})) \in Z$ and outcome $x \in X$.

We say that P is a protocol *for* choice rule ϕ with strategies $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ if

$$P(\boldsymbol{\sigma}(\boldsymbol{\theta})) = \phi(\boldsymbol{\theta}).$$

If there is an $\mathfrak{S}$-protocol for ϕ with some strategies $\boldsymbol{\sigma}$, we say that it is $\mathfrak{S}$-*computable*.

TABLE II

NOTATION FOR PROTOCOLS AND RELATIONS

Name	Sets	Representative Element
Agents	$\{1,\ldots,n\}$	i
Type profiles	$\boldsymbol{\Theta} = \Theta^n$	$\boldsymbol{\theta} = (\theta_1,\ldots,\theta_n)$
Message profiles	$\mathbf{M} = M^n$	$\mathbf{m} = (m_1,\ldots,m_n)$
Elicitation technology	$\mathfrak{S}$	$\mathcal{S},\ s_v$
Protocols		$P = (V,E)$
Nodes	V	v,w
Edges	E	$e = (v,w)$
Terminal nodes of protocol P	Z	z
Information at node v	$\boldsymbol{\Theta}_v$	$\boldsymbol{\theta}_v$
	Relations	
Node v precedes v' in protocol P		$v \succ_P v'$
Protocol P' is contextual privacy equivalent to P for ϕ		$P' \sim_\phi P$
Type θ is succeeded by θ' in the type space		$\theta' = \mathrm{succ}(\theta)$ or $\theta = \mathrm{pred}(\theta')$

2.3. *Contextual Privacy Violations and Contextual Privacy.*

Protocols give rise to a notion of *distinction* for type profiles $\boldsymbol{\theta},\boldsymbol{\theta}' \in \boldsymbol{\Theta}$.

DEFINITION—Distinction: A protocol $P = (V,E,r)$ with strategies $\boldsymbol{\sigma}$ *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 $\boldsymbol{\theta} \in \boldsymbol{\Theta}_w$ and $\boldsymbol{\theta}' \in \boldsymbol{\Theta}_{w'}$.

Note that type profiles $\boldsymbol{\theta},\boldsymbol{\theta}' \in \boldsymbol{\Theta}$ lead to different terminal nodes $z \neq z' \in Z$ if and only if they are distinguished at some node $v \in V$.

The main idea of this article is the idea of *contextual privacy violations*. A contextual privacy violation is a piece of agent i's private information learned by the designer that did not play a role in determining the outcome. All of the key concepts of the paper employ this definition.

DEFINITION—Contextual Privacy Violation: Let P be a protocol for ϕ with strategies $\boldsymbol{\sigma}$. We say that P and σ produce a ϕ-*contextual privacy violation* for agent i at $\boldsymbol{\theta},\boldsymbol{\theta}' \in \boldsymbol{\Theta}$ with $\boldsymbol{\theta} = (\theta_i,\boldsymbol{\theta}_{-i})$ and $\boldsymbol{\theta}' = (\theta_i',\boldsymbol{\theta}_{-i})$, if $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are distinguished at some node $v \in V$ in P under $\boldsymbol{\sigma}$, and $\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}')$. We denote by $\Gamma(P) \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$ the set of contextual privacy violations produced by protocol P.

In other words, there is a contextual privacy violation for choice rule ϕ in protocol P with strategies $\boldsymbol{\sigma}$ if the designer can tell types θ_i, θ_i' apart for *some* partial type profile $\boldsymbol{\theta}_{-i} \in \boldsymbol{\Theta}_{-i}$ for the other agents, but this leads to the same outcome under ϕ. A contextual privacy violation is an instance of the designer learning more than they *need to know* to compute the outcome of the choice rule ϕ.

DEFINITION—Contextual Privacy of Protocols: A protocol P with strategies $\boldsymbol{\sigma}$ is *contextually private at type profile* $\boldsymbol{\theta} = (\theta_{-i}, \boldsymbol{\theta}_i)$ for agent i if there does not exist a $\boldsymbol{\theta}' = (\theta_i', \boldsymbol{\theta}_{-i})$ that produces a contextual privacy violation. We say that P is *contextually private* if it is contextually private for all agents i at all type profiles $\boldsymbol{\theta} \in \boldsymbol{\Theta}$.

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.[9]

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 1.

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 of Choice Rules: A choice rule ϕ is *contextually private with elicitation technology* $\mathfrak{S}$ if there exists an $\mathfrak{S}$-protocol for ϕ that is contextually private.

[9] Another way of understanding contextual privacy is that there is no agent i or type profile $\boldsymbol{\theta}$ at which protocol P produces a contextual privacy violation.

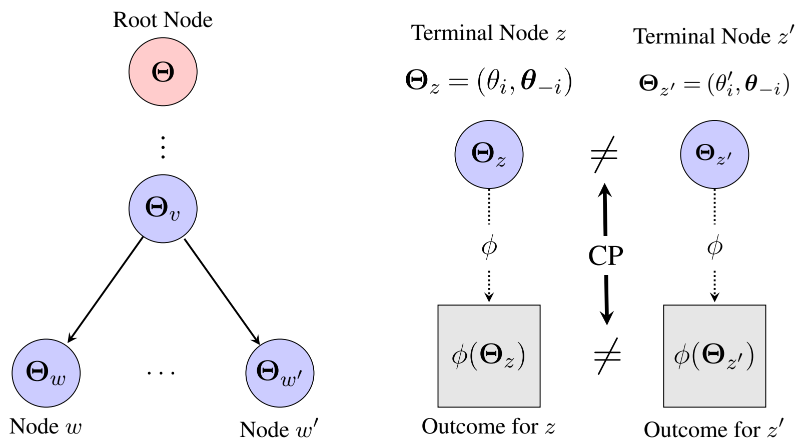

FIGURE 1.—Definitions: Generic direct protocol (left), contextually private protocol (right)

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 referring to a 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. Note that with permissive elicitation technologies, all choice rules may be (trivially) contextually private. Contextual privacy is non-trivial when the designer's elicitation technology is restricted.

2.4. *Direct Protocols*

In the rest of this section, and most of this paper (with the exception of Section 6 and Section 7), we will restrict attention to a class of protocols we call *direct*.

To define direct protocols, we will assume that there is an injective mapping $\iota\colon \Theta \to M$ that maps $\theta_i \in \Theta$ to *corresponding* messages $m \in M$. We say that strategy $\sigma_i\colon \Theta \times V \to M$ at node v is *truthful* if $\sigma_i(\theta_i, v) = \iota(\theta_i)$. As we assume ι to be an injection, we may write $\Theta \subseteq M$ to identify the types with a subset of messages. With this notation, σ_i is truthful if $\sigma_i(\theta_i, v) = \theta_i$ for all $v \in V$ and $\theta_i \in \Theta$. We also say that a protocol P with truthful strategies $\boldsymbol{\sigma}$ is *direct*.

In the rest of this subsection, we will show that for any $\mathfrak{S}$-protocol P there is a $\mathfrak{S}^*$-protocol P^* with the same contextual privacy violations, for a related elicitation technology $\mathfrak{S}^*$. We call $\mathfrak{S}^*$ the *strategy-enhanced elicitation technologies*.

DEFINITION: Let $\mathfrak{S}$ be an elicitation technology with element partitions $\mathcal{S}$. A partition $\mathcal{S}^*$ is contained in the *strategy-enhanced elicitation technology* $\mathfrak{S}^*$ if and only if there exists a partition $\mathcal{S} \in \mathfrak{S}$ and a set of functions $f_1, f_2, \ldots, f_n \colon \Theta \to M$ such that

$$\mathcal{S}^* = \{(f_1, f_2, \ldots, f_n)^{-1}(\{\mathcal{S}\}) : \mathcal{S} \in \mathfrak{S}\}. \tag{2}$$

Here, $(f_1, f_2, \ldots, f_n)^{-1}(\{\mathcal{S}\})$ is the set of message profiles $\mathbf{m} = (m_1, m_2, \ldots, m_n)$ such that the partition $(f_1(m_1), f_2(m_2), \ldots, f_n(m_n)) \in \mathfrak{S}$.

Less formally, the technology $\mathfrak{S}^*$ consists of partitions of messages that the designer can identify if she can "simulate" mappings $f_1, f_2, \ldots, f_m \colon \Theta \to M$ before observing which set S of a partition $\mathcal{S} \in \mathfrak{S}$ the messages lie in. Formally, $\mathfrak{S}^*$ will depend on ι. We will suppress this dependence for brevity of notation. In our reduction to direct protocols, $f_1, f_2, \ldots, f_m$ will simulate agents' messaging strategies.

Under sequential elicitation technologies as defined in (1) above, the strategy-enhanced protocols are those in which agents are queried for subsets of types instead of messages.

EXAMPLE—Strategy-Enhanced Sequential Elicitation Technology: For the sequential elicitation technology $\mathfrak{S}_{\mathrm{SE}}$, the strategy-enhanced technology $\mathfrak{S}^*_{\mathrm{SE}}$ consists of partitions

$$\{\{\mathbf{m} \in \mathbf{M} : m_i \in \tilde{\Theta}_l\} : l = 1, 2, \ldots, k\}$$

for all agents $i \in N$ and all partitions $\Theta = \Theta_1 \cup \Theta_2 \cup \cdots \cup \Theta_k$, with $1 \leq k \leq |\Theta|$.[10] This technology allows the designer to request that a single agent identify in which cell of a partition their type lies.

The strategy-enhanced count technologies are those that partition the type space of agents, potentially differently per agent, and count those.

[10]Concretely, the partition for agent i and partition $\Theta_1 \cup \Theta_2 \cup \cdots \cup \Theta_k = \Theta$ is induced in (2) by $\mathcal{S} = \mathcal{S}_i$, and functions that project θ_i on representative elements from the partition cells $\Theta_1, \Theta_2, \ldots, \Theta_k$. That is,

$$f_i \colon \theta_i \mapsto \tilde{\theta}_l \text{ such that } \theta_i \in \Theta_l$$

for representative elements $\tilde{\theta}_l \in \Theta_l$.

EXAMPLE—Strategy-Enhanced Count Technology: The strategy-enhanced technology associated with this elicitation technology, $\mathfrak{S}^*_{\text{Count}}$ consists of type profile partitions

$$\mathfrak{S}^*_{\text{Count}} = \{\{\boldsymbol{\theta} \in \boldsymbol{\Theta}\colon |\{\theta_i \in \Theta_{il}\}| = k\} : k = 1, 2, \ldots, n\}$$

for all partitions $(\Theta_{il})_{l=1,2,\ldots,|M|}$ of Θ. This technology allows us to distinguish type profiles that have different numbers of agents that are contained in some cell of (potentially individualized) partitions.

The following lemma establishes that, for the purposes of understanding contextual privacy desiderata, we can restrict attention to direct protocols.

LEMMA 1: *There is an $\mathfrak{S}$-protocol P with strategies $\boldsymbol{\sigma}$ for ϕ with contextual privacy violations $\Gamma \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$ if and only if there is a direct $\mathfrak{S}^*$-protocol P^* for ϕ with contextual privacy violations $\Gamma \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$.*

PROOF: We first show sufficiency. Let P be a $\mathfrak{S}$-protocol for ϕ with strategies $\boldsymbol{\sigma}$. Define P^* where for all $v \in V \setminus Z$, $s_v^{P^*}(\boldsymbol{\theta}) = s_v^P(\boldsymbol{\sigma}(\boldsymbol{\theta}))$. For $z \in Z$, assign the same outcome under P^* as under P.

First, we show that P^* is an $\mathfrak{S}^*$-protocol. Note that $(s_v^{P^*})^{-1}(w) = \boldsymbol{\sigma}^{-1}(v, \cdot)((s_v^P)^{-1}(w))$ and hence $s_v^{P^*} \in \mathfrak{S}^*$ as $(s_v^P)^{-1}(\{w\}) \in \mathcal{S}$ and $\boldsymbol{\sigma}$ consists of mappings $\Theta \to M$.

Next, we observe that, by construction, P^* is a protocol for ϕ in that each type profile $\boldsymbol{\theta}$ leads to the same terminal node under P as in P^*.

Finally, we show that the information of the principal is the same at the terminal nodes that each type profile leads to in P and P^*. We show this by induction over the nodes v. It is clear at the root node $r \in V$ that the principal's information is $\boldsymbol{\Theta}_r = \boldsymbol{\Theta}$ in P and P^*. The fact that $s_v^{P^*}(\boldsymbol{\theta}) = s_v^P(\boldsymbol{\sigma}(\boldsymbol{\theta}))$ implies that this holds for all children w of $v \in V$, which yields the inductive step. This concludes the proof of sufficiency.

For necessity, assume that $P^* = (r, V, E, Z)$ is a direct $\mathfrak{S}^*$-protocol. By definition of $\mathfrak{S}^*$, for all nodes $v \in V \setminus Z$, there are $f_1, f_2, \ldots, f_n\colon \Theta \to M$ such that there is $s_v \in \mathfrak{S}$ and

$$s_v^*(\boldsymbol{\theta}) = s_v(f_1(\theta_1, v), f_2(\theta_2, v), \ldots, f_n(\theta_n, v)). \tag{3}$$

We define the protocol P where all queries s_v^* are replaced by s_v together with strategies $\sigma_i = f_i$, $i \in N$. By assumption, P is a $\mathfrak{S}$-protocol. Given the definition of P, $s_v^{P^*}(\boldsymbol{\theta}) = s_v^P(\boldsymbol{\sigma}(\boldsymbol{\theta}))$ also holds in this case, and show that P is a protocol for ϕ leading to the same information for the designer as P^* at every node $v \in V$. In particular, P and P^* have the same contextual privacy violations. *Q.E.D.*

This lemma allows us to separate contextual privacy concerns from implementation concerns. Implementation concerns center on whether there exists a truthful strategy profile $\boldsymbol{\sigma}$. To focus on the demands of contextual privacy, we assume in Section 3, Section 4 and Section 5 that there is such a strategy profile and that the designer knows it.

This reduction is highly abstract, but a brief and informal concrete example helps to show that it is relatively straightforward. Consider a sealed bid first-price auction in the standard independent private values environment. Consider the following protocol P: Each agent is queried once, and asked to identify their type in the partition of the type space that includes only singletons. This protocol is an $\mathfrak{S}_{\mathrm{SE}}$-protocol. Assume the message space is M is equal to the type space Θ. Suppose agents follow something like the symmetric Bayes-Nash equilibrium strategy in a static first-price sealed bid auction, $\sigma(\theta_i, v) = \chi\theta_i$ where $\chi \in [0, 1)$, for all agents i.[11] Agents submit bids $\sigma(\theta_i) \in M$. The strategy-enhanced elicitation technology $\mathfrak{S}^*_{\mathrm{SE}}$ is *enhanced* with the strategy function, allowing us to talk about agents' messages (reports of types) instead of types themselves. The lemma states that any contextual privacy violation in the $\mathfrak{S}_{\mathrm{SE}}$-protocol P is also present in a suitably defined $\mathfrak{S}^*_{\mathrm{SE}}$-protocol P^*, and vice versa. Note that this reduction requires that the designer knows $\boldsymbol{\sigma}$, and thus can "simulate" the agents' reporting.

2.5. *Beyond Contextual Privacy.*

In some cases, under restrictive elicitation technologies $\mathfrak{S}$, it will be hard to find contextually private protocols for desired choice rules. In such cases, we can still try to simplify the search for privacy-preserving protocols by finding representatives that are *as contextu-*

[11] We are not claiming that this strategy is in fact an equilibrium, we are just considering this strategy for illustrative purposes.

ally private as other protocols, and select among those using contextual privacy violations. We will consider these questions in Section 5.

We say that two protocols are *contextual privacy equivalent* if they produce contextual privacy violations for the same agents. This notion of equivalence captures the idea that the key difference between two protocols that are not contextual private is *whose* privacy is violated.

DEFINITION—Contextual Privacy Equivalence: We say that a protocol P is *contextual privacy equivalent* to P' under ϕ if for all agents i, and for all type profiles $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i}), \boldsymbol{\theta}' = (\theta_i, \boldsymbol{\theta}_{-i}) \in \boldsymbol{\Theta}$,

$$\bigvee_{\boldsymbol{\theta}' \in \boldsymbol{\Theta}} \Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \iff \bigvee_{\boldsymbol{\theta}' \in \boldsymbol{\Theta}} \Gamma(P', \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}'). \tag{4}$$

We denote this equivalence by $P \sim_\phi P'$.

This notion of equivalence means that for some fixed type profile $\boldsymbol{\theta}$, the set of agents $i = 1, 2, \ldots, n$ who have a contextual privacy violation with some other type profile $\boldsymbol{\theta}'$ is the same. In other words, a protocol P for ϕ is contextual privacy equivalent to a protocol P' for ϕ if the same agents have a privacy violation in P as in P'.[12]

In some cases, we may be interested in further refining among protocols that produce violations for the same agents. For this, we define a notion under which even two contextual privacy equivalent protocols may be compared. A protocol P for ϕ *improves* on a protocol P' for ϕ if there the set of privacy violations under P is a strict subset of the set of privacy violations under P'.

DEFINITION—Improvement: We say that P is a *strict ϕ-improvement* of P' if

$$\Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \implies \Gamma(P', \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}'), \tag{5}$$

[12] 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.

and not $\Gamma(P, \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}') \iff \Gamma(P', \phi, i, \boldsymbol{\theta}, \boldsymbol{\theta}')$.

For some choice rules, there may be protocols that are neither equivalent, nor are improvements of one another.[13]

2.6. *Incentives.*

For the most part, this paper focuses on contextual privacy properties of specific protocols, keeping privacy concerns distinct from implementation concerns. So, the most substantive sections of the paper— Section 3, Section 4, and Section 5—all assume truthful protocols. We will consider, to some extent in Section 6 the dynamic incentives of a few particular protocols shown to have "good" contextual privacy properties in prior sections.

3. NECESSARY CONDITIONS FOR CONTEXTUAL PRIVACY

We first characterize the choice rules that have a contextually private $\mathfrak{S}$-protocol, where $\mathfrak{S}$ is an arbitrary fixed class of admissible protocols.

Determining whether a social choice function is contextually private might be very challenging, since the space of protocols is potentially vast. In the following characterization, however, we see that we need not search over the full class of potential protocols in order to determine whether a choice rule has a contextually private protocol—we need only look at the behavior of the choice rule on subsets of the type space. As we will discuss further below, the necessary condition in this characterization is particularly useful. After presenting the characterization Theorem 1, we derive two implications.

THEOREM 1: *An $\mathfrak{S}$-computable choice rule ϕ is $\mathfrak{S}$-contextually private 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 $s_v \in \mathfrak{S}^*$ with $|s_v(\hat{\boldsymbol{\Theta}})| \geq 2$, there are type profiles $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i})$ and $\boldsymbol{\theta}' = (\theta_i', \boldsymbol{\theta}_{-i})$, $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \hat{\boldsymbol{\Theta}}$ with $s_v(\boldsymbol{\theta}) \neq s_v(\boldsymbol{\theta})$ and $\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}')$.*

PROOF: By Lemma 1, we may restrict our analysis to direct $\mathfrak{S}^*$-protocols. We first show that contextual privacy implies that no such set $\hat{\boldsymbol{\Theta}}$ exists. We prove this statement

[13]In Appendix D, we offer a brief discussion a related definition of equivalence of protocols.

in the contrapositive. Hence, assume that $\hat{\mathbf{\Theta}}$ exists. As $\phi|_{\hat{\mathbf{\Theta}}}$ is non-constant (by hypothesis (i)) and P is a protocol for ϕ, the set of nodes that distinguish two type profiles in $\hat{\mathbf{\Theta}}$ is non-empty. Let v be any earliest (i.e. minimal in precedence order) node that distinguishes two type profiles in $\hat{\mathbf{\Theta}}$. The query at node v, s_v, must have $|s_v(\hat{\mathbf{\Theta}})| \geq 2$ as it distinguishes types in $\hat{\mathbf{\Theta}}$. 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.

For the converse direction, assume that no such $\hat{\mathbf{\Theta}}$ exists. We construct a contextually private protocol inductively. For the base case, we consider $\mathbf{\Theta}_r = \mathbf{\Theta}$. For the inductive step, we consider an arbitrary node v associated with $\mathbf{\Theta}_v = \hat{\mathbf{\Theta}}$. Either, $\phi|_{\hat{\mathbf{\Theta}}}$ is constant, and the protocol can terminate and compute ϕ, or not. If $\phi|_{\hat{\mathbf{\Theta}}}$ is not constant then there must be a query s_v' such that $|s_v'(\hat{\mathbf{\Theta}})| \geq 2$ because, by assumption, there is an $\mathfrak{S}^*$-protocol for ϕ. Since hypothesis (i) holds, hypothesis (ii) cannot: there must be a query s_v such that $|s_v(\hat{\mathbf{\Theta}})| \geq 2$ and there are no types $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i})$ and $\boldsymbol{\theta}' = (\theta_i', \boldsymbol{\theta}_{-i})$, $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \hat{\mathbf{\Theta}}$ with $s_v(\boldsymbol{\theta}) \neq s_v(\boldsymbol{\theta}')$ and $\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}')$. Hence, the query s_v does not introduce any contextual privacy violations. The induction terminates, as $|\mathbf{\Theta}| < \infty$ and the cardinality of $\mathbf{\Theta}_v$ strictly decreases along paths on the tree P. *Q.E.D.*

This characterization reduces the search for contextual private protocols to a search over local properties of a social choice function. As the space of protocols is potentially vast (trees of depth up to $|\mathbf{\Theta}|$), this is a significant simplification. The proof is straightforward. We prove that contextual privacy implies that there is no "counterexample" that satisfies (i) and (ii) in the contrapositive—we directly show that (i) and (ii) produce a contextual privacy violation under our assumptions. The other direction proceeds inductively: we assume that there is no "counterexample" satisfying (i) and (ii) and then construct a contextually private protocol inductively.

To illustrate the value of the necessary condition of Theorem 1, we return to the sequential elicitation and count technologies. We first observe that $\mathfrak{S}^*_{\text{Count}} \supseteq \mathfrak{S}^*_{\text{SE}}$, hence all queries that are possible under sequential elicitation are also possible under count elicitation.

LEMMA 2: *The strategy-enhanced count technology* $\mathfrak{S}^*_{\text{Count}}$ *can simulate all queries that the strategy-enhanced sequential elicitation technology allows, i.e.* $\mathfrak{S}^*_{\text{Count}} \supseteq \mathfrak{S}^*_{\text{SE}}$.

PROOF: Let $\mathcal{S} \in \mathfrak{S}^*_{\text{SE}}$. We know that this $\mathcal{S}$ corresponds to a partition $\Theta_1 \cup \Theta_2 \cup \cdots \Theta_k = \Theta$, and an agent i. By the definition of strategy-enhanced technologies, it suffices to provide strategies $f_1, f_2, \ldots, f_n$ that partition type profiles as $\mathcal{S}$. This indeed holds for $f_j(\theta_j) = m_j$, $j \in N \setminus \{i\}$ and $f_i(\theta_i) = \tilde{\theta}_l$ for representative elements $\theta_l \in \Theta_l$. *Q.E.D.*

The idea of this result is that with a ballot box, it is possible to learn something about a particular agent, if all other agents send a fixed message.

3.1. *The Corners Lemma*

We show a necessary condition for the count elicitation technology, which, because of Lemma 2 also means an impossibility for sequential elicitation.

COROLLARY 1—Corners Lemma for Sequential Elicitation and Count Queries: *Let* ϕ *be contextually private under sequential elicitation or count elicitation protocols. Then, for any fixed* $\boldsymbol{\theta}_{-ij} \in \boldsymbol{\Theta}_{-ij}$*, for all types* $\theta_i, \theta_i', \theta_j, \theta_j' \in \Theta$,

$$\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}) = \phi(\theta_i', \theta_j, \boldsymbol{\theta}_{-ij}) = \phi(\theta_i, \theta_j', \boldsymbol{\theta}_{-ij}) = x \implies \phi(\theta_i', \theta_j', \boldsymbol{\theta}_{-ij}) = x. \quad (6)$$

PROOF: Assume that (6) does not hold. In particular,

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

but

$$\phi(\theta_i', \theta_j', \boldsymbol{\theta}_{-ij}) \neq x.$$

We apply Theorem 1 to the set $\hat{\boldsymbol{\Theta}} = \{\theta_i, \theta_i'\} \times \{\theta_j, \theta_j'\} \times \boldsymbol{\Theta}_{-ij}$. By assumption, $\phi|_{\hat{\boldsymbol{\Theta}}}$ is not constant. The only queries that separate this set of type profiles lead to the following three partitions:

$$\{(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}), (\theta_i', \theta_j, \boldsymbol{\theta}_{-ij})\} \cup \{(\theta_i, \theta_j', \boldsymbol{\theta}_{-ij}), (\theta_i', \theta_j', \boldsymbol{\theta}_{-ij})\}$$

$$\{(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}), (\theta_i, \theta'_j, \boldsymbol{\theta}_{-ij})\} \cup \{(\theta'_i, \theta_j, \boldsymbol{\theta}_{-ij}), (\theta'_i, \theta'_j, \boldsymbol{\theta}_{-ij})\}$$

$$\{(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij})\} \cup \{(\theta'_i, \theta'_j, \boldsymbol{\theta}_{-ij})\} \cup \{(\theta'_i, \theta_j, \boldsymbol{\theta}_{-ij}), (\theta_i, \theta'_j, \boldsymbol{\theta}_{-ij})\}.$$

The first query corresponds to a query that separates θ_i from θ'_i in the partition $(\Theta_{il})_{l=1,2,...,|\Theta|}$, but does not separate θ_j from θ'_j in $(\Theta_{jl})_{l=1,2,...,|\Theta|}$. The second query corresponds to a query that separates θ_j from θ'_j in the partition $(\Theta_{jl})_{l=1,2,...,|\Theta|}$, but does not separate θ_i from θ'_i in $(\Theta_{il})_{l=1,2,...,|\Theta|}$. The third query corresponds to a query that separates both θ_i from θ'_i in the partition $(\Theta_{il})_{l=1,2,...,|\Theta|}$ and θ_j from θ'_j in the partition $(\Theta_{il})_{l=1,2,...,|\Theta|}$.

All three of these partitions lead to contextual privacy violations if (6) does not hold. The first partition leads to a violation for $(\theta'_i, \theta_j, \boldsymbol{\theta}_{-ij}), (\theta'_i, \theta'_j, \boldsymbol{\theta}_{-ij})$, the second partition leads to a violation for $(\theta_i, \theta'_j, \boldsymbol{\theta}_{-ij}), (\theta'_i, \theta'_j, \boldsymbol{\theta}_{-ij})$ and the third partition leads to a violation for $(\theta_i, \theta'_j, \boldsymbol{\theta}_{-ij}), (\theta'_i, \theta'_j, \boldsymbol{\theta}_{-ij})$. *Q.E.D.*

We call this result the Corners Lemma because an analogue of it appears in the context of *unconditional privacy* (Chor and Kushilevitz, 1989, Chor et al., 1994) for decentralized computing—in Chor et al. (1994), the authors named the analogous result the Corners Lemma. A function of agents' private information can be computed in a way that preserves *unconditional privacy* if the agents can, through a decentralized messaging protocol, jointly compute the function through messages about their private information sent *to each other* without revealing more to anyone than is contained in the outcome. In their setting, they show a function cannot be computed with unconditional privacy if for any three "corners" of a two-by-two square that lead to one outcome, the fourth "corner" must also lead to that outcome.

It turns out that contextual privacy under sequential elicitation technologies is exactly analogous to the notion of unconditional privacy studied in Chor and Kushilevitz (1989). To see this, first note that under strategy-enhanced sequential elicitation technologies $\mathfrak{S}^*_{\mathrm{SE}}$, the designer queries one agent at a time, asking a question about their type. This corresponds to the assumption in the unconditional privacy setting that agents send messages only about their own information, one at a time. Next, to see the equivalence between the two concepts of privacy, note that in the decentralized setting, unconditional privacy is violated when some agent learns something that was not contained in the outcome, and in our centralized setting, all of this information would still be required, but would be concentrated in the hands of the designer.

So, the Corners Lemma also holds under sequential elicitation technologies $\mathfrak{S}^*_{\mathrm{SE}}$. We could have shown this fact using similar methods to those used in Chor and Kushilevitz (1989). Instead, we showed this through a novel observation: first we observed that any sequential elicitation protocol with queries from $\mathfrak{S}^*_{\mathrm{SE}}$ can be simulated through a series of count queries from $\mathfrak{S}^*_{\mathrm{Count}}$. Then we showed that the Corners Lemma holds for count elicitation technologies—thus our statement of the Corners Lemma is more general than the analogous statement in the decentralized computing literature initiated by Chor and Kushilevitz (1989).

In this subsection, we have shown that for two practically relevant technologies, there is a particularly simple "counterexample" $\phi_{\hat{\boldsymbol{\Theta}}}$ for contextual privacy. That is, if we find a "corner" under the count elicitation technology $\mathfrak{S}_{\mathrm{Count}}$ or sequential elicitation technology $\mathfrak{S}_{\mathrm{SE}}$, we know that the choice rule under consideration is not contextually private.

3.2. *Type Separability.*

Next, we use the characterization in Theorem 1 to derive a complete characterization of contextual privacy under the sequential elicitation technology $\mathfrak{S}^*_{\mathrm{SE}}$. This characterization will allow us to disprove the contextual privacy of a choice rule even if we do not find a counterexample of the shape implied by the Corners Lemma (and, for example, allow us to show that the failure of contextual privacy of the second-price auction for sequential elicitation does not rely on ties).

Instrumental to this analysis will be a notion of *type inseparability.*[14] Roughly, two types for an agent i are *inseparable* if the designer cannot distinguish between them without violating contextual privacy.

[14] 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 more than 2 agents.

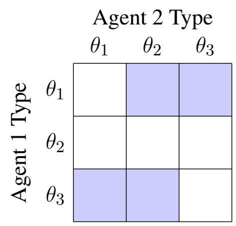

FIGURE 2.—Illustration of Inseparable Types with $n = 2, \hat{\mathbf{\Theta}} = \{\theta_1, \theta_2, \theta_3\}^2$. Shaded regions represent outcome x under ϕ. For agent 1, $\theta_3 \sim_{1,\phi,\hat{\mathbf{\Theta}}} \theta_1$. For agent 2, $\theta_1 \sim_{2,\phi,\hat{\mathbf{\Theta}}} \theta_2 \sim_{2,\phi,\hat{\mathbf{\Theta}}}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

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

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

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

We can view inseparability as a necessary condition for contextual privacy when ϕ is evaluated on a subset of type profiles $\hat{\mathbf{\Theta}}$. Assume that the designer arrives at an interior node v such that $\mathbf{\Theta}_v = \hat{\mathbf{\Theta}}$. 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.

To build further intuition for this definition, consider Figure 2 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 $\hat{\mathbf{\Theta}}$, 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 inseparability 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 $\boldsymbol{\Theta}_v = \hat{\boldsymbol{\Theta}}$. However, the fact that some product set $\hat{\boldsymbol{\Theta}}$ 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.

PROPOSITION 1—Characterization of Contextually Private Choice Functions Under Sequential Elicitation Protocols: *A choice function ϕ is contextually private if and only if there is no cylinder set $\hat{\boldsymbol{\Theta}}$ such that $\phi|_{\hat{\boldsymbol{\Theta}}}$ is non-constant and for all agents i and all $\theta_i, \theta_i' \in \boldsymbol{\Theta}_i'$, θ_i and θ_i' are inseparable.*

We use the characterization of contextual private rules under sequential elicitation technologies $\mathfrak{S}_{\mathrm{SE}}$ to prove impossibility results in cases where the Corners Lemma does not apply. In particular, we use this characterization to prove that the second-price auction choice rule is not contextually private under $\mathfrak{S}_{\mathrm{SE}}$ even when ties are forbidden in Appendix A.3 (there is also a proof of this statement that uses the Corners Lemma in subsection 4.2 which relies on the deterministic tie-breaking rule).

4. APPLICATIONS

We next turn to a practical application of the results in Section 3. We consider a handful of specific choice rules in three domains: assignment, auctions and voting. We show which choice rules are contextually private under sequential elicitation technologies $\mathfrak{S}^*_{\mathrm{SE}}$.[15] A summary of the results is collected in Table III. With one notable exception (the failure of the second-price auction even without ties), we prove all negative results in this section using the Corners Lemma.

[15]Note that, using the construction from Lemma 2, all results for $\mathfrak{S}^*_{\mathrm{SE}}$ imply the analogous result for the more powerful count elicitation technology $\mathfrak{S}^*_{\mathrm{Count}}$. However, we focus on sequential elicitation technologies as the protocols with count technologies are much less straightforward to interpret.

TABLE III

CONTEXTUAL PRIVACY OF CHOICE RULES UNDER SEQUENTIAL ELICITATION TECHNOLOGY

Choice Rule ϕ	Contextually Private Under $\mathfrak{S}_{\mathrm{SE}}$?	
	Assignment (Subsection 4.1)	
Serial Dictatorship Rule	✓	Appendix B
Efficient and IR Rules (House Assignment)	✗	Proposition 2
Stable Rules (Matching with Priorities)	✗	Proposition 3
	Auctions (Subsection 4.2)	
First-price Auction Rule	✓	Appendix C
Second-price Auction Rule	✗	Proposition 4
kth-price Auction Rule	✗	Proposition 4
Efficient Double Auction	✗	Proposition 5
	Voting (Subsection 4.3)	
Generalized Median Voting Rule	✗	Proposition 6

4.1. *Assignment.*

In the assignment domain, we fix a set C of objects. The set of outcomes is $X = 2^{N\times C}$, i.e. outcomes are matchings between agents in N and objects in 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 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.[16]

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

[16]In fact, this feature of serial dictatorships implies that they are not just contextually private but in fact satisfy the stronger properties of *individual* and *group* contextually privacy, as discussed in Section 7.

agents are initially endowed with an object from C. Denote the initial assignment by an injective function $e\colon N \to C$, where $e(i) \in C$ 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 2: *Assume agents have initial endowments $e(i)$. Then there is no individually rational, efficient and contextually private choice rule under sequential elicitation.*

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. $\boldsymbol{\theta}_{-ij} = (e(k) \succeq_k c \text{ for all } c \in C \setminus \{e(k)\})_{k \in N \setminus \{i,j\}}$.

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 3: $\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-i,-j}) = (\theta_i, \theta_j', \boldsymbol{\theta}_{-i,-j}) = (\theta_i', \theta_j, \boldsymbol{\theta}_{-i,-j}) = x$. Meanwhile, efficiency requires $\phi(\theta_i, \theta_j, \boldsymbol{\theta}_{-i,-j}) = y$ (shown on the mid-right in Figure 3). 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.

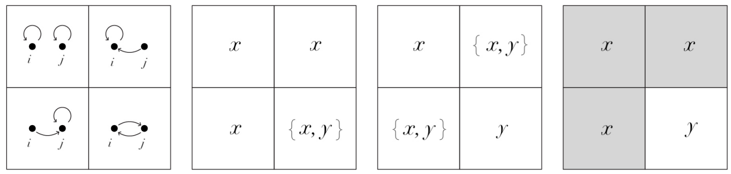

FIGURE 3.—Applying the Corners Lemma for house assignment. Type profiles $\{\theta_i, \theta_i'\} \times \{\theta_j, \theta_j'\}$ used in the proof (left, arrows denote whether agent prefers own or other's endowed object); required outcomes for each type profile under individual rationality (mid-left), under efficiency (mid-right), and under both efficiency and individual rationality (right).

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 (also called "student") is matched to at most one object (also called "school"), and at most $\kappa(c)$ agents are matched to an object c, for every $c \in C$, for some *capacities* $\kappa(c)$. That is, the set of outcomes is

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

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

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 has a vector of *scores* s_c, representing their score at each object $c \in C$. 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 $s_i\colon C \to \mathbb{R}$ 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), $i \in N$, $c \in C$ such that $c \succ_i \phi_i(\boldsymbol{\theta})$ and $s_i(c) > s_i(\phi_i(\boldsymbol{\theta}))$.

PROPOSITION 3: *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. Types are real numbers $\Theta \subseteq [0, 1]$. 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

$$u_i((q, t); \theta_i) = \theta_i q_i - t_i, \tag{7}$$

We call an auction choice rule *standard* if there is at most one agent $i \in N$ such that $t_i \neq 0$ and $q_i = 1$. We call an auction *efficient* if

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

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 lexicographic tiebreaking. The *first-price auction* is a choice rule

$\phi^{\mathrm{FP}}(\boldsymbol{\theta}) = (\phi_1^{\mathrm{FP}}(\boldsymbol{\theta}), \ldots, \phi_n^{\mathrm{FP}}(\boldsymbol{\theta})) = ((q_1, t_1), \ldots, (q_n, t_n))(\boldsymbol{\theta})$, where

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

for some bid shading function $b_i \colon \boldsymbol{\theta}_i \to \boldsymbol{\theta}$. The *second-price auction* is a choice rule $\phi^{\mathrm{SP}}(\boldsymbol{\theta})$, where

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

where b is the bidding function. 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 F.

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 for three or more bidders. 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 4: *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 $\{\underline{\theta}, \bar{\theta}\} \times \{\underline{\theta}, \bar{\theta}\} \times \bigtimes_{k \in N \setminus \{i,j\}} \theta_k$. This corresponds to a square depicted in Figure 4.

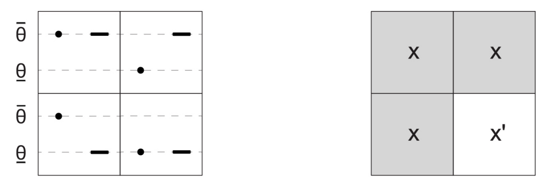

FIGURE 4.—Applying the Corners Lemma to the second-price auction. Type profiles for agent i and agent j (left, agent j's type is represented by a dot, and agent i's type is represented by a dash); required outcome under the second-price auction rule ϕ^{SP}.

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 $t_i = \bar{\theta}$. Let x' be the outcome under which the highest type wins and pays the price $t_i = \underline{\theta}$. Then, $\phi^{\text{SP}}(\bar{\theta}, \bar{\theta}, \boldsymbol{\theta}_{-i,-j}) = \phi^{\text{SP}}(\underline{\theta}, \bar{\theta}, \boldsymbol{\theta}_{-i,-j}) = \phi^{\text{SP}}(\bar{\theta}, \underline{\theta}, \boldsymbol{\theta}_{-i,-j}) = x$. But, under ϕ^{SP}, it must be the case that $\phi(\underline{\theta}, \underline{\theta}, \boldsymbol{\theta}_{-i,-j}) = x'$. 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 $\underline{\theta}, \bar{\theta} \in (\theta_{[k-1]}, \theta_{[k+2]})$. *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

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

A double auction price rule seeks to find a price t that maximizes $\sum_{i \in N} u_i((q,t),\theta_i)$ 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 5: *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 and Information Aggregation.*

We next turn to a voting environment. There is an ordered set of social outcomes X. Preferences are single-peaked $\prec_i$ with peaks $\theta_i \in X$.[17]

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

[17]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 $x > \theta > \theta'$.

takes as input submitted peaks of agents' preferences $\theta_1, \theta_2, \dots \theta_n$ as well as *phantom ballots* $k_1, k_2, \dots, k_{n-1} \in X \cup \{-\infty, \infty\}$. The output is the median of the submitted votes and phantom votes, i.e.

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

PROPOSITION 6: *For any $k_1, k_2, \dots, k_{n-1} \in X \cup \{-\infty, \infty\}$ such that neither $\sup(k_1, k_2, \dots, k_{n-1}) \neq -\infty$ nor $\inf(k_1, k_2, \dots, k_{n-1}) \neq \infty$, the generalized median voting rule $\phi_{(k_1,k_2,\dots,k_{n-1})}$ is not contextually private under sequential elicitation.*

The case where $\sup(k_1, k_2, \dots, k_{n-1}) = -\infty$ and $\inf(k_1, k_2, \dots, k_{n-1}) = \infty$ makes the choice rule $\inf(\theta_1, \theta_2, \dots, \theta_n)$ resp. $\sup(\theta_1, \theta_2, \dots, \theta_n)$. In this case, there is an ascending protocol that is contextually private for the choice rule $\inf(\theta_1, \theta_2, \dots, \theta_n)$ and adescending protocol that is contextually private for $\sup(\theta_1, \theta_2, \dots, \theta_n)$. For all other cases, we use the Corners Lemma. Note that this result connects closely to Proposition 5—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. Observe that this result also leads to an impossibility of computing an interior quantile of a set of values.

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

When contextual privacy is unattainable, how should designers design with contextual privacy in mind? In this section, we compare the contextual privacy of different protocols, using the notions of *equivalence* and *improvement* defined in Section 2.

This section has two parts. The first part centers on a key theoretical insight of the paper—a representation theorem which helps us to reduce the practical complexity of designing for privacy (Theorem 2). In particular, we show that for choice rules of a particular structure, every protocol is contextual privacy equivalent to what we call a *bimonotonic* protocol. This statement is practically relevant because it can help designers to simplify their consideration of different protocols: it is without loss for designers to consider only bimonotonic protocols when designing for contextual privacy, as every possible protocol is contextual privacy equivalent to a bimonotonic one.

The second part of this section aims to lend insight into how a designer might choose among different protocols that are contextual privacy equivalent. We consider which protocols in a particular equivalence class can be *improved*: improvements take into account the type profiles at which contextual privacy is violated. Here, it is difficult to make statements with a high degree of generality, given that the space of protocols for a given choice rule is vast. So, we focus on a particular choice rule, the second-price choice rule, both because it is theoretically illustrative and because it is practically relevant. We show that for the second-price choice rule, the commonly used ascending protocol has a *strict improvement* which we call the *ascending-join protocol*. We then briefly discuss how the principles behind the ascending-join protocol may extend to other settings.

5.1. *A Representation Theorem*

The main goal of this subsection is to present our representation theorem (Theorem 2) for contextual privacy equivalence classes under the restriction to sequential elicitation protocols. The theorem we prove shows that a privacy-concerned designer can, without loss, restrict attention to *bimonotonic* protocols under the restriction to sequential elicitation protocols $\mathfrak{S}_{\mathrm{SE}}$.

First, we formally define biomonotonicity. A bimonotonic protocol draws all of its queries from a subset of the sequential elicitation technology that we call the *threshold elicitation technology*. We denote the set of threshold elicitation queries to be

$$\mathfrak{S}_{\text{thresh}} = \{\{\theta_i \leq \tilde{\theta}\} \mid \tilde{\theta}_i \in \Theta\} \cup \{\{\theta_i > \tilde{\theta}\} \mid \tilde{\theta}_i \in \Theta\}.$$

That is, a threshold query is a query to a single agent, asking whether their type falls above or below some value $\tilde{\theta}_i \in \Theta$. A bimontonic protocol is a protocol in which all the threshold queries to a single agent i form an increasing or decreasing interval in the type space Θ.

DEFINITION—Bimonotonic Protocol: A $\mathfrak{S}_{\text{thresh}}$-protocol P is *bimonotonic* if for any path in P, the sequence of thresholds to agent i form an increasing or decreasing interval in Θ.

The main theorem will show that for many choice rules of interest, $\mathfrak{S}_{\text{SE}}$-protocol is contextual privacy equivalent to a bimonotonic protocol. The choice rules of interest must have a particular property which we call *interval pivotality*.

DEFINITION—Interval Pivotality: A social choice function ϕ defined on ordered type space Θ 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 profile of other agents' types $\boldsymbol{\theta}_{-i}$, the choice rule $\phi(\theta_i, \boldsymbol{\theta}_{-i})$ 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 social choice rules of interest are interval pivotal: Any kth 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 are interval pivotal. Note, however, that interval pivotality only applies to choice rules defined on ordered type spaces, so assignment and matching rules, such as those considered Subsection 4.1, cannot be interval pivotal.

Now that we have defined bimonotonicity and interval pivotality, we can present the main result of this section, and indeed a central theoretical insight of the paper. This representation theorem simplifies the designer's search for privacy-preserving protocols when choosing among sequential elicitation protocols for an interval pivotal choice rule: the designer need only consider bimonotonic protocols, because every $\mathfrak{S}_{\mathrm{SE}}$-protocol is contextual privacy equivalent to a bimonotonic protocol.

THEOREM 2: *Let P be a $\mathfrak{S}_{\mathrm{SE}}$-protocol for ϕ, where ϕ exhibits interval pivotality. There is a bimonotonic protocol P' that is contextual privacy equivalent to P.*

In other words, every contextual privacy equivalence class has a bimonotonic representative.

The proof of Theorem 2 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 holes between two separated agent types in queries to the same agent. A "deduplication" operation keeps only a monotonic sequence of threshold queries.

Several common protocols are bimonotonic, and we define them formally in Appendix F: 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. 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, et cetera.

The fact that every equivalence class has a bimonotonic representative emphasizes that the decisions a designer makes about privacy amount to: 1) the threshold of the initial query to each agent, and 2) the order in which the designer asks agents these threshold queries.

5.2. *The Case of the Second-Price Auction*

To investigate the implications of the bimonotonicity result (Theorem 2), we focus on a particular choice rule, the second-price auciton rule. We first consider in Subsection 5.2.1 three different contextual privacy equivalence classes for the second-price auction rule. Their bimonotonic representatives—the ascending, overdescending and guessing protocols—highlight tradeoffs between the privacy of different agents and at different type profiles. Next, in Subsection 5.2.2, we consider an *improvement* in one of these equivalence classes: we show that the ascending protocol can be contextual privacy *improved* by using instead what we call an *ascending-join* protocol.

5.2.1. *Three Equivalence Classes.*

In this subsection we consider three contextual privacy equivalence classes, and their bimonotonic representatives. We begin with the *ascending* and *overdescending* protocols which are formally defined in Appendix F. We focus on the ascending protocol because it is widely used in practice, and widely studied, and we focus on the overdescending protocol because it resembles the descending ("Dutch") protocol for first-price rules, which we showed to be contextually private under sequential elicitation in Appendix C.

EXAMPLE—Ascending and Overdescending Protocols: While the ascending protocol for the second-price choice rule is standard (corresponding to open "English" auctions), 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.

Consider a type profile $(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij})$ such that $\theta_i = \max \Theta$ and $\theta_j = \min \Theta$ and the second highest type $\theta_{[2]}$ is not equal to the highest type, i.e. $\theta_{[2]} \neq \theta_i$. Then, on the one hand, the ascending protocol has a contextual privacy violation for agent j and type profiles $(\theta_i, \mathrm{succ}(\theta_j), \boldsymbol{\theta}_{-ij})$, $(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij})$, which the overdescending protocol does not have. On the other hand, the overdescending protocol has a contextual privacy violation for agent i and type profiles $(\mathrm{pred}(\theta_i), \theta_j, \boldsymbol{\theta}_{-ij})$, $(\theta_i, \theta_j, \boldsymbol{\theta}_{-ij})$, which the ascending protocol does not have.

In general, when $|n| > 2$, the ascending protocol violates the privacy of the $n-2$ losing bidders, and the overdescending protocol violates the privacy of the single winning bidder. These two protocols thus illustrate that the contextual privacy order helps us to sharpen our understanding of the tradeoffs between violating the privacy of different (groups of) agents. As described in Milgrom and Segal (2020), in the FCC's 2017 spectrum auction, it was important that the auction design protected the privacy of the winning bidder: so, the auction design used there resembles the ascending protocol. However, in other cases it may be more important to protect the privacy of the losing bidders. For example, in the case of Google's second-price auctions for advertising, where a 2022 lawsuit alleged that the company was storing advertisers' losing bids from prior auctions to set personalized reserve prices in future auctions, it may be more important to protect the privacy of losing bidders, and thus an overdescending auction may be the better design (Texas v. Google, 2022).

Different equivalence classes not only trade off the privacy of different agents, but also trade off the agents whose privacy is violated at different type profiles. To see this, we turn to the next example, which considers a *guessing protocol*.

EXAMPLE—Guessing Protocols: The *guessing protocol* for the second-price choice rule essentially guesses a price $\tilde{\theta} \in \Theta$ and then aims to verify that it is the second-highest bid. The protocol starts with a query $\theta_i > \tilde{\theta}$ for all agents.

Consider first type profiles in which a single agent is higher than the initial query $\tilde{\theta}$. Then, no ascending questions are asked anymore. If all agents have types at most $\tilde{\theta}$, the protocol descends for other agents, asking all agents except the agent with the highest bid $\theta_i > \operatorname{pred} \tilde{\theta}$, and then $\theta_i > \operatorname{pred}(\operatorname{pred} \tilde{\theta})$, and so on until the second-highest bid is found.

For the type profile in which the second-highest bid is exactly $\tilde{\theta}$ and the highest bid is larger than $\tilde{\theta}$, that is, the guess is correct, there are no contextual privacy violations. Denote this type profile $\boldsymbol{\theta}$, the terminal node following it by z, the winner of the auction by i and the second-highest bidder by j. Then,

$$\boldsymbol{\Theta}_z = \{\boldsymbol{\theta} \in \boldsymbol{\Theta} : \theta_i > \tilde{\theta},\, \theta_j = \tilde{\theta},\, \theta_k < \tilde{\theta}\ \forall k \in N \setminus \{i, j\}\}.$$

That is, the guessing auction learns the second-highest bidder's type exactly, and 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. For example, if all types are smaller than $\tilde{\theta}$, the guessing protocol will be an overdescending protocol on the restricted type spaces $\{\theta \in \Theta \mid \theta \leq \tilde{\theta}\}$, which (like the overdescending protocol) will violate the privacy of the winner. At such type profiles, the set of violations are the same as in the overdescending protocol. But, at other type profiles, the violations are the same as in the ascending protocol: if all types are above p, the protocol amounts to an ascending protocol on the restricted type space $\{\theta \in \Theta \mid \theta \geq \tilde{\theta}\}$, violating the privacy of all the losers.

So, the guessing protocol example shows that there are some type profiles at which the guessing protocol has fewer violations than the overdescending or ascending protocols, and other type profiles at which the guessing protocol produces the same violations as either the ascending or overdescending protocols. When considering all type profiles, the guessing protocol produces some violation for every agent, but when considering only one type profile, the guessing protocol can produce no violations for any agent.

The guessing protocol suggests an interesting direction for future work: If the designer cares about the *expected* contextual privacy violations, then the guessing protocol may be preferable to the ascending or overdescending protocols. In other words, if the designer uses their prior $f(\theta)$ to choose a "good guess", and if the designer cares about privacy violations *in expectation*, then the guessing protocol might be preferable, from a privacy standpoint, to the ascending or overdescending protocols, because the designer can leverage their prior to minimize contextual privacy violations in expectation. But, if the designer can't use their prior to choose a "good guess", because e.g. the prior is diffuse, then the guessing protocol may produce more contextual privacy violations in expectation than the ascending or overdescending protocols.

5.2.2. *Comparisons Within Equivalence Classes.*

We have now considered three contextual privacy equivalence classes for the second-price auction choice rule. Next, we discuss how a designer might select a particular protocol within an equivalence class. In other words, equivalence classes are large, is there a way for a designer to select among contextual privacy equivalent protocols? We focus on the ascending protocol for the second-price choice rule and make one simple observation: there is an *improvement* to the ascending protocol, which we call the *ascending-join* protocol.

We make the observation in this section for two reasons. First, it is design-relevant in itself, as it speaks to a commonly-used protocol (the ascending auction) for a commonly-used choice rule (the second-price choice rule). Second, we make this observation because it is illustrative more broadly of how "ascending"-type protocols for choice rules with interval pivotality might be improved upon.

We introduce the ascending-join protocol through an example. Roughly speaking, ascending-join protocols work as follows. The protocol begins at some threshold "price" $\tilde{\theta} \in \Theta$. Two agents are selected as "active" agents. The designer asks these two agents whether their type is above $\tilde{\theta}$. If the answer from both agents is "yes", then the designer raises the threshold to $\operatorname{succ} \tilde{\theta} \in \Theta$ and asks the same two active agents again whether their type is above the new threshold value $\operatorname{succ} \tilde{\theta}$. If one of the agents answers "no", then that agent exits the set of active agents and the designer chooses a different agent who has never been active to *join* the set of active agents. The agent who joins the active set is asked whether their type is above $\tilde{\theta}$. At all points, the designer maintains a set of exactly two active agents, and agents cannot be in the active set more than once. When only one agent answers a question of the form "Is your type above x?" in the affirmative, the protocol stops.

EXAMPLE—Ascending-Join Protocol: Consider a setting with $N = 5$, $\Theta = \{1, 2, 3, 4, 5, 6\}$ and a true type profile $\boldsymbol{\theta} = (4, 3, 1, 1, 5)$. The second-price auction choice rule gives the object to agent 5 at a price of 3.

The ascending auction produces violations for agents 2, 3 and 4 because the designer learns their types exactly when the designer only needed to know that their types were below 4. In other words, and with a slight abuse of notation, the designer learns

$$\boldsymbol{\theta} \in \{4\} \times \{3\} \times \{1\} \times \{1\} \times \{5, 6\} \tag{8}$$

Now consider a protocol, which we call an ascending-join protocol, that runs as follows:

- Ask agents 1 and 2 "Is your type above 1?"
  $\rightarrow$ Both answer "yes."
- Ask agents 1 and 2 "Is your type above 2?"
  $\rightarrow$ Both answer "yes."
- Ask agents 1 and 2 "Is your type above 3?"
  $\rightarrow$ Agent 1 answers "yes" and agent 2 answers "no."
- Since agent 2 answered no, agent 3 *joins* the set of active agents. Ask agent 3 "Is your type above 3?"
  $\rightarrow$ Agent 3 answers "no."
- Since agent 3 answered no, agent 4 *joins* the set of active agents. Ask agent 4 "Is your type above 3?"
  $\rightarrow$ Agent 4 answers "no."
- Since agent 4 answered no, agent 5 *joins* the set of active agents. Ask agent 5 "Is your type above 3?"
  $\rightarrow$ Agent 5 answers "yes".
- Since there are now two active agents with types above going threshold 3, increase threshold to 4. Ask agents 1 and 5 "Is your type above 4?"
  $\rightarrow$ Agent 1 answers "no" and agent 5 answers "yes."
- The protocol concludes: allocate object to agent 5, set price at 4.

Through this ascending-join protocol, the designer learns that the type profile lies in the set (again with a slight abuse of notation)

$$\boldsymbol{\theta} \in \{4\} \times \{3\} \times \{1,2\} \times \{1,2\} \times \{5,6\}. \tag{9}$$

Compare what is learned by the ascending auction (8) to what is learned in the ascending-join auction (9). We see that, although these two auctions are contextual privacy equivalent, in that they both produce violations for agents 2, 3 and 4, the violations in the ascending-join auction are not as extensive for agents 3 and 4. In particular, in the ascending-join protocol the designer learns that agents 3 and 4 have types strictly below 3, whereas in the ascending protocol the designer learns these agents' types exactly.

We define ascending-join protocols formally in Appendix F. The next result shows that the ascending-join protocol is a contextual privacy improvement on the ascending protocol for the second-price choice rule.

PROPOSITION 7: *Assume* $|N| \geq 2$*. Then,* $P_{\text{Asc-Join}} \sim_\phi P_{\text{Asc}}$*, but* $P_{\text{Asc-Join}}$ *is a strict* ϕ_{SPA}*-improvement.*

While this result is for a very specific choice rule, the second-price auction rule, it also offers insight for other protocols. For example, consider an interval-pivotal rule that requires computing any order statistic of the type profile, such as the introductory example in Section 1. The rule that asks agents from the bottom of the type profile is bimonotonic, and can be improved by having only two "active" agents at any given moment, where the "active" agents set the threshold "price".

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. In this section, we provide an initial study of the conjunction of contextual privacy and implementation, i.e. whether there are contextually private protocols that are also implementable. While it is beyond the scope of this article to treat incentives in depth, we take a step in this direction by providing positive results on the dynamic incentives of a few sequential elicitation protocols highlighted in Section 4 and Section 5. [18]

We begin this section by formally defining two dynamic implementation notions in subsection 6.1: implementation in obviously dominant strategies and in perfect Bayesian strategies. Then, we turn attention to three protocols that we singled out as having good contextual privacy properties in Section 4 and Section 5. We first show that the ascending-join protocol for the second-price auction rule, which we showed to be an improvement on the ascending protocol in Section 5, is implementable in obviously dominant strategies. Then we show that the serial dictatorship, which we showed to be contextually private in

[18]Given Lemma 2, one may also consider the incentives of protocols under the count technology. The incentives of such protocols do depend on what the principal does in case of deviations, as under count queries they cannot necessarily detect a deviator.

Section 4, is also implementable in obviously dominant strategies. Finally, we show that the descending protocol for the first-price auction rule, which we showed to be contextually private in Section 4, is implementable in perfect Bayesian equilibrium strategies.

6.1. *Implementation in Obviously Dominant and Perfect Bayesian Strategies.*

Recall that agents submit messages from a set M according to strategies $\sigma_i \colon \Theta \times V \to M$. The state of the protocol is *public*, so that their strategy takes into account information known at node $v \in V$, i.e. $\boldsymbol{\Theta}_v$. We will assume that if an off-path outcome is reached, the designer does not complete the allocation implied by the choice rule, and all agent utilities are some fixed outside option (e.g. 0 in an auction environment).

Implementation in obviously dominant strategies requires that the worst outcome following the messaging strategy is better than the best outcome from a deviation.

DEFINITION: We say that a protocol P for ϕ *implements* ϕ in *obviously dominant strategies* $\boldsymbol{\sigma}$ if for all agents i, nodes v and $\boldsymbol{\theta}_{-i} \in \boldsymbol{\Theta}_{-i}$, we have

$$\inf_{\substack{\boldsymbol{\theta} := (\theta_i, \boldsymbol{\theta}_{-i}) \in \boldsymbol{\Theta}_w \\ w = s_v(\boldsymbol{\sigma}(\boldsymbol{\theta}))}} u_i\left(\phi(\boldsymbol{\theta}); \theta_i\right) \geq \sup_{\substack{\boldsymbol{\theta}' := (\theta_i', \boldsymbol{\theta}_{-i}) \in \boldsymbol{\Theta}_w \\ w \in \text{children}(v) \setminus s_v(\boldsymbol{\sigma}(\boldsymbol{\theta}))}} u_i\left(\phi(\boldsymbol{\theta}'); \theta_i\right).$$

Consider this in the case of a direct protocol. Then, 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 some other type θ_i', regardless of what other agents do in the future. Notice that when the agent tells the truth, the child node of v is $w = s_v(\boldsymbol{\sigma}(\theta_i, \boldsymbol{\theta}_{-i}))$. When the agent does not tell the truth, the child node of v is some $w \in \text{children}(v) \setminus s_v(\boldsymbol{\sigma}(\theta_i, \boldsymbol{\theta}_{-i}))$.

We next define a notion of implementation in perfect Bayesian equilibrium. For this, 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*. This definition relies strongly on the assumption that information is public, i.e. at each node v, all agents know $\boldsymbol{\Theta}_v$.

DEFINITION: We say that a protocol P for ϕ with strategies $\boldsymbol{\sigma}$ *implements* ϕ in *perfect Bayesian equilibrium strategies* if for all agents $i \in N$, and nodes $v \in V$

$$\mathbb{E}[u_i(\phi(\boldsymbol{\theta}); \theta_i) \mid \boldsymbol{\theta} \in \boldsymbol{\Theta}_{s_v(\boldsymbol{\theta})}] \geq \mathbb{E}[u_i(\phi(\boldsymbol{\theta}); \theta_i) \mid \boldsymbol{\theta} \in \boldsymbol{\Theta}_w]$$

for all $w \in \text{children}(v)$.

If P implements ϕ with direct strategies $\boldsymbol{\sigma}$, that is, P is a direct protocol for ϕ, we also say that the direct protocol P implements ϕ. For both notions of implementation, the reduction in Lemma 1 applies.

COROLLARY 2: *There is a $\mathfrak{S}$-protocol P implementing ϕ in obviously dominant resp. perfect Bayesian strategies and contextual privacy violations $\Gamma \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$ if and only if there is a direct $\mathfrak{S}^*$-protocol P^* implementing ϕ with contextual privacy violations $\Gamma \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$.*

6.2. *Implementation Results for Three Contextual Privacy-Preserving Protocols.*

There are three protocols that we showed to have particularly good properties from the standpoint of contextual privacy. We begin with the ascending-join protocol for the second-price auction rule, which is a contextual privacy improvement on the ascending protocol. These protocols, though we have not seen them defined explicitly as we have, have equilibria in obviously dominant strategies.

PROPOSITION 8: *The ascending-join protocols for ϕ^{SP} are a contextual privacy improvement on ascending protocols, and implement ϕ^{SP} in obviously dominant strategies.*

To show that the second part of this statement holds, we show that the ascending-join 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 every personal-clock auction protocol has an equilibrium in obviously dominant strategies.

We continue to rely on results from Li (2017) to show that the serial dictatorship, which we showed to be contextually private in Section 4, is also implementable in obviously dominant strategies.

PROPOSITION 9: *The serial dictatorship choice rule is $\mathfrak{S}_{\mathrm{SE}}$-contextually private and is implementable in obviously dominant strategies.*

Again, we show that the serial dictatorship amounts to a personal-clock auction.

Next we turn to the descending protocol for ϕ^{FP}, which we showed to be contextually private in Section 4. It is well known that this protocol has good dynamic incentives, namely that it is strategically equivalent to a static sealed-bid first price auction which has a symmetric Bayes-Nash equilibrium.

PROPOSITION 10: *The descending protocol for ϕ^{FP} is $\mathfrak{S}_{\mathrm{SE}}$-contextually private and implementable in Perfect Bayesian equilibrium strategies.*

We view further study of incentives and contextual privacy as a key direction for future work. The intention of this section was primarily to show that our formalism is capacious enough to discuss incentives and to formally define implementation notions, and to begin to show how our results connect to the literature on dynamic implementation.

7. VARIATIONS

In this section, we consider two concepts that strengthen the notion of contextual privacy violation. 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.

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 X, specifies an allocation for each agent $i \in N$.[19] 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 with strategies $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ has an *individual contextual privacy violation* at $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i}), \boldsymbol{\theta}' = (\theta_i', \boldsymbol{\theta}_{-i})$ if $\boldsymbol{\theta}, \boldsymbol{\theta}'$ are distinguished under P and $\boldsymbol{\sigma}$ but lead to the same outcome for agent i. If a protocol does not have any individual contextual privacy violations, we call it individually contextually private. If there is a protocol P (with some strategies $\boldsymbol{\sigma}$) for ϕ that is individually contextually private, we say that ϕ is individually contextually private.

As in the main text, and with an identical proof, we may reduce to truthtelling protocols.

LEMMA 3: *For every $\mathfrak{S}$-protocol P, there is a direct $\mathfrak{S}^*$-protocol P^* with the same individual contextual privacy violations.*

Notice that individual contextual privacy is stronger than contextual privacy—any choice rule that is individually contextually private is also 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."

[19]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.

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."[20] 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 $\mathfrak{S}$-individually contextually private if and only if it is $\mathfrak{S}$-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 $(\theta_i', \boldsymbol{\theta}_{-i})$ 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 individually contextually private 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 individually contextually private. *Q.E.D.*

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

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

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 individually 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 the characterization of efficient, non-bossy and individually rational auctions 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 with strategies $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ is *group contextually private* if for all type profiles $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \boldsymbol{\Theta}$ that are distinguished by P and $\boldsymbol{\sigma}$, the type profiles $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ lead to different outcomes under $\boldsymbol{\sigma}$. If a protocol does not have any group-contextual privacy violations, we call it individually contextually private. If there is a protcol P (for some strategies $\boldsymbol{\sigma}$) for ϕ that is individually contextually private, we say that ϕ is group-contextually private.

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 $\boldsymbol{\Theta}_{-i}$.

As in the main text, and with an identical proof, we may reduce to truthtelling protocols.

LEMMA 4: *For every $\mathfrak{S}$-protocol P for ϕ, there is a direct $\mathcal{S}^*$-protocol P^* for ϕ with the same group-contextual privacy violations.*

We hence may restrict to direct protocols.

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 $X_v \coloneqq \phi(I_v)$.

THEOREM 3: *A protocol $P = (V, E)$ is $\mathfrak{S}$-group contextually private if and only if for any query, $\bigcup_{(v,w)\in E} X_w = X_v$ 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 $X_w \cap X_{w'} \neq \emptyset$. Hence, there are $\boldsymbol{\theta} \in \boldsymbol{\Theta}_w$ and $\boldsymbol{\theta}' \in \boldsymbol{\Theta}_{w'}$ such that $\phi(\boldsymbol{\theta}) = \phi(\boldsymbol{\theta}')$, 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 3: *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 E 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. We discuss here our connection to extensive-form mechanism design, as well as other literature on designing for privacy in computer science and cryptography.

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.[21] Under sequential elicitation protocols, contextual privacy can be understood as a form of privacy that is

[21]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.

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*.[22]

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., 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. The construction allows for particular elicitation technologies that allow, e.g., envelopes-

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

inside-envelopes. This paper does not consider such technologies. 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.[23] 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 without relying on any trusted third party. To this literature we bring analysis of the impact

[23] In addition, we strengthen the impossibility result Brandt and Sandholm (2008, Theorem 4.9) paralleling our Proposition 4 to cases where no ties are allowed (see the proof in subsection A.3).

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.[24] 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.[25]

9. CONCLUSION

This paper introduced a notion of *contextual privacy violations* into mechanism design. An agent's contextual privacy is violated when the designer learns more about her private information than is needed in context, i.e. more than is needed to compute the outcome of the choice rule. Whether the designer violates agents' privacy depends not only on the structure of the choice rule, but on details of the social and technological environment, i.e. whether the designer's has access to mediating technologies that anonymize or shuffle the reports of participants.

[24]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).

[25]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).

There are two directions for future work that may be especially fruitful: (i) contextual privacy and incentives, and (ii) statistical contextual privacy violations. First, although we discuss traditional implementation concerns briefly in Section 6, we do not offer a full treatment of designing for contextual privacy in conjunction with designing for good incentives. It would be valuable to better understand how contextual privacy interacts with incentives for truthtelling more generally. Second, the notion of contextual privacy violation studied here focused only on *whether* a violation occurred, as opposed to *how likely* it is that a given violation might occur. As discussed briefly in the context of the "guessing protocol" for the second-price auction rule in Subsection 5.2.2, understanding contextual privacy violations in expectation might lead to further design-relevant comparisons among protocols. When shifting to a statistical perspective that takes an expectation over possible contextual privacy violations, the best protocol, from a contextual privacy standpoint, may depend on the designer's prior beliefs about the type profile.

BIBLIOGRAPHY

ACQUISTI, ALESSANDRO, CURTIS TAYLOR, AND LIAD WAGMAN (2016): "The economics of privacy," *Journal of economic Literature*, 54, 442–492. [2, 51]

AKBARPOUR, MOHAMMAD AND SHENGWU LI (2020): "Credible auctions: A trilemma," *Econometrica*, 88, 425–467. [4, 30, 50]

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

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. [51]

AUSUBEL, LAWRENCE M (2004): "An efficient ascending-bid auction for multiple objects," *American Economic Review*, 94, 1452–1475. [2]

BADE, SOPHIE AND YANNAI A GONCZAROWSKI (2016): "Gibbard-Satterthwaite success stories and obvious strategyproofness," *arXiv preprint arXiv:1610.04873*. [51]

BALINSKI, MICHEL AND TAYFUN SÖNMEZ (1999): "A tale of two mechanisms: student placement," *Journal of Economic theory*, 84, 73–94. [28]

BENTHALL, SEBASTIAN, SEDA GÜRSES, HELEN NISSENBAUM, ET AL. (2017): *Contextual integrity through the lens of computer science*, Now Publishers. [53]

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. [2, 30, 52]

——— (2008): "On the existence of unconditionally privacy-preserving auction protocols," *ACM Transactions on Information and System Security (TISSEC)*, 11, 1–21. [29, 52, 59]

CANETTI, RAN, AMOS FIAT, AND YANNAI A GONCZAROWSKI (2023): "Zero-Knowledge Mechanisms," *arXiv preprint arXiv:2302.05590*. [51]

CHOR, BENNY, MIHÁLY GERÉB-GRAUS, AND EYAL KUSHILEVITZ (1994): "On the structure of the privacy hierarchy," *Journal of Cryptology*, 7, 53–60. [22, 23]

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. [2, 22, 23]

DWORCZAK, PIOTR (2020): "Mechanism design with aftermarkets: Cutoff mechanisms," *Econometrica*, 88, 2629–2661. [51]

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. [53]

EILAT, RAN, KFIR ELIAZ, AND XIAOSHENG MU (2021): "Bayesian privacy," *Theoretical Economics*, 16, 1557–1603. [51]

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. [52]

GHOSH, ARPITA AND AARON ROTH (2015): "Selling privacy at auction," *Games and Economic Behavior*, 91, 334–346. [53]

GOLOWICH, LOUIS AND SHENGWU LI (2021): "On the computational properties of obviously strategy-proof mechanisms," *arXiv preprint arXiv:2101.05149*. [51]

GRIGORYAN, ARAM AND MARKUS MÖLLER (2023): "A theory of auditability for allocation and social choice mechanisms," *arXiv preprint arXiv:2305.09314*. [51]

GUL, FARUK AND ANDREW POSTLEWAITE (1992): "Asymptotic efficiency in large exchange economies with asymmetric information," *Econometrica: Journal of the Econometric Society*, 1273–1292. [53]

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

HATFIELD, JOHN WILLIAM (2009): "Strategy-proof, efficient, and nonbossy quota allocations," *Social Choice and Welfare*, 33, 505–515. [48]

IZMALKOV, SERGEI, MATT LEPINSKI, AND SILVIO MICALI (2011): "Perfect implementation," *Games and Economic Behavior*, 71, 121–140. [51]

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. [51]

LI, SHENGWU (2017): "Obviously strategy-proof mechanisms," *American Economic Review*, 107, 3257–87. [4, 8, 30, 44, 50, 69]

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. [53]

LIU, DE AND ADIB BAGH (2020): "Preserving bidder privacy in assignment auctions: design and measurement," *Management Science*, 66, 3162–3182. [51]

MACKENZIE, ANDREW (2020): "A revelation principle for obviously strategy-proof implementation," *Games and Economic Behavior*, 124, 512–533. [51]

MACKENZIE, ANDREW AND YU ZHOU (2022): "Menu mechanisms," *Journal of Economic Theory*, 204, 105511. [18, 51]

MCLEAN, RICHARD AND ANDREW POSTLEWAITE (2002): "Informational size and incentive compatibility," *Econometrica*, 70, 2421–2453. [53]

MCMILLAN, JOHN (1994): "Selling spectrum rights," *Journal of Economic Perspectives*, 8, 145–162. [1, 2]

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. [53]

MILGROM, PAUL AND ILYA SEGAL (2020): "Clock auctions and radio spectrum reallocation," *Journal of Political Economy*, 128, 1–31. [2, 38, 52]

MOULIN, H. (1980): "On Strategy-Proofness and Single Peakedness," *Public Choice*, 35, 437–455. [32]

NISSENBAUM, HELEN (2004): "Privacy as contextual integrity," *Washington Law Review*, 79, 119. [53]

NISSIM, KOBBI, CLAUDIO ORLANDI, AND RANN SMORODINSKY (2012): "Privacy-aware mechanism design," in *Proceedings of the 13th ACM Conference on Electronic Commerce*, 774–789. [53]

NURMI, HANNU AND ARTO SALOMAA (1993): "Cryptographic protocols for Vickrey auctions," *Group Decision and Negotiation*, 2, 363–373. [52]

OLLAR, MARIANN, MARZENA J ROSTEK, AND JI HEE YOON (2021): "Privacy in markets," *Available at SSRN 3071374*. [51]

PAI, MALLESH M AND AARON ROTH (2013): "Privacy and mechanism design," *ACM SIGecom Exchanges*, 12, 8–29. [53]

PYCIA, MAREK AND MADHAV RAGHAVAN (2022): "Non-bossiness and First-Price Auctions," Tech. rep., CEPR Discussion Paper No. DP16874. [8, 48]

PYCIA, MAREK AND PETER TROYAN (2023): "A theory of simplicity in games and mechanism design," *Econometrica*, 91, 1495–1526. [8, 50, 51]

ROTH, AARON AND GRANT SCHOENEBECK (2012): "Conducting truthful surveys, cheaply," in *Proceedings of the 13th ACM Conference on Electronic Commerce*, 826–843. [53]

ROTHKOPF, MICHAEL H, THOMAS J TEISBERG, AND EDWARD P KAHN (1990): "Why are Vickrey auctions rare?" *Journal of Political Economy*, 98, 94–109. [2]

SATTERTHWAITE, MARK A AND HUGO SONNENSCHEIN (1981): "Strategy-proof allocation mechanisms at differentiable points," *The Review of Economic Studies*, 48, 587–597. [8, 47, 48]

SEGAL, ILYA (2007): "The communication requirements of social choice rules and supporting budget sets," *Journal of Economic Theory*, 136, 341–378. [18]

SHAPLEY, LLOYD AND HERBERT SCARF (1974): "On cores and indivisibility," *Journal of mathematical economics*, 1, 23–37. [26]

SÖNMEZ, TAYFUN AND M UTKU ÜNVER (2010): "Course bidding at business schools," *International Economic Review*, 51, 99–123. [28]

THOMPSON, WILLIAM (2014): "Non-bossiness," *Rochester Center for Economic Research, Working Paper Number 586*. [47]

VICKREY, WILLIAM (1961): "Counterspeculation, auctions, and competitive sealed tenders," *The Journal of finance*, 16, 8–37. [30]

WILSON, ROBERT (1989): "Game Theoretic Analysis of Trading," in *Advances in Economic Theory: Fifth World Congress*, CUP Archive, 12, 33. [30]

WOODWARD, KYLE (2020): "Self-Auditable Auctions," Tech. rep., Working paper. [51]

XIAO, DAVID (2013): "Is privacy compatible with truthfulness?" in *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, 67–86. [53]

APPENDIX A: PROOFS

A.1. *Proof of Corollary 1.*

PROOF OF PROPOSITION 1: 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 $\hat{\mathbf{\Theta}}$ such that all types are inseparable under $\hat{\mathbf{\Theta}}$ and ϕ is non-constant on this set.

As ϕ is non-constant on $\hat{\mathbf{\Theta}}$, the protocol must make a query separating type profiles $(\theta, \boldsymbol{\theta}_{-i})$ and $(\theta', \boldsymbol{\theta}_{-i})$ for $\theta \not\sim_{i,\phi,\hat{\mathbf{\Theta}}} \theta'$ 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

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

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 $(\theta^l, \boldsymbol{\theta}_{-i}), (\theta^{l+1}, \boldsymbol{\theta}_{-i}) \in \hat{\boldsymbol{\Theta}} \subseteq \boldsymbol{\Theta}_v$, $(\theta^{l+1}, \boldsymbol{\theta}_{-i})$ and $(\theta^{l+1}, \boldsymbol{\theta}_{-i})$ lead to distinct terminal nodes. By direct inseparability, there is $\boldsymbol{\theta}_{-i}$ such that $(\theta^l, \boldsymbol{\theta}_{-i}), (\theta^{l+1}, \boldsymbol{\theta}_{-i}) \in \hat{\boldsymbol{\Theta}}$ 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:

$$\text{For any terminal nodes } w, w' \text{ whose earliest point of departure in } P \text{ is } v, \text{ there are no } (\theta_i, \boldsymbol{\theta}_{-i}) \in \Theta_{v'} \text{ and } (\theta_i', \boldsymbol{\theta}_{-i}) \in \Theta_{v''} \text{ such that } \phi(\theta_i, \boldsymbol{\theta}_{-i}) = \phi(\theta_i', \boldsymbol{\theta}_{-i}). \tag{10}$$

Note that a protocol that satisfies (10) 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 $\boldsymbol{\Theta}_v \subseteq \boldsymbol{\Theta}$. 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 $\hat{\boldsymbol{\Theta}}$ under which all types are inseparable, there are types θ, θ' that are separable under $\boldsymbol{\Theta}_v$ for agent i. Consider the binary query that separates the equivalence class of θ, $[\theta]_{i,\phi,\boldsymbol{\Theta}_v}$, from its complement $\Theta_{v,i} \setminus [\theta]_{i,\phi,\boldsymbol{\Theta}_v}$, which is non-empty as it contains at least θ'. By definition of $\sim$, for any $\boldsymbol{\theta}_{-i}$,

$$\phi(\tilde{\theta}_i, \boldsymbol{\theta}_{-i}) \neq \phi(\tilde{\theta}_i', \boldsymbol{\theta}_{-i}), \tag{11}$$

implying that (10) continues to hold. By induction, it holds at all internal nodes. *Q.E.D.*

A.2. *Proof of Proposition 3.*

PROOF OF PROPOSITION 3: 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.

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 $y = ((j,a),(i,b))$).

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 C there is no oversupply, i.e. $\sum_{c \in C} \kappa(c) = n$, 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 $i, j \in N$. 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 $\theta_i, \theta'_i, \theta_j, \theta'_j \in \Theta$ such that the condition (6) 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. *Strengthening of Proposition 4 Allowing for Ties.*

We include a strengthening of Proposition 4 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. We use type separability to construct a larger counterexample for contextual privacy.

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.

PROOF OF PROPOSITION 4: This proof proceeds in two steps. First, we construct a direct contradiction of Corollary 1 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.

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.

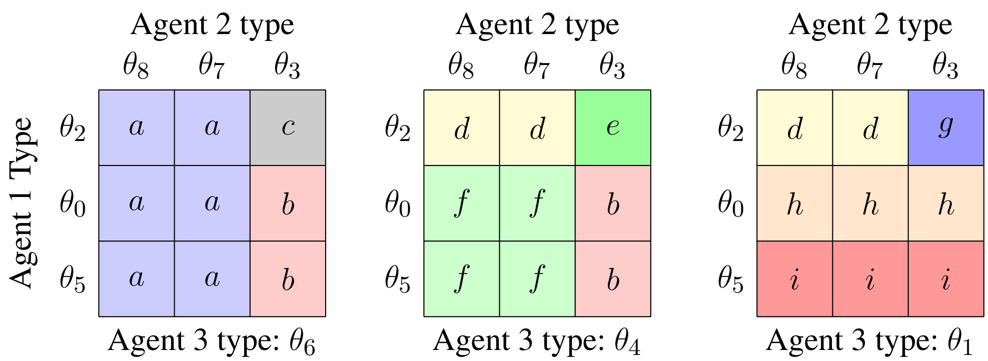

FIGURE A.2.—Counterexample ϕ^{SPA} with $n = 3$ and $|\Theta| = 9$.

- 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 $\boldsymbol{\Theta}'' = \boldsymbol{\Theta}' \times \bigtimes_{i \in \{4,\ldots,n\}} \Theta_i$ where each Θ for agents $i \in \{4, \ldots, n\}$ 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 5.*

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 5. 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.

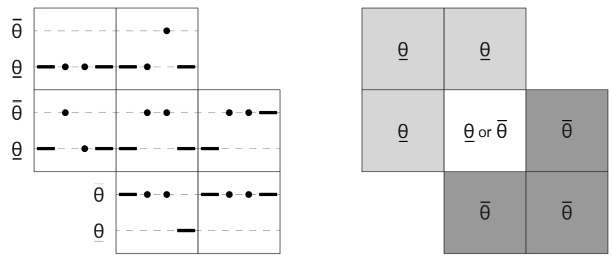

FIGURE A.3.—Applying the Corners Lemma to the double auction. Combinations of types for agents i, j, k, ℓ (left); required prices t in an efficient choice rule.

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 6.*

PROPOSITION: For any $k_1, k_2, \ldots, k_{n-1} \in X \cup \{-\infty, \infty\}$ 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.

PROOF: The proof is related to the one for Proposition 5, 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

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

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 2.*

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), $s_v \colon \boldsymbol{\Theta} \to \text{children}(v)$ be a query and let

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

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

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

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

DEFINITION: Let $(\Theta, \leq)$ be a finite ordered type space and $\text{succ}(\theta)$ resp. $\text{pred}(\theta)$ 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*

$$\overline{\theta}_{v,v'} = \max\{\overline{\theta}_v, \overline{\theta}_{v'}\} \tag{12}$$

$$\underline{\theta}_{v,v'} = \min\{\underline{\theta}_v, \underline{\theta}_{v'}\} \tag{13}$$

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'',[26] i.e.

[26]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.

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 (12). 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 (12):

- $(\theta_i, \boldsymbol{\theta}_{-i})$ 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 (13).

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,

$$P \succeq_\phi P_{v'',v'}. \tag{14}$$

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

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

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 5: *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 $w \in \mathrm{children}(v)$ such that $\boldsymbol{\Theta}_w \neq \emptyset$. Denote by $\mathrm{Dedup}_v(P)$ the protocol that attaches $\mathrm{subtree}(w)$ to $\mathrm{parent}(v)$ and deletes $\mathrm{subtree}(w')$ for all $w' \in \mathrm{children}(v) \setminus \{w\}$.

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 $\mathrm{Dedup}_v(P)$. Continuing with the next-earliest repetition v', we apply $\mathrm{Dedup}'_v(P)$. 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 6: *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 $\mathrm{thresh}(v) = \mathrm{thresh}(v'')$ and $\mathrm{thresh}(v') \neq \mathrm{thresh}(v), \mathrm{thresh}(v')$. This, however, would mean that the the protocol is not fully deduplicated. *Q.E.D.*

Q.E.D.

A.7. *Proof of Proposition 7.*

PROPOSITION: Assume $|N| \geq 2$. Then, $P_{\text{Asc.Join}} \sim_\phi P_{\text{Asc.}}$, but $P_{\text{Asc.Join}}$ is a strict ϕ^{SPA}-contextual privacy improvement.

PROOF: The ascending auction is a weak contextual privacy improvement as, for every type profile, it asks a subset of the queries to agents. Consider a type profile in which the third-highest agent i is asked last in the order of the ascending join auction. This agent will be asked whether their type is at least the second-highest type, to which they give a negative answer. Consider a type profiles $(\theta_i, \boldsymbol{\theta}_{-i}), (\theta'_i, \boldsymbol{\theta}_{-i})$ for $\theta_i > \theta'_i$. The triple $(i, (\theta_i, \boldsymbol{\theta}_{-i}), (\theta'_i, \boldsymbol{\theta}_{-i}))$ will constitute a contextual privacy violation under the ascending auction (as all loser types are fully learned), but not under the ascending join auction. *Q.E.D.*

A.8. *Proof of Lemma 2.*

LEMMA: There is a $\mathfrak{S}$-protocol P implementing ϕ in dominant resp. obviously dominant resp. perfect Bayesian strategies and contextual privacy violations $\Gamma \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$ if and only if there is a direct $\mathfrak{S}^*$-protocol P^* implementing ϕ with contextual privacy violations $\Gamma \subseteq N \times \boldsymbol{\Theta} \times \boldsymbol{\Theta}$.

PROOF: Note that the reduction in Lemma 1 leads to the same outcomes and information state under P and P^*. This means it preserves incentives. *Q.E.D.*

A.9. *Proof of Proposition 10.*

PROPOSITION: The descending protocol for ϕ^{FP} is implementable in perfect Bayesian strategies.

PROOF: This result is a direct consequence of the observation that in an independent private value setting, the best response for the bidding agent depends only on the expectation of $\max \boldsymbol{\theta}_{-i}$ conditional on θ_i which is not changing conditional on events $\{\boldsymbol{\theta} : \theta_i < \theta, i = 1, 2, \ldots, n\}$. *Q.E.D.*

A.10. *Proof of Proposition 8.*

PROPOSITION: An ascending-join protocol for ϕ^{SP} have equilibria in obviously dominant strategies.

PROOF: We show that both the ascending and the ascending join protocols are *personal-clock auctions* Li (2017, Theorem 3). Given this Li (2017, Theorem 3) implies our result. The "personal clocks" in this case are to only raise running prices for agents that are active in the join auction. *Q.E.D.*

APPENDIX B: SERIAL DICTATORSHIPS ARE CONTEXTUALLY PRIVATE UNDER SEQUENTIAL ELICITATION

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

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 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 $A \subseteq N \times C$ be an outcome. A *partial assignment* $N(A)$ is the set of agents who have an assigned object in A, i.e. $N(A) = \{i \in N : \exists c \in C : (i, c) \in A\} \subseteq N$. 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. $R(A) \coloneqq \{c \in C \colon \nexists i \in N : (i,c) \in A\}$.

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 $\pi(1)$, $\pi(2)$, $\ldots$, $\pi(i)$ 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.

	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 ARBITRARY EFFICIENT 2-AGENT SOCIAL CHOICE FUNCTIONS (LEFT); UNDER AN EFFICIENT CHOICE RULE WHICH BREAKS TIES LEXICOGRAPHICALLY (ϕ^{FAIR}) (MID-LEFT, MIDDLE); UNDER A SERIAL DICTATORSHIP ϕ^{sd} (MID-RIGHT, RIGHT)

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 an example of 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'.

Four different assignments remain. The first two assignments contain a Corner in the sense of Corollary 1, 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 $N \times \Theta \times \{0, 1\}$ and edges from $((i, \theta, 0)$ to $(i + 1, \theta, 0))$, for $i \in N \setminus \{n\}$ 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 $i \in N$ 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 $\Theta \times N \times \{1\}$. 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 $\max_{i \in N} \theta_i$. 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, $t(\boldsymbol{\theta}) = f(\theta_{[k]})$, $k \in N$.

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 4. The quantile that the price depends on is chosen by two types. The Corners Lemma can be applied analogously. *Q.E.D.*

APPENDIX D: A TYPE-BASED NOTION OF PROTOCOL EQUIVALENCE

One may also consider equivalence based on equivalence of the sets of contextual privacy violations.

DEFINITION: We say that protocols P and P' for ϕ are *perfectly equivalent* with respect to contextual privacy if $(i, \boldsymbol{\theta}, \boldsymbol{\theta}')$ is a contextual privacy violation under P if and only if it is a contextual privacy violation under P'.

The reader might wonder whether a result similar to Theorem 2 holds also for perfect equivalence. This is not true. Consider the following 2-player social choice function with 2- resp. 4-element type spaces and two outcomes.

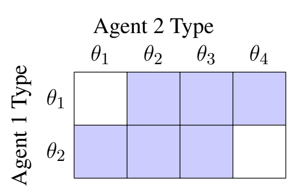

FIGURE D.1.—Counterexample for analogue of Theorem Theorem 2 for perfect contextual privacy equivalence. Blue corresponds to the same outcome, white corresponds to another outcome.

Three distinct equivalence classes in $\sim_p hi$ are represented by the following protocols:

- Query agent 1 first, and then ask a query to agent 2 to compute ϕ;
- Query agent 2 whether they are θ_4; If the answer is positive, query agent 1. If the answer is negative, ask agent 2 whether they are θ_1. If the answer is positive, query agent 1.
- Query agent 2 whether they are θ_1; If the answer is positive, query agent 1. If the answer is negative, ask agent 2 whether they are θ_4. If the answer is positive, query agent 1.

Note that the second and third protocol are not bimonotonic, and the only element in their perfect equivalence classes.

APPENDIX E: 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 $X = \{x_1, x_2, x_3, x_4\}$. 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 F: DEFINITIONS OF PROTOCOLS

Algorithm 1: 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_{next};
- $R \leftarrow R_{\text{next}}$;
- $\tilde{\theta} \leftarrow \operatorname{succ}_{\Theta} \tilde{\theta}$;

Algorithm 2: 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 $\tilde{\theta}$*
  - $\tilde{\theta} \leftarrow \operatorname{pred}_{\Theta}(\tilde{\theta})$;

Algorithm 3: 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: $\text{active} \leftarrow$ first two agents from N

while *true* **do**
  foreach *agent i in* active **do**
    if $\theta_i < \tilde{\theta}$ **then**
      Remove i from active;
      if *There is agent i that never has been active* **then**
        $\text{active} \leftarrow \text{active} \cup \{i\}$;
      else
        return *Remaining agent i from active,* $\text{pred}_{\Theta}(\tilde{\theta})$

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**
  $\text{winnerFound} \leftarrow$ False;
  $\text{winner} \leftarrow \emptyset$;
  foreach *agent i in $R \setminus$ winner* **do**
    if winnerFound **then**
      if $\theta_i \geq \tilde{\theta}$ **then**
        $\text{winnerFound} \leftarrow$ True;
        $\text{winner} \leftarrow \{i\}$;
    else
      if $\theta_i \geq \tilde{\theta}$ **then**
        return Agent winner and price $\tilde{\theta}$;
  $\tilde{\theta} \leftarrow \text{pred}_{\Theta}(\tilde{\theta})$;