Unconditional Privacy in Social Choice

Felix Brandt
Computer Science Department
Stanford University
Stanford CA 94305
brandtf@cs.stanford.edu

Tuomas Sandholm
Computer Science Department
Carnegie Mellon University
Pittsburgh PA 15213
sandholm@cs.cmu.edu

Abstract

The aggregation of conflicting preferences is an important issue in human society and multiagent systems. Due to its universality, voting among a set of alternatives has a central role among preference aggregation mechanisms. We consider the most general case of voting in which the voters' rankings of alternatives are mapped to a collective ranking of alternatives by a so-called social welfare functional (SWF). Maintaining privacy of individuals' preferences is crucial in order to guarantee freedom of choice (*e.g.*, lack of vote coercing and reputation effects), and to not facilitate strategic voting. We investigate whether *unconditional full privacy* can be achieved in preference aggregation, that is, privacy that relies neither on trusted third parties (or on a certain fraction of the voters being trusted), nor on computational intractability assumptions. More precisely, we study the existence of distributed protocols that allow voters to jointly determine the collective preference ranking without revealing further information. We prove that there exists no SWF that is non-dictatorial, Paretian, monotonic, and privately computable (any three of these properties can be achieved). Moreover, we show that replacing privacy with anonymity enables the joint computation of arbitrary symmetric SWFs.

1 Introduction

Whenever a group of autonomous entities, such as humans or computational agents, strive for a global decision, they need to aggregate their possibly conflicting preferences. Typically, this is achieved by voting among a set of alternatives such as candidates, plans, or resource assignments. One of the most prominent results in social choice theory is Arrow's impossibility theorem (see Section 5) which states that even a rather modest set of desiderata cannot be obtained when aggregating preferences. This explains why a wide variety of social welfare functionals (SWFs), *i.e.*, functions that aggregate a vector of diverse individual preference rankings into one collective preference ranking, with differing advantages and disadvantages have evolved.

Computer scientists from various fields such as multiagent systems, artificial intelligence, and complexity theory are showing increasing interest in social choice theory. Some recent advances include complexity analyses of strategic voting [Bartholdi, III *et al.*, 1989a; Conitzer and Sandholm, 2003] and of computing the SWF itself [Bartholdi, III *et al.*, 1989b; Hemaspaandra *et al.*, 1997; Rothe *et al.*, 2003]. This paper deals with the possibility of jointly computing the SWF while preserving *privacy* in an information-theoretic sense.

Maintaining privacy of individuals' preferences is crucial in preference aggregation. For one, this is required to achieve freedom of choice: avoiding vote coercing, allowing an agent to vote for a casino over a school without fear of adverse reputation effects, *etc.* Second, learning about others'

preferences opens the possibility for an agent to benefit from voting insincerely—and according to another seminal impossibility theorem [Gibbard, 1973; Satterthwaite, 1975], *all* SWFs (except dictatorial ones) are manipulable in this sense, as long as there are more than two alternatives.

Traditionally, privacy is obtained by introducing one or more third parties who privately receive the individual preferences and then publicly declare the outcome. In such a model, privacy completely depends on the trustworthiness of these parties as it is virtually impossible to prevent a third party from revealing sensitive information. This paper investigates whether privacy that does not rely on trusted third parties can be achieved in social choice. More precisely, we study the existence of distributed protocols that allow voters to jointly determine the outcome of a SWF by exchanging messages without revealing unnecessary information. This makes the problem a special case of secure multiparty computation as first proposed by Yao [Yao, 1982].

One dimension along which privacy guarantees in distributed protocols differ is how many of the agents need to collude before privacy is breached. In this paper, we will require the strongest variant, *full privacy* or so-called $(n-1)$-privacy, which means that no information (beyond what can be inferred from the outcome) can be uncovered by a coalition that does not include *all* of the n agents.

Another criterion along which privacy guarantees differ is whether or not the computational power of the adversary is limited. In fact, using computational intractability as a barrier against undesirable behavior has a long tradition in modern cryptography since Diffie and Hellman's seminal paper [Diffie and Hellman, 1976]. When relying on intractability assumptions (such as the conjectured hardness of factoring), it has been shown that arbitrary functions can be jointly computed so that no private input can be revealed by a polynomially-bounded adversary [Goldreich *et al.*, 1987]. Unfortunately, computational intractability not only relies on the unproven assumption $\mathcal{P} \neq \mathcal{NP}$ but also on the widely unknown field of average-case complexity and further, more specific assumptions. Moreover, even when these conjectures are true, it may be possible to breach privacy in the future when sufficient computational power becomes available. In this paper, we will study *unconditional privacy* (aka. *non-cryptographic* or *information-theoretic* privacy), where the adversary's computational power is unlimited and a complete network of private channels between agents is given. When we speak of *privacy* in the rest of this paper, we mean unconditional full privacy.

It is known that only a restricted class of functions can be computed privately in this model.[1] There is yet no complete characterization of this class of functions (except for special cases like Boolean [Chor and Kushilevitz, 1989] and 2-ary functions [Kushilevitz, 1989]). However, there are sufficient conditions for showing that a function is *not* privately computable. Using these conditions, we derive the the main theorem of this paper which states that any SWF satisfying a very reasonable set of desiderata cannot be privately computed.

Theorem 1

(a) There is no social welfare functional that is non-dictatorial, Paretian, monotonic, and privately computable.[2]

(b) There are social welfare functionals that satisfy any combination of three of these four properties.

[1] When assuming that a majority of the agents is trustworthy (this is not *full* privacy), *all* functions can be jointly computed in the unconditional model [Ben-Or *et al.*, 1988; Chaum *et al.*, 1988] (assuming passive adversaries).

[2] When there are just two alternatives, Pareto-optimality can be replaced with the weaker property of non-constancy.

The remainder of this paper is structured as follows. In Section 2, the underlying social choice and security model are explained. Theorem 1 (a) (and a similar result for social choice functions) is proven in Section 3. Section 4 shows that arbitrary symmetric SWFs can be computed when replacing privacy with anonymity. In Section 5, we review two related impossibility results and existing work on unconditional privacy in voting. The paper concludes with a summary of the results in Section 6. Part (b) of Theorem 1 is proven in Appendix A.

2 Preliminaries

We consider a set of n agents A who vote among a set of m alternatives O. Each agent i possesses a private preference ranking $x_i \in \Theta$ among these alternatives ($|\Theta| = m!$). The unique rational (*i.e.*, transitive and complete) preference relation induced by x_i is denoted by $\succ_i$. A *social welfare functional* (SWF) $f : \Theta^n \to \Theta$ aggregates preferences into one collective preference ranking x (the collective preference relation $\succ$). For ease of notation, we will use $f(x_1, x_2, \ldots, x_n) = f(\vec{x}) = x$ as well as $f(\succ_1, \succ_2, \ldots, \succ_n) = \succ$. Often, only the socially *most-preferred* alternative is of interest (rather than a complete ranking of alternatives). A *social choice function* (SCF) $g : \Theta^n \to O$ maps preferences to a *single* alternative or candidate: the election winner. If $m = 2$, the notions of SWF and SCF are equivalent. Although our results apply to both SWFs and SCFs, we will refer to the more general concept of SWFs in the remainder of this section.[3]

The SWF is jointly computed by agents using a distributed, randomized protocol consisting of several rounds. In order to enable secure message exchange, we assume a complete synchronous network of private channels between agents. In each round, each agent may send a message to any other agent. Each message an agent sends is a function of his preferences, his independent random input r_i, the messages he received so far, and the recipient. When the protocol is finished, all agents know the value of $f(\cdot)$.

We assume that the adversary is *passive* (or "honest-but-curious"). In other words, agents do not deviate from the prescribed protocol but nevertheless try to deduce private information from any data available to them. Clearly, *negative* results in the passive adversary model considered in this paper also hold in more adversarial models, *e.g.*, models with active or adaptive adversaries.

In the following, we formally define the four properties to be used in the main theorem. The first three properties are traditionally used in social choice theory whereas the last one is commonly used in secure multiparty computation. Generally, it can be said that we impose very weak restrictions on the "social" attributes of the SWF and very strong restrictions on privacy.

Definition 1 (Dictatorship) *A SWF $f(\vec{x})$ is* dictatorial *if*

$$\exists i \in \{1, 2, \ldots, n\} \text{ so that } \forall \vec{x} \in \Theta^n : \ f(\vec{x}) = x_i.$$

When using a dictatorial SWF, there is always one agent who alone enforces the "collective" ranking. Clearly, this is undesirable and consequently non-dictatorship is arguably one of the weakest properties that any SWF should satisfy.

Another weak property is Pareto-optimality which states that the collective ranking should be optimal in the sense that no agent can be made better off without reducing the contentment of another agent. In the context of SWFs, this merely means that if *all* agents prefer one alternative over another, then this is also the case in the collective preference ranking. For this reason, it is sometimes said that a Paretian SWF "respects unanimity".

[3] In contrast to common social choice theory, a version of Theorem 1 for SCFs is not "stronger" in the sense that it trivially implies the same statement for SWFs.

Definition 2 (Pareto-optimality) *A SWF* $f(\succ_1, \succ_2, \ldots, \succ_n) = \succ$ *is* Paretian *if*

$$\forall a, b \in O: \qquad (\forall i: \ a \succ_i b) \quad \Rightarrow \quad a \succ b.$$

A related, but slightly stronger, property is monotonicity. Loosely speaking, a SWF is monotonic if no agent can make an alternative rise in the collective ranking by reducing its rank in his individual preferences while all remaining preferences are unchanged. This is also called *positive responsiveness* or *positive association of social and individual values.*[4]

Definition 3 (Monotonicity) *Let* $f(\succ_1, \succ_2, \ldots, \succ_n) = \succ$, $f(\succ_1, \succ_2, \ldots, \succ'_i, \ldots, \succ_n) = \succ'$, *and* $\succ_i$ *and* $\succ'_i$ only *differ in the relative ranking of alternatives* a *and* b *so that* $b \succ_i a$ *and* $a \succ'_i b$. *SWF* $f(\cdot)$ *is* monotonic *if*

$$\forall i \in \{1, 2, \ldots, n\}, a, b \in O: \qquad a \succ b \quad \Rightarrow \quad a \succ' b.$$

Monotonicity is a very reasonable property that all common SWFs satisfy. A non-monotonic SWF is counter-intuitive (at least) because supporting an alternative may reduce its collective rank. Consequently, a rational agent may have to vote for $b \succ a$ even though his preference is $a \succ b$.[5]

As usual, full privacy in the context of information-theoretic function evaluation is defined as follows: A distributed protocol for computing SWF $f(x_1, x_2, \ldots, x_n) = x$ is unconditionally fully private if any coalition of agents is incapable of uncovering any information besides what can be inferred from x and the coalition's preferences. More formally:

Definition 4 (Privacy) *For any* $T \subseteq \{1, 2, \ldots, n\}$ *and every two input vectors* $\vec{x}, \vec{y} \in \Theta^n$ *satisfying* $\forall i \in T: \ x_i = y_i$ *and* $f(\vec{x}) = f(\vec{y})$, *and for every choice of random inputs* $\{r_i\}_{i \in T}$, *the messages seen by agents belonging to* T *in both cases are identically distributed. Let* VIEW_T *be a function that, given the vector of individual inputs and random values, yields the concatenation of all (prefix-free) messages exchanged between members of* T *and* $\bar{T} = \{1, 2, \ldots, n\} \setminus T$. *A protocol for computing* $f(\cdot)$ *is* private *if*

$$\langle \text{VIEW}_T(\vec{x}, \{r_i\}_{i \in T}) \rangle = \langle \text{VIEW}_T(\vec{y}, \{r_i\}_{i \in T}) \rangle$$

where $\langle \ldots \rangle$ *denotes the probability distribution of the inner term with the probability taken over* $\{r_i\}_{i \in \bar{T}}$.

All four properties defined above can be translated from SWFs to SCFs in a straightforward way.

3 Main Theorem

It might seem unlikely that any "relevant" function can be computed at all, given this restrictive definition of privacy. Indeed, it is well-known and often mentioned that "only very few" functions are privately computable. However, there are some simple privately computable functions whose relevance cannot be denied. For example, the sum of individual inputs (and thus the arithmetic mean) can be computed privately (see Proposition 1). Moreover, it has recently been shown that (in the absence of ties) the outcome of first-price sealed-bid auctions can be computed privately whereas

[4] Sometimes, a different, much stronger property that is equivalent to "independence of irrelevant alternatives" (see Section 5) in the context of SCFs is also called *monotonicity.*

[5] It is important to note that monotonicity is weaker than strategy-proofness if $m > 2$ since "irrelevant" alternatives are not considered. As a matter of fact, strategy-proofness implies monotonicity but not vice versa.

the outcome of second-price sealed-bid (Vickrey) auctions cannot [Brandt and Sandholm, 2004]. As a matter of course, the lack of a *complete* characterization of privately computable functions adds to the obscurity of this class of functions. The main goal of this paper is to investigate whether any "reasonable" SWF can be computed privately. It turns out that existing results on unconditional privacy are sufficient to prove the impossibility of privately computing a wide (and relevant) class of SWFs. We will apply the following necessary conditions for the private computability of a function.

Lemma 1 (Corners Lemma) *[Chor and Kushilevitz, 1989] Let $f : Y \times Z \to W$ be a privately computable 2-ary function. For every $y_1, y_2 \in Y$ and $z_1, z_2 \in Z$, if $f(y_1, z_1) = f(y_1, z_2) = f(y_2, z_1) = a$, then $f(y_2, z_2) = a$.*

Lemma 2 (Partition Lemma) *[Chor and Kushilevitz, 1989] Let $f : Y_1 \times Y_2 \times \cdots \times Y_n \to W$ be a privately computable n-ary function. Then, for each $i \in \{1, 2, \ldots, n\}$ the 2-ary function $f_2(y_i, (y_1, y_2, \ldots, y_{i-1}, y_{i+1}, y_{i+2}, \ldots, y_n)) \stackrel{def}{=} f(y_1, y_2, \ldots, y_n)$ is privately computable.*[6]

By combining Lemma 1 and Lemma 2, one can obtain a necessary condition for the existence of protocols that privately compute an n-ary function.[7] This can be used to prove that a SWF is *not* privately computable.

Lemma 3 *Let $\vec{y}$ and $\vec{z}$ be vectors of $n-1$ preference rankings and y and z be single preference rankings. It is* impossible *to privately compute SWF $f(\cdot)$ if*

$$\exists\, \vec{y}, \vec{z} \in \Theta^{n-1},\, y, z \in \Theta : \quad f(\vec{y}, y) = f(\vec{y}, z) = f(\vec{z}, y) = a \;\wedge\; f(\vec{z}, z) \neq a \quad .$$

This is called an "embedded OR*".*

The proof of Theorem 1 is composed as follows. We first show that any non-dictatorial, monotonic, and privately computable SWF for $m = 2$ is constant. We then reduce the impossibility of SCFs and SWFs with the same properties and an arbitrary number of candidates m to the former case by adopting Pareto-optimality.

Theorem 2 *Every non-dictatorial, monotonic and privately computable SWF for two alternatives ($m = 2$) is constant.*

Proof: Let $O = \{a, b\}$ be a set of two alternatives.

Definition 5 *A subset $D \subseteq A$ is called* decisive *for alternative a if $\forall i \in D : x_i = a$ implies that*

$$\forall x_i \in O \text{ with } i \in A \backslash D : \; f(x_1, x_2, \ldots, x_n) = a \; .$$

In other words, D is decisive for a if all agents in D voting for a always leads to social choice a, no matter what the remaining agents do.

Lemma 4 *Let SWF $f(\cdot)$ be non-dictatorial, monotonic, and privately computable. If any set of agents D is decisive for a, than $f(\cdot)$ is constant and always yields a.*

[6] This is a special case of the Partition Lemma as defined in [Chor *et al.*, 1994] for $t = n - 1$.

[7] In fact, in the Boolean case where $Y = Z = W = \{a, b\}$ (which we will consider in Theorem 2) this condition is necessary *and* sufficient (see Theorem 6).

Proof: Relying on the induction principle, it suffices to show that $D\backslash\{d\}$ for any agent d is decisive if D is decisive. This ultimately leads to the empty set being decisive for a, *i.e.*, $f(\vec{x}) = a$ for any $\vec{x}$. We will prove the above statement by contradiction. Without loss of generality, let $D = \{1, 2, \ldots, d\}$ and suppose that D is decisive. Let us furthermore assume that $D\backslash\{d\}$ is *not* decisive in order to cause a contradiction. $\vec{y} \in \{a, b\}^{d-1}$ and $\vec{z} \in \{a, b\}^{n-d}$ are vectors of arbitrary preference profiles. In a slight abuse of notation, we will write a^d for the vector $(\underbrace{a, \ldots, a}_{d})$.

The following statements can be deduced from these premisses.

(i) $f(a^{d-1}, a, \vec{z}) = a$ for any $\vec{z}$ because D is decisive.

(ii) $f(a^{d-1}, b, b^{n-d}) = b$. Otherwise, if $f(a^{d-1}, b, b^{n-d}) = a$, $D\backslash\{d\}$ would be decisive because $f(a^{d-1}, \vec{w}) = a$ for any $\vec{w} \in \{a, b\}^{n-d-1}$ (if any of the last $n-d-1$ agents changes his preference from b to a, the collective preference cannot revert back to b according to monotonicity).

(iii) $f(a^{d-1}, b, \vec{z}) = b$ for any $\vec{z}$. Otherwise, $f(a^{d-1}, a, b^{n-d}) = a$, $f(a^{d-1}, b, b^{n-d}) = b$, $b(a^{d-1}, a, \vec{z}) = a$, $f(a^{d-1}, b, \vec{z}) = a$ would form an embedded OR as defined in Lemma 3 (see Table 1).

(iv) $f(\vec{y}, b, \vec{z}) = b$ for any $\vec{y}, \vec{z}$. Since $f(a^{d-1}, b, \vec{z}) = b$ according to (iii), monotonicity implies $f(\vec{y}, b, \vec{z}) = b$ for any $\vec{y}$ (if any of the first $d-1$ voters changes his preference from a to b, this cannot revert the collective preference back to a).

(v) $f(\vec{y}, a, \vec{z}) = a$ for any $\vec{y}, \vec{z}$. Otherwise, (a^{d-1}, a, b^{n-d}), (a^{d-1}, b, b^{n-d}), $(\vec{y}, a, \vec{z})$, $(\vec{y}, b, \vec{z})$ would form an embedded OR as defined in Lemma 3 (see Table 1).

$f(\cdot)$	a	b
$a^{d-1}\,.\,b^{n-d}$	a (i)	b (ii)
$a^{d-1}\,.\,\vec{z}$	a (i)	b (iii)
$\vec{y}\,.\,\vec{z}$	a (v)	b (iv)

Table 1: Construction of the proof of Lemma 4

The last two statements say that d's preference is always equal to the collective preference, *i.e.*, agent d is a dictator, violating the precondition that f is non-dictatorial. This contradiction implies that $D\backslash\{d\}$ is decisive. By induction, this yields that any subset of D, including the empty set, is decisive. As a consequence, $f(\vec{x}) = a$ for any $\vec{x}$. ■

We will now show that any non-dictatorial, monotonic, and privately computable SWF $f(\cdot)$ is constant. Consider the set of all agents A. Suppose that A is not decisive for a, *i.e.*, $f(a^n) = b$. It follows from monotonicity that $f(\vec{x}) = b$ for any $\vec{x}$, resulting in a constant SWF. On the other hand, if A is decisive for a, Lemma 4 implies that the collective preference is always a, *i.e.*, $f(\vec{x}) = a$ for any $\vec{x}$. In other words, $f(\cdot)$ is constant as well. This contradiction completes the proof of Theorem 2. ■

Bringing Pareto-optimality into play, we can extend the impossibility to SCFs and SWFs with an arbitrary number of alternatives.

Corollary 1 *There is no non-dictatorial, Paretian, monotonic, and privately computable SCF for any number of alternatives.*

Proof: We prove the statement by reducing it to Theorem 2. Let $O = \{a_1, a_2, \ldots, a_m\}$ be a set of alternatives and let preference relations $\succ_i$ be defined so that

$$\forall i \in \{1, 2, \ldots, n\}: \quad (a_1 \succ_i a_3) \wedge (a_2 \succ_i a_3) \wedge (\forall j \in \{3, 4, \ldots, m-1\}: \; a_j \succ_i a_{j+1}), \qquad (1)$$

i.e., all agents rank alternatives $a_3, a_4, \ldots, a_m$ in the same fixed order below alternatives a_1 and a_2. If there were a SCF $g(\cdot)$ that satisfies the mentioned properties, it would do so for any given combination of preferences, including the one defined above. Since $g(\cdot)$ is Paretian, alternatives $a_3, a_4, \ldots, a_m$ can never be ranked above a_1 or a_2 in the collective preference ranking because all agents prefer a_1 and a_2 over any other alternative. As a consequence, $g(\cdot)$ must yield either a or b. Furthermore, the collective choice of a or b only depends on the agents' relative ranking of a and b, *i.e.*, whether $a \succ_i b$ or $b \succ_i a$ holds. It cannot depend on the ranking of the remaining alternatives because they are all ranked identically by all agents. Therefore, if $g(\cdot)$ were non-dictatorial, Paretian, monotonic, and privately computable, it would yield a two-alternative SWF with the same properties when preferences satisfy (1). Such a SWF does not exist according to Theorem 2 (Pareto-optimality trivially implies non-constancy). ■

Corollary 2 *There is no non-dictatorial, Paretian, monotonic, and privately computable SWF for any number of alternatives.*

Proof: This statement differs from Corollary 1 in the fact that a SWF yields a complete ranking of alternatives rather than just the top-ranked alternative. Nevertheless, it can also be reduced to Theorem 2. Let $O = \{a_1, a_2, \ldots, a_m\}$ be a set of alternatives and let preference relations $\succ_i$ be defined according to (1). Following the argumentation of Corollary 1, the collective preference ranking yielded by any Paretian SWF $f(\cdot)$ will also satisfy (1). As a consequence, the only new information that agents learn from the collective preference ranking is whether $a \succ b$ or $b \succ a$ holds. Restricting preferences to (1), the existence of a SWF with the proclaimed properties for any m thus implies the existence of a SWF with the same properties for $m = 2$ which cannot exist due to Theorem 2. ■

Remarkably, the impossibility result is independent of symmetry among voters and neutrality (symmetry among alternatives). As a consequence, the impossibility also holds for asymmetric voting schemes in which agents' votes are treated unequally and voting schemes that are not neutral, *e.g.*, veto voting.[8] A proof showing that none of the four properties used in Corollary 2 is redundant for the impossibility to hold is given in Appendix A.

4 Anonymity

It is natural to ask which kind of privacy relaxations enable the private distributed computation of a SWF. When allowing more information to be revealed during the computation, it becomes more likely that a SWF can be privately computed. The lowest level of symmetric privacy is anonymity. Loosely speaking, anonymity means that exchanging two agents' preferences does not lead to different information to be revealed. For a formal definition, we restrict the equality of distributions in Definition 4 to the case of permuted input vectors.

[8] Veto voting is used to vote among two alternatives. The first alternative (usually the *status quo*) is chosen unless *all* agents vote for the other alternative.

Definition 6 (Anonymity) *Let T, $\vec{x}$, $\vec{y}$, and* VIEW_T *be defined as in Definition 4 and furthermore assume that $\vec{x}$ is a permutation of $\vec{y}$ (in addition to $\forall i \in T : x_i = y_i$ and $f(\vec{x}) = f(\vec{y})$). A symmetric SWF $f(x_1, x_2, \ldots, x_n)$ can be computed* anonymously *if*

$$\langle \text{VIEW}_T(\vec{x}, \{r_i\}_{i \in T}) \rangle = \langle \text{VIEW}_T(\vec{y}, \{r_i\}_{i \in T}) \rangle.$$

This notedly weaker, but practically relevant, restriction can be satisfied without imposing additional restrictions on the SWF.

Proposition 1 *Any symmetric SWF can be computed anonymously (in two rounds).*

Proof: As first observed by Benaloh, modular additions can be computed privately using a simple two-round protocol which builds upon the homomorphism of secret sharing [Benaloh, 1987]. Suppose each agent submits a vector consisting of $m!$ components where each component represents a complete preference ranking. The component belonging to an agent's ranking is 1, all remaining components are 0. Let $h : \{0,1\}^{m! \, n} \to \{1, 2, \ldots, n\}^{m!}$ be a function that adds all vectors (using a modulus greater than n). Any symmetric SWF can be computed by first privately computing $h(\cdot)$ and then (publicly) applying an appropriate function $k : \{1, 2, \ldots, n\}^{m!} \to \Theta$ that yields the collective preference ranking. ■

Obviously, the construction in the previous proof is inefficient in terms of communication complexity. The number of bits each bidder has to send is exponential in the number of candidates ($m! \, n \log n$). Nevertheless, most common SWFs can be computed anonymously with polynomial communication complexity since they only take voters' top choices or relations between pairs of alternatives into account [Brandt and Sandholm, 2005].

5 Related Research

Social choice theory is rife with impossibility results, probably the most well-known being Arrow's impossibility theorem. Besides the properties defined in Section 3, it uses the notion of *independence of irrelevant alternatives*. A SWF is independent of irrelevant alternatives if the relative ranking of a and b, *i.e.*, whether $a \succ b$ or $b \succ a$ holds, only depends on the agents' relative ranking of a and b (and not the rankings of any other alternatives).

Theorem 3 *[Arrow, 1963] There is no non-dictatorial, Paretian SWF that is independent of irrelevant alternatives unless there are only two alternatives.*

More recently, the impossibility of *efficiently* computing a SWF that satisfies a list of desirable properties has been shown. A SWF is *neutral* if it is symmetric among alternatives (no alternative is favored by the SWF). A collective preference ranking is called a *Condorcet ranking* if alternative a is globally preferred over b ($a \succ b$) when a *majority* of agents have the preference $a \succ_i b$. A well-defined Condorcet ranking only exists when the above procedure yields a transitive, *i.e.*, cycle-free, collective preference relation. A SWF is *Condorcet-consistent* if a Condorcet ranking is chosen whenever it exists. Finally, a SWF is consistent if all subgroups in any partitioning of A opt for the same ranking, then this ranking is also selected for the entire group of agents A.

Theorem 4 *[Young and Levenglick, 1978; Bartholdi, III* et al.*, 1989b] There is no SWF that is neutral, Condorcet-consistent, consistent, and polynomial-time computable (unless $\mathcal{P} = \mathcal{NP}$).*

Besides a huge body of research on voting protocols in the cryptographic (or so-called computational) model (which relies on intractability), there have also been advances in unconditionally private voting (*e.g.*, [Chaum, 1988; Pfitzmann and Waidner, 1992]). These approaches differ in several aspects from this paper as they pursue different goals. To the best of our knowledge, prior cryptographic work deals exclusively with plurality voting (a specific SWF). Moreover, privacy is just preserved to the extent of *anonymity*.

6 Conclusion

The aggregation of conflicting preferences is an important issue in human society and multiagent systems. Due to its universality, voting among a set of alternatives has a central role among preference aggregation mechanisms. We considered the most general case of voting in which the voters' rankings of alternatives is mapped to a collective ranking of alternatives by a SWF. We investigated whether *unconditional full privacy* can be achieved in preference aggregation, that is, privacy that relies neither on trusted third parties, nor on computational intractability assumptions.

We showed that, even though there are SWFs that can be privately computed, none of them can satisfy three modest desiderata at the same time: non-dictatorship, Pareto-optimality, and monotonicity. This impossibility can be transferred to the case of SCFs in which only the highest-ranked alternative is determined. Furthermore, we showed that if privacy is reduced to mere anonymity, arbitrary symmetric SWFs can be computed. Future work includes the investigation of privacy restrictions that lie in between anonymity and the rigorous version of privacy considered in the main part of this paper.

Acknowledgements

This material is based upon work supported by the Deutsche Forschungsgemeinschaft under grant BR 2312/1-1 and by the National Science Foundation under grants IIS-9800994, ITR IIS-0081246, and ITR IIS-0121678, and a Sloan Fellowship.

References

[Arrow, 1963] K. Arrow. *Social choice and individual values*. New Haven: Cowles Foundation, 2nd edition, 1963. 1st edition 1951.

[Bartholdi, III *et al.*, 1989a] J. Bartholdi, III, C. A. Tovey, and M. A. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.

[Bartholdi, III *et al.*, 1989b] J. Bartholdi, III, C. A. Tovey, and M. A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(3):157–165, 1989.

[Ben-Or *et al.*, 1988] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 1–10. ACM Press, 1988.

[Benaloh, 1987] J. Benaloh. Secret sharing homomorphisms: Keeping shares of a secret secret. In *Advances in Cryptology - Proceedings of the 13th Annual International Cryptology Conference*

(CRYPTO), volume 263 of *Lecture Notes in Computer Science (LNCS)*, pages 251–260. Springer, 1987.

[Brandt and Sandholm, 2004] F. Brandt and T. Sandholm. (Im)possibility of unconditionally privacy-preserving auctions. In C. Sierra and L. Sonenberg, editors, *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 810–817. ACM Press, 2004.

[Brandt and Sandholm, 2005] F. Brandt and T. Sandholm. Decentralized voting with unconditional privacy. In S. Koenig and M. Wooldridge, editors, *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. ACM Press, 2005. To Appear.

[Chaum *et al.*, 1988] D. Chaum, C. Crépeau, and I. Damgård. Multi-party unconditionally secure protocols. In *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 11–19. ACM Press, 1988.

[Chaum, 1988] D. Chaum. Elections with unconditionally-secret ballots and disruption equivalent to breaking RSA. In *Proceedings of the 5th Eurocrypt Conference*, volume 330 of *Lecture Notes in Computer Science (LNCS)*, pages 177–182. Springer, 1988.

[Chor and Kushilevitz, 1989] B. Chor and E. Kushilevitz. A zero-one law for Boolean privacy. In *Proceedings of the 21st Annual ACM Symposium on the Theory of Computing (STOC)*, pages 62–72. ACM Press, 1989.

[Chor *et al.*, 1994] B. Chor, M. Geréb-Graus, and E. Kushilevitz. On the structure of the privacy hierarchy. *Journal of Cryptology*, 7(1):53–60, 1994.

[Conitzer and Sandholm, 2003] V. Conitzer and T. Sandholm. Universal voting protocol tweaks to make manipulation hard. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 781–788, 2003.

[Diffie and Hellman, 1976] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.

[Gibbard, 1973] A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41:587–602, 1973.

[Goldreich *et al.*, 1987] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *Proceedings of the 19th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 218–229. ACM Press, 1987.

[Hemaspaandra *et al.*, 1997] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP. *Journal of the ACM*, 44(6):806–825, 1997.

[Kushilevitz, 1989] E. Kushilevitz. Privacy and communication complexity. In *Proceedings of the 30th Symposium on Foundations of Computer Science (FOCS)*, pages 416–421. IEEE Computer Society Press, 1989.

[Pfitzmann and Waidner, 1992] B. Pfitzmann and M. Waidner. Unconditionally untraceable and fault-tolerant broadcast and secret ballot election. Hildesheimer Informatik-Berichte, Institut für Informatik, Universität Hildesheim, 1992.

[Rothe *et al.*, 2003] J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of the winner problem for Young elections. *Theory of Computing Systems*, 36(4):375–386, 2003.

[Satterthwaite, 1975] M. A. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10:187–217, 1975.

[Yao, 1982] A. C. Yao. Protocols for secure computation. In *Proceedings of the 23th Symposium on Foundations of Computer Science (FOCS)*, pages 160–164. IEEE Computer Society Press, 1982.

[Young and Levenglick, 1978] H. Young and A. Levenglick. A consistent extension of Condorcet's election principle. *SIAM Journal on Applied Mathematics*, 35(2):285–300, 1978.

A Necessity of Impossibility Conditions

Theorem 5 *There are SWFs that satisfy any three of these four properties: Non-dictatorship, Pareto-optimality, monotonicity, and private computability.*[9]

Proof:

- Paretian, monotonic, privately computable, but **not non-dictatorial**:
  Apparently, any dictatorial SWF satisfies these criteria. A dictatorial SWF can be "computed" privately by simply letting the dictator announce his preferences. All other preferences remain private.

- non-dictatorial, Paretian, privately computable, but **not monotonic**:
  Let $O = \{a, b\}$ and n be odd. We define $f(\succ_1, \succ_2, \ldots, \succ_n) = \succ$ so that

  $$\begin{array}{ll} a \succ b & \text{if } \left|\{i \mid i \in \{1, 2, \ldots, n\} \wedge a \succ_i b\}\right| \equiv 1 \pmod 2, \text{ and} \\ b \succ a & \text{otherwise.} \end{array}$$

  It can easily be seen that this SWF is non-dictatorial (in fact, it is symmetric) and Paretian. Private computability follows from the following theorem by Chor and Kushilevitz.

  Theorem 6 *[Chor and Kushilevitz, 1989] A* Boolean *function is privately computable if and only if it is of the form $f(x_1, x_2, \ldots, x_n) = B_1(x_1) \oplus B_2(x_2) \oplus \cdots \oplus B_n(x_n)$, where $B_i(x_i)$ are Boolean predicates and $\oplus$ is the Boolean exclusive-or operator.*

  Consider the Boolean function $f'(z_1, z_2, \ldots, z_n) = \bigoplus_{i=1}^{n} z_i$ where $z_i = 1$ if agent i prefers a over b and $z_i = 0$ otherwise. This function is equivalent to $f(\cdot)$ and is privately computable according to Theorem 6.

- non-dictatorial, monotonic, privately computable, but **not Paretian**:
  A SWF that satisfies these properties can be constructed with some efforts. An example is given in Table 2. The SWF given in Table 2 is clearly non-dictatorial. Verification of monotonicity is left to the inclined reader. $f(\cdot)$ is not Pareto-optimal since *e.g.*, $f(c \succ_1 a \succ_1 b, a \succ_2 b \succ_2 c) = b \succ c \succ a$ (all agents prefer a over b, but the collective preference is $b \succ a$). Private computability can be verified using the complete characterization of privately computable 2-ary functions given in [Kushilevitz, 1989].

[9] This statement also holds for SCFs. All counter-examples except the one given in Table 2 can be transferred directly.

$f(x_1, x_2)$	$a \succ_1 b \succ_1 c$	$a \succ_1 c \succ_1 b$	$b \succ_1 a \succ_1 c$	$b \succ_1 c \succ_1 a$	$c \succ_1 a \succ_1 b$	$c \succ_1 b \succ_1 a$
$a \succ_2 b \succ_2 c$	$a \succ b \succ c$	$a \succ b \succ c$	$a \succ b \succ c$	$b \succ c \succ a$	$b \succ c \succ a$	$b \succ c \succ a$
$a \succ_2 c \succ_2 b$	$a \succ c \succ b$	$a \succ c \succ b$	$a \succ c \succ b$	$a \succ c \succ b$	$a \succ c \succ b$	$a \succ c \succ b$
$b \succ_2 a \succ_2 c$	$b \succ a \succ c$	$b \succ a \succ c$	$b \succ a \succ c$	$b \succ a \succ c$	$b \succ a \succ c$	$b \succ a \succ c$
$b \succ_2 c \succ_2 a$	$a \succ b \succ c$	$a \succ b \succ c$	$a \succ b \succ c$	$b \succ c \succ a$	$b \succ c \succ a$	$b \succ c \succ a$
$c \succ_2 a \succ_2 b$	$c \succ a \succ b$	$c \succ a \succ b$	$c \succ a \succ b$	$c \succ a \succ b$	$c \succ a \succ b$	$c \succ a \succ b$
$c \succ_2 b \succ_2 a$	$c \succ b \succ a$	$c \succ b \succ a$	$c \succ b \succ a$	$c \succ b \succ a$	$c \succ b \succ a$	$c \succ b \succ a$

Table 2: Non-dictatorial, non-Paretian, monotonic, and privately computable SWF ($m = 3$, $n = 2$)

- non-dictatorial, Paretian, monotonic, but **not privately computable**:
  Almost all "reasonable" voting schemes fulfill these conditions. For example, a SWF implementing majority voting is not privately computable (due to embedded ORs). See Table 3 for an example with three voters (in order to avoid ties).

$f(x_1, x_2, x_3)$	$a \succ_1 b$	$b \succ_1 a$
$a \succ_2 b$, $a \succ_3 b$	$a \succ b$	$a \succ b$
$a \succ_2 b$, $b \succ_3 a$	$a \succ b$	$b \succ a$

Table 3: Majority voting ($m = 2$, $n = 3$)

■