Information and Computation **167**, 2–9 (2001)
doi:10.1006/inco.2000.3013, available online at http://www.idealibrary.com on IDEAL®

On Privacy and Partition Arguments[1]

Benny Chor[2] and Yuval Ishai[3]

Department of Computer Science, Technion—Israel Institute of Technology, Haifa, Israel
E-mail: benny@cs.technion.ac.il, yuval@dimacs.rutgers.edu

Received October 7, 1997

A function and $f(x_1, x_2, \ldots, x_n)$ is said to be *t-private* if there exists a (randomized) communication protocol for computing f, such that no coalition of at most t participants can infer any additional information from the execution of the protocol other than what follows from their inputs and the value of f. It is known that every n-argument function f defined over finite domains can be computed $\lfloor \frac{n-1}{2} \rfloor$-privately. The classes of 1-private two-argument functions and of t-private *Boolean* functions admit relatively simple characterizations. In contrast, the general question of characterizing the class of t-private functions of n arguments is still open. The only technique that appears in the literature for proving non-t-privacy of a function $f(x_1, x_2, \ldots, x_n)$ over a finite domain, where $n \geq 3$ and $\lceil \frac{n}{2} \rceil \leq t \leq n-1$, uses a reduction to the two-party case via a partition argument. A necessary condition for f being t-private is that for every partition $(S; \bar{S})$ of the parties $\{1, 2, \ldots, n\}$ such that both $|S| \leq t$ and $|\bar{S}| \leq t$, the two-argument function obtained by viewing f as a function of $\{x_i\}_{i \in S}$ and $\{x_i\}_{i \in \bar{S}}$ is 1-private. The question whether the use of such partition reductions together with the two-party characterization is powerful enough to characterize privacy in the multiparty case was raised as an open problem in previous works. These works also exhibit an affirmative answer for specific classes of functions. We answer this question negatively. We show that even if more general partition reductions are used, in which the n parties are partitioned into k sets ($2 \leq k \leq n-1$) rather than just two, those reductions are still too weak to characterize privacy. On the other hand, we show that increasing the number of sets k does give some extra characterization power.

Key Words: private distributed computation; partition arguments.

1. INTRODUCTION

A set of n parties, each holding an input value x_i from some domain X_i, wishes to distributively compute a given function $f(x_1, x_2, \ldots, x_n)$. The participants communicate via a complete network of reliable and secure channels (no eavesdropping). The participants are honest—they send messages according to the prescribed randomized protocol for f. However, a subset of the participants (a coalition) might get together after the execution of the protocol in an attempt to infer *additional information* on the inputs of non-coalition parties. Additional information is any information that does not follow from the value of the function, $f(\vec{x})$, and the inputs of the coalition parties. A protocol $\mathcal{F}$ is called *t-private* if no coalition containing at most t parties can get any additional information from $\mathcal{F}$'s execution.

Private multiparty computation of general functions, under various models and assumptions, has been the subject of a considerable amount of work, originating from [2, 4, 8, 11]. The model considered here is a minimalistic one, referred to as the model of "honest but curious" parties, in the *information-theoretic* setting. Stronger adversarial scenarios, including Byzantine [2, 4] and adaptive [3] adversaries, have been studied in the literature. Negative results on private computation in our model hold in the more adversarial (information-theoretic) models as well.

The seminal works of [2, 4] showed that all n-argument functions over finite domains X_i can be computed $\lfloor \frac{n-1}{2} \rfloor$-privately. In [5] it is shown that there exists a dense privacy hierarchy: for any $\lfloor \frac{n-1}{2} \rfloor \leq t \leq n-2$ there exists an n-argument function which is t-private (i.e., can be computed by a t-private protocol) but is not $(t+1)$-private. In the works of [1, 10] a complete characterization of 1-private two-argument functions is given. The general question of characterizing the class of t-private functions

[1] A preliminary version of this work appeared in the Proceedings of the Fourth Israel Symposium on Theory of Computing and Systems (ISTCS 1996).

[2] Supported by the fund for promotion of research at the Technion.

[3] Current address: DIMACS Center, Rutgers University, 96 Frelinghuysen Road, Piscataway, NJ 08854-8018.

0890-5401/01 $35.00

of n-argument, for any $n \geq 3$ and $\lceil \frac{n}{2} \rceil \leq t \leq n-1$, is still open. The only technique which appears in the literature for proving non-t-privacy of functions with $n \geq 3$ arguments uses a reduction to the two-party case, via a partition argument [5–7]. If f is t-private, where $\lceil \frac{n}{2} \rceil \leq t \leq n-1$, then for every partition $(S; \bar{S})$ of the parties $\{1, 2, \ldots, n\}$ such that $|S|, |\bar{S}| \leq t$, the two-argument function obtained by viewing f as a function of $\{x_i\}_{i \in S}$ and $\{x_i\}_{i \in \bar{S}}$ is 1-private.

In [6], in the course of proving a characterization of t-private Boolean functions, it is shown that, with respect to *Boolean* functions, such a partition argument always suffices for proving non-t-privacy. That is, for any n-argument Boolean function which is not t-private, there exists a suitable partition of its variables into two sets such that the induced two-argument function is not 1-private. A similar result holds for a certain class of symmetric functions [7]. The question whether such an argument always suffices for proving non-t-privacy in the general case is raised as an open problem in [5, 7], where extensive use of the partition technique has been made.

In this work we provide a negative answer to this question: partitions *cannot* always be used to prove non-privacy. The proof starts in Section 4, where we give a necessary condition for the $(n-1)$-privacy of any n-argument function that generalizes the necessary (and sufficient) condition of the two-party case. This result opens up the possibility of proving non-privacy of an n-argument function by partitioning its variables into $k > 2$ sets and using the k-party necessary condition to show that the induced k-argument function is not private.

Now how powerful are those generalized partition arguments? Is it possible that their use together with a characterization of all private k-argument functions, where k is bounded by some *fixed* K, is universal for non-privacy proofs? Section 5 addresses these questions. We show that the power of generalized partition arguments is in fact rather limited. We construct an n-argument function g_n which is *not* fully private (i.e., not $(n-1)$-private), but every nontrivial partition of its variables induces a fully private function. Putting it in another way, the non-privacy of g_n is very "fragile." It collapses whenever any two (or more) of the parties unite. This means that the non-privacy of g_n cannot be reduced via partition arguments to the non-privacy of some function of $k < n$ variables. On the other hand, we show that increasing the number of sets in partitions does give some extra power. For any $k \geq 3$ we construct n-argument functions ($n > k$) which, for some $t < n-1$, can be proven to be non-t-private using partition into k sets, but cannot be proven to be non-t-private using partition into a smaller number of sets.

2. MODEL AND DEFINITIONS

In this section we define the model of distributed computation that is used, give a formal definition of privacy in this model, and introduce some notation.

The system consists of a complete synchronous network of n *honest* parties $P_1, P_2, \ldots, P_n$ with secure and reliable point-to-point communication (no eavesdropping). (By saying that the parties are honest it is meant that they send messages according to a predetermined protocol $\mathcal{F}$.) At the beginning of an execution, each party P_i has an input $x_i \in X_i$ (no probability space is associated with the inputs). In addition, each party has a random independent input r_i taken from a source of randomness R_i. The parties wish to compute some given function $f(\vec{x})$. To this end, they exchange messages as prescribed by the protocol $\mathcal{F}$. Messages are sent in rounds, where in each round every party can send a *message* to every other party. The protocol's definition determines which message a party sends in the kth round as a function of its input, its random input, the messages it received so far, and the identity of the receiver. At the end of the computation, one of the parties (say P_1) outputs the value $f(\vec{x})$.

The *communication* passed in the network when the parties have inputs $\vec{x}$ and random inputs $\vec{r}$ is denoted $\vec{C}(\vec{x}, \vec{r})$ and is represented by a vector of strings, whose kth entry includes the concatenation of all messages sent during kth round of the execution of the protocol, parsed according to sender and receiver. For a given communication $\vec{C}$ and a subset T of the n parties, $\vec{C}_T$ denotes the restriction of $\vec{C}$ to the messages sent or received by the parties in T (i.e., $\vec{C}_T$ excludes from $\vec{C}$ all messages sent internally between the parties of T).

Let $\mathcal{F}$ be a randomized protocol which computes $f(\vec{x})$ (with no error). We say that a coalition (i.e., a set of parties) T *does not learn any additional information from the execution of* $\mathcal{F}$ (other than what follows from its input and $f(\vec{x})$) if the following holds: For every two inputs $\vec{x}$, $\vec{y}$ that agree in their T

entries (i.e., $\forall i \in T : x_i = y_i$) and satisfy $f(\vec{x}) = f(\vec{y})$, and for every choice of random inputs $\{r_i\}_{i\in T}$, the messages seen by T are identically distributed. That is, for every communication $\vec{C}$,

$$Pr(\vec{C}_T \mid \vec{x}, \{r_i\}_{i\in T}) = Pr(\vec{C}_T \mid \vec{y}, \{r_i\}_{i\in T}),$$

where the probability space is over all random inputs in T, namely $\{r_i\}_{i\in \bar{T}}$ (each r_i is distributed according to R_i and they are all independent). We say that $\mathcal{F}$ is *t-private* if any coalition T, which contains at most t parties, does not learn any additional information from the execution of the protocol $\mathcal{F}$. We say that a function f is *t-private* if there exists a t-private protocol computing f. We say that the n-argument function f is *fully private* if it is $(n-1)$-private.

3. PROVING NON-PRIVACY VIA PARTITION ARGUMENTS

In this section we state the Partition Lemma, which allows the reduction of proving the non-privacy of an n-argument function $f(x_1, x_2, \ldots, x_n)$ to proving the non-privacy of a k-argument function $f'(y_1, y_2, \ldots, y_k)(k < n)$. Previous works [5–7], relying only on the two-party characterization of privacy, restrict the statement of the lemma to the case where $k = 2$. The following more general form is an immediate generalization of the special case used in those works.

DEFINITION 3.1. A *k-partition* of a set of n parties (or variables) $\{1, 2, \ldots, n\}$ is an ordered partition of the set into k non-empty, mutually disjoint sets and is denoted $(S_1; S_2; \ldots; S_k)$.

DEFINITION 3.2. Given an n-argument function $f : X_1 \times X_2 \times \cdots \times X_n \to Z$ and a k-partition $(S_1; S_2; \ldots; S_k)$ of the parties, denote by Y_i the Cartesian product of the X_j with $j \in S_i$, and let $f' : Y_1 \times Y_2 \times \cdots \times Y_k \to Z$ be the function obtained by viewing f as a k-argument function; that is, f' is defined by $f'(\{x_i\}_{i\in S_1}, \{x_i\}_{i\in S_2}, \ldots, \{x_i\}_{i\in S_k}) = f(x_1, x_2, \ldots, x_n)$. Given an n-argument function f, the k-argument function f' will be referred to as the function *induced by the partition* $(S_1; S_2; \ldots; S_k)$.

LEMMA 3.1 (The Partition Lemma [6]). *Suppose $f : X_1 \times X_2 \times \cdots \times X_n \to Z$ is t-private. Then for every k-partition $\Pi = (S_1; S_2; \ldots; S_k)$ and every t' such that the size of the union of any t' sets S_i does not exceed t, the induced k-argument function f' is t'-private.*

We remark that the cases of interest in the above lemma are $t \geq \lceil \frac{n}{2} \rceil$ and $t' \geq \lceil \frac{k}{2} \rceil$. The simple proof of the lemma is by direct simulation: an original t-private n-party protocol may be simulated by the k parties, where each "new" party P'_i takes the role of all original parties belonging to S_i, embedding internal communications into the definition of the new protocol. The t'-privacy of the new protocol follows from the t-privacy of the original protocol and from the sizes of coalitions induced by Π.

Note that in using partition reductions to prove the non-t-privacy of some n-argument function, the cardinality k of the partitions can be restricted to range from 2 to $n-1$. If $k = 1$, then the induced function is always fully private (as a 1-argument function), so non-privacy cannot be proven at all. If $k = n$, on the other hand, then the original problem is reduced to itself, which is not much of a help either.

4. NECESSARY CONDITION FOR FULL PRIVACY

In this section we state and prove a necessary condition for the full privacy of any n-argument function. This condition generalizes the necessary condition for the two-party case [1, 10].

In order to simplify (and clarify) the statement and proof of the following lemma, we do not use the most general form possible. We restrict our attention to functions mapping from $[m]^n$ (where $[m]$ denotes the set $\{1, 2, \ldots, m\}$) into some arbitrary range Z. The lemma can be applied to functions of arbitrary (even infinite) domains and further strengthened using a generalization of the concept of "forbidden rectangle" from [10] to the multiparty case.

x_1	x_2	x_3	$g(\vec{x})$
*	1	1	1
2	*	2	2
3	3	*	3
any other $\vec{x}$			$\vec{x}$

FIG. 1. The function g.

DEFINITION 4.1. Given a vector $\vec{u} \in [m]^n$ and $d \in [m]$, denote by $\vec{u}|_{i \leftarrow d}$ the vector obtained by replacing the value of the ith coordinate of $\vec{u}$ with d. A function $f : [m]^n \to Z$ is called *non-separable at its* ith *coordinate* if there exists a vector $\vec{u}$, such that f attains the same value on all m vectors $\vec{u}|_{i \leftarrow d}, 1 \leq d \leq m$. The function f is called *non-separable* if it is non-separable at each of its n coordinates.

Any *constant* function is clearly non-separable. As a less trivial example, Fig. 1 describes a specific function g, whose domain is $[3]^3$. This function is nonseparable, as $g(1,1,1) = g(2,1,1) = g(3,1,1)$, $g(2,1,2) = g(2,2,2) = g(2,3,2)$, and $g(3,3,1) = g(3,3,2) = g(3,3,3)$.

LEEMA 4.1. *If $f : [m]_n \to Z$ is a non-constant non-separable function, then f is not fully private.*

Proof. The following proof generalizes the proof for the two-party case appearing in [10]. Suppose toward a contradiction that $\mathcal{F}$ is an $(n-1)$-private protocol which computes f. We may assume, without loss of generality, that the parties "take turns" in sending messages; i.e., party P_i gets to send messages only in rounds $i + nj$, $j = 0, 1, 2, \ldots$.

PROPOSITION 4.1. *For every two inputs $\vec{x}, \vec{y} \in [m]^n$ and communication $\vec{C}$, $Pr(\vec{C} \mid \vec{x}) = Pr(\vec{C} \mid \vec{y})$ (where the probabilities here and in the following proof are taken over the choices of all random inputs).*

Proof. The intuition behind the proof is simple: In each round, if the active party P_i has no information on the inputs of the other parties, the distribution of the messages it sends should not depend on its input. Because otherwise, the nonseparability of f at the ith coordinate implies that for some possible combination of the other parties' inputs, the coalition of all other parties will violate the privacy of P_i. Since initially no party has any information on the other parties' inputs, this condition is preserved throughout the execution of the protocol. This argument applies to the final round as well, forcing errors in the final output. We now formalize this idea.

Let $\vec{C}$ be any communication vector, and let $\vec{C}_k$ denote the restriction of $\vec{C}$ to its first k entries (i.e., $\vec{C}_k$ includes all messages of $\vec{C}$ sent during the first k rounds). Let C_k denote the restriction of $\vec{C}$ to its kth entry alone. We prove, by induction on k, that for any two inputs $\vec{x}, \vec{y}$, $Pr(\vec{C}_k \mid \vec{x}) = Pr(\vec{C}_k \mid \vec{y})$.

The case of $k = 0$ is trivial (both probabilities are 1). We now assume that the claim holds for $k-1$ and prove that it holds for k. Assume that P_i is the active party in the kth round. By the inductive assumption, $Pr(\vec{C}_{k-1} \mid \vec{z})$ is independent of the input $\vec{z}$. If this probability is 0, then also $Pr(\vec{C}_k \mid \vec{x}) = Pr(\vec{C}_k \mid \vec{y}) = 0$. We may thus assume from now on that (for all inputs $\vec{z}$) $Pr(\vec{C}_{k-1} \mid \vec{z}) > 0$.

Let $\vec{x}, \vec{y}$ be any two input vectors, and $\vec{u}, \vec{v}$ be two vectors such that $u_i = x_i, v_i = y_i$, for all j other than $i, u_j = v_j$, and $f(\vec{u}) = f(\vec{v})$ (the existence of such $\vec{u}, \vec{v}$ is guaranteed by the assumption that f is nonseparable). For any communication $\vec{C}$, the combined view of a coalition, consisting of all parties except P_i, includes all messages of $\vec{C}$. Thus, the requirement that $\mathcal{F}$ is $(n-1)$-private implies that $Pr(\vec{C} \mid \vec{u}) = Pr(\vec{C} \mid \vec{v})$. Since this is true for *any* communication $\vec{C}$, we have in particular that $Pr(C_k \mid \vec{u}, \vec{C}_{k-1}) = Pr(C_k \mid \vec{v}, \vec{C}_{k-1})$ (note that $Pr(\vec{C}_{k-1}) > 0$ by an assumption made earlier).

Relying on the fact[4] that for any $\vec{z}$ the probability $Pr(C_k \mid \vec{z}, \vec{C}_{k-1})$ depends only on z_i, the input of the ith party, we have that

$$\begin{aligned} Pr(C_k \mid \vec{x}, \vec{C}_{k-1}) &= Pr(C_k \mid \vec{u}, \vec{C}_{k-1}) \\ &= Pr(C_k \mid \vec{v}, \vec{C}_{k-1}) \\ &= Pr(C_k \mid \vec{y}, \vec{C}_{k-1}), \end{aligned}$$

[4] A formal proof of this basic property of general randomized protocols appears in [9].

and finally that

$$\begin{aligned} Pr(\vec{C}_k \mid \vec{x}) &= Pr(C_k \mid \vec{x}, \vec{C}_{k-1}) Pr(\vec{C}_{k-1} \mid \vec{x}) \\ &= Pr(C_k \mid \vec{y}, \vec{C}_{k-1}) Pr(\vec{C}_{k-1} \mid \vec{y}) \\ &= Pr(\vec{C}_k \mid \vec{y}), \end{aligned}$$

as required. ■

In particular we conclude that the distribution of messages sent in the final round of $\mathcal{F}$, which are assumed to contain the value of f, is the same for every input. Since f is non-constant, this contradicts the requirement that $\mathcal{F}$ computes f (even if we allow the protocol to err with some positive probability smaller than $\frac{1}{2}$). This concludes the proof of Lemma 4.1. ■

Remark. The converse of Lemma 4.1 does not hold; this necessary condition is not *sufficient* for full privacy.[5] For instance, consider the function $f : \{0,1\}^3 \to \{0,1\}^3 \cup \{1\}$ such that $f(\vec{x})$ equals 1 for $\vec{x} = 000, 110, 001$ and equals $\vec{x}$ otherwise. The function f can be shown to be not fully private via a partition argument,[6] but is clearly not non-separable.

5. MAIN RESULTS

In this section we prove two results, both dealing with the power of partition reductions. The first (and more significant) shows that this power is rather limited: partitions cannot always be used to prove non-privacy. The second result shows that while this power is limited, it gradually increases as the number of sets in partitions is allowed to grow. Both results can be obtained as special cases of the following lemma.

LEMMA 5.1. *Let $k \geq 3, \ell \geq 1$ be two integers. There exists a $k\ell$-argument function $f_{k,\ell}$ for which there is some k-partition inducing a k-argument function which is not fully private, but for every $k' < k$, all k'-partitions induce fully private k'-argument functions.*

Proof. Set $n = k\ell$. Define k sets D_i as follows: $D_1 = \{1, 2, \ldots, \ell\}$, $D_2 = \{\ell+1, \ell+2, \ldots, 2\ell\}$, $\ldots$, $D_k = \{n-\ell+1, n-\ell+2, \ldots, n\}$, and let d_i denote the minimal element of D_i. Define an n-argument function $f_{k,\ell} : [k]^n \to [k] \cup [k]^n$ as follows: If there exists $i \in [k]$ such that for all j in $[n] \backslash D_i$, $x_j = i$ (since $k \geq 2$ there can be at most one such i), then let $f_{k,\ell}(\vec{x}) = i$; otherwise let $f_{k,\ell}(\vec{x}) = \vec{x}$. Note that for any $i \in [k]$, whether $f_{k,\ell}(\vec{x}) = i$ depends only on the values assigned to the variables in $[n] \backslash D_i$. The instance $f_{3,2}$ is described in Fig. 2. We show that such an $f_{k,\ell}$ meets the requirements of Lemma 5.1.

PROPOSITION 5.1. *There exists a k-partition of the n variables inducing a k-argument function which is not fully private.*

Proof. The k-partition $(D_1; D_2; \ldots; D_k)$ induces a k-argument function f' which is both non-constant and non-separable. (Non-separability at the ith coordinate follows from the "all i" input vector.) By Lemma 4.1, f' is not fully private. ■

PROPOSITION 5.2. *For every $k' < k$, every k'-partition of the n variables induces a fully private k'-argument function.*

Proof. Let $\Pi = (S_1; S_2; \ldots; S_{k'})$ be a k'-partition of the $n = k\ell$ variables. Since $k' < k$ there must be some set $S_i \in \Pi$ whose size is greater than ℓ, so without loss of generality we assume $|S_1| > \ell$. Define $A = \{h : S_1 \cap D_h \neq \emptyset\}$. Since $|S_1| > \ell$, the set A must contain at least two elements.[7] In other words,

[5] A slightly generalized formulation of the non-separability condition, as is done for the two-party case in [1, 10] would make it sufficient for two-party privacy; however, the three-argument function defined below is not non-separable even under the generalized condition.

[6] The two-argument function induced by the partition ({1, 2}, {3}) contains an "embedded OR" (cf. [10]), and hence, by the two-party characterization, it is not private.

[7] The set A depends on the partition Π and does not depend on the actual input $\vec{x}$.

x_1	x_2	x_3	x_4	x_5	x_6	$f_{3,2}(\vec{x})$
*	*	1	1	1	1	1
2	2	*	*	2	2	2
3	3	3	3	*	*	3
any other $\vec{x}$						$\vec{x}$

FIG. 2. The function $f_{3,2}$.

party P_1 holds inputs with indices in at least two different sets D_i. Denote by f the function induced by the partition Π. We describe a (deterministic) fully private protocol $\mathcal{F}$ computing f, in which each party P_i, $1 \le i \le k'$, holds the input variables with indices in S_i. Intuitively, the protocol will specify a "cautious" sequence of partial disclosures of information, guaranteeing that the information disclosed in any execution can be inferred from the value $f(\vec{x})$.

Protocol $\mathcal{F}$:

Round 1: P_1 finds the set $R \subset \{1, 2, \ldots, k\}$ of all values j' such that P_1's inputs rule out the possibility that $f(\vec{x}) = j'$. It then announces a value $a \in R$, which is determined as follows:

1. If $|A| > 2$ or $A \subseteq R$, then a is the minimal element in R. (Proof of correctness will show that R is nonempty.)
2. Otherwise, a is the minimal element in $R \backslash A$. (Proof of correctness will show that in this case $R \backslash A$ is nonempty.)

Round 2: The party holding input variable x_{d_a} (the element with minimal index in D_a) announces its value, b.

Round 3: Parties announce the values of all variables with indices from $[n] \backslash D_b$ (i.e., each party P_i announces the values of the variables x_j such that $j \in S_i \cap ([n] \backslash D_b)$).

Round 4: If *all* variables revealed in the previous round are equal to b, P_1 outputs "$f = b$," and the protocol terminates. Otherwise all parties announce the values of all variables not yet revealed.

Round 5: P_1, knowing all inputs, outputs "$f = \vec{x}$."

We now prove that $\mathcal{F}$ computes f correctly and with full privacy.

Correctness: We first show that the choice of a in Round 1 is well defined. As illustrated by Fig. 2, each input value x_i is consistent with at most two values for f taken from $[k]$, namely either $f = x_i$ or $f = j$, where $i \in D_j$. Since $k \ge 3$ it follows that P_1 can rule out at least one value $a \in [k]$ from being equal to $f(\vec{x})$, and so R is always nonempty. In order to prove that the choice of a is well defined in the second case of Round 1 as well, it suffices to show that if $R \backslash A$ is empty then the first case must hold. Suppose that $|A| = 2$ (otherwise the first case obviously holds) and $R \backslash A = \emptyset$. Since $k \ge 3$, there exists some $j_0 \in [k] \backslash A$, and such j_0 cannot be in R (otherwise $R \backslash A$ would be nonempty). By the definition of f, all variables held by P_1 must be equal to j_0. Since $j_0 \notin A$, this implies that each $j' \in A$ can be ruled out from being equal to the value of f. It follows that $A \subseteq R$, and the first case holds.

The protocol always terminates with an output. We will show that this output is indeed equal to the value of f. Since $f(\vec{x}) \ne a$, and $x_{d_a} = b$, when the protocol reaches Round 4 it must be the case that $f(\vec{x})$ is either b or $\vec{x}$. Now, if "$f = b$" is output in Round 4, this must be because $x_i = b$ for all $i \notin D_b$, and so by definition of $f_{k,\ell}$ we have $f(\vec{x}) = b$. If $x_i \ne b$ for some $i \notin D_b$ then by definition $f(\vec{x}) = \vec{x}$, so the value of f output in Round 5 is correct.

Privacy: By the definition of full privacy, it suffices to show that for every $j \in [k]$ the protocol yields identical messages when run on all inputs $\vec{x}$ such that $f(\vec{x}) = j$. (Inputs such that $f(\vec{x}) = \vec{x}$ impose no privacy constraint.) Suppose $f(\vec{x}) = j$, where $j \in [k]$. By the protocol, the value of a announced by P_1 in Round 1 depends on R. So it may seem that this message could reveal some information on P_1's inputs. We will begin by showing that (given the fixed k'-partition Π and the set A associated with it) the value of a chosen in Round 1 depends only on j. This implies that the Round 1 message does not violate the privacy requirements.

Case 1. $|A| > 2$. Since A contains at least three elements, then for every $j' \in [k] \backslash \{j\}$, P_1 holds some input variable x_i whose index i satisfies $i \notin D_j \cup D_{j'}$. Since $f(\vec{x}) = j$, the value of such x_i must equal j. This allows P_1 to rule out the possibility that $f(\vec{x}) = j'$. It follows that in this case $R = [k] \backslash \{j\}$.

Case 2. $|A| = 2$ and $j \in A$. In this case $j \in A \backslash R$, so $A \not\subseteq R$ and P_1 acts according to the second case in Round 1. Let $A = \{j, \tilde{j}\}$. Then P_1 holds some input variable x_i whose index i satisfies $i \in D_{\tilde{j}}$. By the definition of f, since $f(\vec{x}) = j$, the equality $x_i = j$ must hold. Therefore, for every $j' \in \{1, 2, \ldots, k\} \backslash A$, the inputs held by P_1 rule out the possibility that $f(\vec{x}) = j'$. It follows that in this case $R \backslash A = [k] \backslash A$, implying that $a = \min([k] \backslash A)$.

Case 3. $|A| = 2$ and $j \notin A$. Let $A = \{j_1, j_2\}$, then P_1 holds two input variables x_{i_1}, x_{i_2} whose indices i_1, i_2 satisfy $i_1 \in D_{j_1}, i_2 \in D_{j_2}$. By the definition of f, since $f(\vec{x}) = j$, the equality $x_{i_1} = x_{i_2} = j$ must hold. Therefore, for both elements of A, the inputs held by P_1 rule out the possibility that the value of f equals this element. This means that $A \subseteq R$ and P_1 acts according to the first case in Round 1. As A contains two elements from $[k]$ that are different than j, the same reasoning as that in Case 1 implies that in this case $R = [k] \backslash \{j\}$.

Now whether Case 1, 2, or 3 holds is completely determined by A and j, the value of f. Given each case, the value a is again determined by A and j. Therefore the value a sent by P_1 at Round 1 does not violate the privacy requirements.

It can be easily verified that for any $\vec{x}$ with $f(\vec{x}) = j$, the value of all messages sent in rounds 2–4 is equal to j; these messages are sent by the same set of parties, and the protocol terminates at Round 4. This establishes the full privacy of $\mathcal{F}$. ■

Propositions 5.1 and 5.2 directly imply Lemma 5.1. ■

We can now prove the two main results.

THEOREM 5.1. *The converse of the Partition Lemma does not hold. More specifically, for every $n \geq 3$ there exists an n-argument function g_n which is not fully private, but every k-partition of its variables to fewer than n parts ($2 \leq k \leq n-1$) induces a fully private function.*

Proof. Define $g_n = f_{n,1}$ (Fig. 1 describes the instance g_3). Lemma 5.1 implies that g_n (as the function induced by its own n-partition) is not fully private, but every k-partition of its variables, $2 \leq k \leq n-1$, induces a fully private function. ■

THEOREM 5.2. *The power of partition reductions strictly increases as the number of sets in the partitions is allowed to grow. More formally, for any $k \geq 3$ there exist n-argument functions ($n > k$) which, for some t, can be proven to be non-t-private using partition into k sets, but cannot be proven to be non-t-private using partition into a smaller number of sets.*

Proof. Set $f = f_{k,\ell}$, where $\ell \geq 2$. The proof of Proposition 5.1 exhibits a partition of the $k\ell$ variables of f into k sets of size ℓ each, such that the induced k-argument function is not $(k-1)$-private. Since each union of $k-1$ sets from this partition is of size $(k-1)\ell$, the Partition Lemma implies that f is not $(k-1)\ell$-private. In addition, Lemma 5.1 states that for every $k' < k$, all k'-partitions induce fully private functions, and thus such partitions cannot be used to prove non-privacy of $f_{k,\ell}$. We conclude that such f satisfies the requirements of Theorem 5.2 with $n = k\ell$ and $t = (k-1)\ell = (1 - \frac{1}{k})n$. ■

6. OPEN PROBLEMS

As indicated by Theorem 5.2, the combination of a generalized Partition Lemma with our necessary condition for full privacy is stronger than the combination of two-partitions and the two-party characterization, used in previous works [5–7]. This gives hope to characterize wider natural classes of functions using the generalized techniques.

The general problem of characterizing the t-private functions is still wide open. Our necessary condition for full privacy is not a sufficient one, and this leaves even the problem of characterizing *full* privacy open. To gain better understanding of the information-theoretic notion of privacy, one has to develop new proof techniques for proving non-t-privacy, especially when $t < n-1$.

Finally, while this work focuses on existential results, a potentially interesting question is that of finding the maximal range size M, such that partition reductions are universal for proving non-privacy of functions mapping to $[M]$. The Boolean case characterization of [6] shows that such maximal M is it least 2. Is this bound tight?

ACKNOWLEDGMENTS

We thank Moshe Vardi for raising a question which has led to an extension of the original result, Niv Gilboa and Eyal Kushilevitz for helpful discussions, and an anonymous referee for carefully commenting on an earlier version of this manuscript.

REFERENCES

1. Beaver, D. (1989), "Perfect Privacy for Two Party Protocols," Technical Report TR-11-89, Harvard University.
2. Ben-or, M., Goldwasser, S., and Wigderson, A. (1988), Completeness theorems for non-cryptographic fault-tolerant distributed computation, in "Proc. of 20th STOC," pp. 1–10.
3. Canetti, R., Feige, U., Goldreich, O., and Naor, M. (1996), Adaptively secure multi-party computation, in "Proc. of 28th STOC," pp. 639–648.
4. Chaum, D., Crépeau, C., and Damgård, I. (1988), Multiparty unconditionally secure protocols, in "Proc. of 20th STOC": pp. 11–19.
5. Chor, B., Gereb-Graus, M., and Kushilevitz, E. (1994), On the structure of the privacy hierarchy, *J. Cryptol.* **7**, 53–60.
6. Chor, B., and Kushilevitz, E. (1991), A zero–one law for Boolean privacy, *SIAM J. Discrete math.* **4**(1), 36–47. [Early version appears in "Proc. of 21th STOC," 1989, pp. 62–72]
7. Chor, B., and Shani, N. (1995), The privacy of dense symmetric functions, *Comput. Complex.* **5**, 43–59.
8. Goldreich, O., Micali, S., and Wigderson, A. (1987), How to play any mental game (extended abstract), in "Proc. of 19th STOC," pp. 218–229.
9. Ishai, Y., and Kushilevitz, E. On strong vs. weak privacy, manuscript.
10. Kushilevitz, E. (1992), Privacy and communication complexity. *SIAM J. Discrete Math.* **5**(2), 273–284. [Early version appears in "Proc. of 30th FOCS," pp. 416–421]
11. Yao, A.C. 1982, Protocols for secure computations (extended abstract), in "Proc. of 23rd FOCS," pp. 160–164.