A Communication–Privacy Tradeoff for Modular Addition

Benny Chor *

Eyal Kushilevitz†

Dept. of Computer Science
Technion
Haifa 32000, Israel

Aiken Computation Lab.
Harvard University
Cambridge, MA 02138

Abstract

In this note we consider the following problem: A set of n parties, each holding an input value $x_i \in \{0, 1, \ldots, m-1\}$, wishes to distributively compute the sum of their input values modulo the integer m, (i.e, $\sum_{i=1}^{n} x_i \bmod m$). The parties wish to compute this sum *t–privately*. That is, in a way that no coalition of size at most t can infer any additional information, other than what follows from their input values and the computed sum.

We present an oblivious *protocol* which computes the sum t–privately, using $n \cdot \lceil (t+1)/2 \rceil$ *messages*. This protocol requires fewer messages than the known private protocols for modular addition. Then, we show that this protocol is in a sense optimal, by proving a tight *lower bound* of $\lceil n \cdot (t+1)/2 \rceil$ messages for any *oblivious* protocol that computes the sum t–privately.

Keywords: private computation, message complexity, modular sum.

1 Introduction

A set of n parties, each holding an input value $x_i \in \{0, 1, \ldots, m-1\}$, wishes to distributively compute the function $sum(\vec{x}) \stackrel{\triangle}{=} \sum_{i=1}^{n} x_i \bmod m$. The participants communicate via a complete network of 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, $sum(\vec{x})$, and the inputs of the coalition parties. (Clearly, a coalition can always deduce from $sum(\vec{x})$ and the inputs of coalition members, the sum of inputs of non-coalition members.)

*Research supported by US–Israel BSF Grant 88-00282. e-mail benny@cs.technion.ac.il

†Research supported by ONR-N0001491-J-1981 and NSF-CCR-90-07677. e-mail eyalk@das.harvard.edu

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 computations in this model are the subject of previous research [2, 1, 3, 4, 5, 7]. In particular, the importance of computing the *sum* privately, was exemplified in [3, 4, 5]. An n–private protocol for computing this function was presented in [2]. The protocol is as follows:

1. Every party P_i chooses independently with uniform distribution $n-1$ elements $z_{i,1}, z_{i,2}, \ldots, z_{i,n-1}$ in $\{0, 1, ..., m-1\}$. Every party P_i computes $z_{i,n}$ such that $x_i = \sum_{j=1}^{n} z_{i,j} \bmod m$ and sends $z_{i,j}$ to P_j.
2. Every party P_i computes $w_i = \sum_{j=1}^{n} z_{j,i}$. It sends w_i to P_n.
3. The party P_n computes $sum(\vec{x}) = \sum_{i=1}^{n} w_i \bmod m$.

The correctness of this protocol follows from the fact that $\sum_{i=1}^{n} x_i \bmod m = \sum_{i=1}^{n} w_i \bmod m$. The n-privacy of the protocol follows from the independence of every $n-1$ elements in $\{z_{i,1}, z_{i,2}, \ldots, z_{i,n}\}$. A modification of this protocol, in which every participant distributes its input only to $P_{n-t}, ..., P_n$, is t–private. The number of messages required by the modified protocol is $(t+1) \cdot n + t$.

We present a protocol which computes the sum while sending only $n \cdot \lceil (t+1)/2 \rceil$ messages. This improves, by a factor of 2, the message complexity of known t-private protocols for this problem. All known private protocols, including the one presented in this paper, are *oblivious* protocols. That is, the decision whether party P_i sends a message to party P_j at a given round k is determined by i, j, and k, and does not depend on the input and the random coins. We prove an $\lceil n \cdot (t+1)/2 \rceil$ *lower bound* on the number of messages required by any *oblivious* protocol for computing the sum t–privately. This shows a tight tradeoff between privacy and communication for the modular sum problem. Essentially the same lower bound (but not the matching upper bound) was independently proved by Franklin and Yung [6].

2 Definitions

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

The system consists of a complete synchronous network of n *honest* parties $P_1, P_2, ..., P_n$ 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 $x_i \in \{0, 1, ..., m-1\}$ (no probability space is associated with the inputs). In addition, each party has a random input r_i taken from a source of randomness R_i. The parties wish to compute $sum(\vec{x}) = \sum_{i=1}^{n} x_i \bmod m$. 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. Each message a party sends in the $k-th$ round is determined using its input, its random input, the messages it received so far, and the identity of the receiver. In oblivious protocols, the decision whether P_i should send a message to P_j at round

k is determined by the protocol, regardless of the input or the random input. At the end of the computation, one of the parties (say P_n) outputs $sum(\vec{x})$.

The *communication* passed in the network when the parties have inputs $\vec{x}$ and random inputs $\vec{r}$ is denoted $S(\vec{x}, \vec{r})$. For any $T \subset \{P_1, P_2, \ldots, P_n\}$, S_T denotes the view of the parties in T of the communication S; namely, the communication S with the exception of messages sent between parties in $\bar{T}$.

Let $\mathcal{F}$ be a randomized protocol which computes $sum(\vec{x})$. We say that a coalition (i.e. a set of parties) T *does not learn any additional information* (other than what follows from its input and $sum(\vec{x})$) from the execution of $\mathcal{F}$ 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 $sum(\vec{x}) = sum(\vec{y})$, and for every choice of random inputs $\{r_i\}_{i \in T}$, the messages passed between T and $\bar{T}$ are identically distributed. That is, for every communication S,

$$Pr(S_T | \vec{x}, \{r_i\}_{i \in T}) = Pr(S_T | \vec{y}, \{r_i\}_{i \in T}),$$

where the probability space is over all random inputs in $\bar{T}$, 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 $\mathcal{F}$.

3 The Upper Bound

In this section we show that for every t, there is a t–private oblivious protocol which computes the sum and uses only $\left\lceil \frac{t+1}{2} \right\rceil \cdot n$ messages. For simplicity, we will analyze in detail only the n–private version of the protocol. The general t-private protocol is described at the end of this section and its privacy and message complexity are briefly analyzed there as well. The n-private protocol proceeds in rounds, as following:

Round i: $(1 \le i \le n-2)$

> Party P_i computes the sum of all messages it received so far. That is, $w_i = \sum_{j=1}^{i-1} z_{j,i} \bmod m$ (this sum is 0 for $i = 1$). Then, it chooses at random and uniformly $z_{i,i+1}, z_{i,i+2}, \ldots, z_{i,n-1}$ in $\{0, 1, ..., m-1\}$. Finally, P_i computes $z_{i,n}$ such that $x_i + w_i = \sum_{j=i+1}^{n} z_{i,j} \bmod m$, and sends $z_{i,j}$ to P_j $(j > i)$.

Round $n-1$:

> Party P_{n-1} computes $z_{n-1,n} = x_{n-1} + \sum_{j=1}^{n-2} z_{j,n-1} \bmod m$, and sends $z_{n-1,n}$ to party P_n.

Round n:

> Party P_n outputs $sum(\vec{x}) = x_n + \sum_{j=1}^{n-1} z_{j,n} \bmod m$.

Theorem 1*:* *The protocol computes $sum(\vec{x})$ n–privately, and $n \cdot (n-1)/2$ messages are sent in every execution.*

Proof: In round i, party P_i sends $n-i$ messages. Therefore the total number of messages sent during any execution is $(n-1)+\ldots+1 = n\cdot(n-1)/2$. In order to show that the protocol computes the correct result, we notice that

$$\begin{aligned}
x_1 &= z_{1,2}+z_{1,3}+\ldots+z_{1,n-2}+z_{1,n-1}+z_{1,n} \bmod m \\
x_2+z_{1,2} &= z_{2,3}+\ldots+z_{2,n-2}+z_{2,n-1}+z_{2,n} \bmod m \\
&\vdots \\
x_{n-1}+z_{1,n-1}+z_{2,n-1}+\ldots+z_{n-2,n-1} &= z_{n-1,n} \bmod m
\end{aligned}$$

For every $1\le i<j\le n-1$, $z_{i,j}$ appears once on the right hand side, in the i-th equality (the message sent from P_i to P_j), and once on the left hand side, in the j-th equality (the message received by P_j from P_i). Thus, when summing up all the $n-1$ equalities, these $z_{i,j}$ cancel. We are left with $\sum_{i=1}^{n-1} x_i$ on the left hand side, and $\sum_{i=1}^{n-1} z_{i,n}$ on the right hand side. Since the output value, computed by party P_n, is $x_n+\sum_{i=1}^{n-1} z_{i,n} \bmod m$, the protocol computes the correct value.

In order to prove the n-privacy of the protocol, we show that no coalition T gets any additional information. For T which contains all n parties there is nothing to prove. For T of size $n-1$, the input of the non-member can be inferred from the output (the sum of all n inputs) and the sum of the inputs of coalition parties, so again there is nothing to prove. The interesting case is therefore when T contains at most $n-2$ parties.

Let $a\in\{0,1,...,m-1\}$ be a possible output in the protocol. We will show that for every such T, every choice of random inputs $\{r_i\}_{i\in T}$, every choice of inputs $\{y_i\}_{i\in T}$, every input vector $\vec{x}$ that extends $\{y_i\}_{i\in T}$ (i.e., for every $i\in T: x_i=y_i$) and satisfies $sum(\vec{x})=a$, and for every communication S, the conditional probability that S_T is exchanged between parties in T and $\bar{T}$,

$$Pr(S_T|\vec{x},\{r_i\}_{i\in T})$$

is the same.

There are two cases in the proof, depending on whether $P_n\in T$ or $P_n\in\bar{T}$. Assume, first, that $P_n\in T$. For any given setting of messages from $\bar{T}$ to T, that is $z_{i,j}$ with $P_i\in\bar{T}$, $P_j\in T$, the messages from T to $\bar{T}$ are completely determined by the random inputs of coalition members, $\{r_i\}_{i\in T}$. We will shortly show that messages from $\bar{T}$ to T are uniformly and independently distributed in $\{0,1,\ldots,m-1\}$. Therefore, this portion of S_T is distributed in the same way for the relevant conditional probability space. As to messages from $\bar{T}$ to T, let k be the maximum number satisfying $P_k\in\bar{T}$ (by assumption, $k<n$). For $i<k$ with $P_i\notin T$, messages from P_i to parties in T are distributed uniformly and independently since the message $z_{i,k}$ goes from P_i to P_k, and neither of these two parties are in T. For the same reasoning, these sets of messages are independent for the various such P_i's. We still have to deal with messages from P_k to T. For a given setting of messages $z_{i,j}$ with $P_i\in\bar{T}$, $P_j\in T$, $1\le i<k<j$, let us look at the first k equalities

$$\begin{aligned}
x_1 &= z_{1,2}+z_{1,3}+\ldots+z_{1,n-2}+z_{1,n-1}+z_{1,n} \bmod m \\
x_2+z_{1,2} &= z_{2,3}+\ldots+z_{2,n-2}+z_{2,n-1}+z_{2,n} \bmod m \\
&\vdots \\
x_k+z_{1,k}+z_{2,k}+\ldots+z_{k-1,k} &= z_{k,k+1}+\ldots+z_{k,n-1}+z_{k,n} \bmod m
\end{aligned}$$

Every $z_{i,j}$ with $i < j \leq k$ appears both on the left hand side and on the right hand side, and will be canceled when summing up all k equalities. The remaining variables are $\sum_{i=1}^{k} x_i$ on the left hand side, and $\sum_{i=1}^{k} \sum_{j=k+1}^{n} z_{i,j}$ on the right hand side. Rearranging these terms, we get

$$\sum_{j=k+1}^{n} z_{k,j} = \sum_{i=1}^{k} x_i - \sum_{i=1}^{k-1} \sum_{j=k+1}^{n} z_{i,j} \bmod m \qquad (*)$$

The value of the right hand side of (*) is known to the coalition T, because the $z_{i,j}$ there are messages received by parties in T from P_is with $i < k$, and $\sum_{i=1}^{k} x_i$ contains the sum of *all* non-coalition members, and can therefore be deduced from the total sum and the inputs of coalition members. From the arguments before, the value of the right hand side of (*) is uniformly distributed in $\{0, \ldots, m\}$.

The left hand side of (*) is simply the sum of messages sent by P_k to T. By the protocol,

$$\sum_{j=k+1}^{n} z_{k,j} = x_k + \sum_{i=1}^{k-1} z_{i,k} \qquad (**)$$

Therefore, for any setting of the right hand side, any assignment of the variables $z_{k,k+1}, \ldots, z_{k,n}$ (the messages sent by P_k) which sums up to the right hand side will occur with probability exactly $1/m^{n-k-1}$ (these are $n-k$ variables ranging in $\{0, \ldots, m-1\}$, and there is one linear constraint). Intuitively, the coalition T does not get any additional information on the input x_k because the right hand side of (**) contains at least one $z_{i,k}$ with $P_i \in \bar{T}$, a value which is unknown to T. This completes the argument for the case $P_n \in T$.

The remaining case is when $P_n \in \bar{T}$. This case is somewhat easier than the case $P_n \in T$. This is because every $P_i \in \bar{T}$ sends the message $z_{i,n}$ to P_n, and so the messages received by T from P_i are uniformly and independently distributed in $\{0, \ldots, m-1\}$. The detailed formalization of this proof for this case is similar to the one we presented, and is therefore omitted. This completes the proof for the n – privacy of the protocol. $\square$

The t-private protocol for the general case, when $1 \leq t \leq n-2$, is summarized in Figure 1. For simplicity, assume that t is odd, and denote $s = (t+1)/2$. The protocol proceeds in rounds, where in round i, the party P_i sends its messages. P_i sums all messages received in the first $i-1$ rounds, adds this to its input, and splits the result among all designated receivers (see Figure 1). This splitting is done in a similar way to the n-private protocol described above. Notice that every party communicates with exactly $2s = t+1$ parties. This communication, however, is the union of sent and received messages. This way the total number of messages is $n \cdot s = n \cdot (t+1)/2$ ($n \cdot \lceil t + 1/2 \rceil$ for t even), which represents a saving of 50% over the simpler protocol. The basic idea in proving that the protocol is t-private is the following: For every coalition T of size at most t, and every party $P_i \notin T$, the coalition T misses at least one message sent by P_i or received by P_i. This property suffices to show, in a similar fashion to our detailed n-privacy proof, that the coalition gets no additional information on inputs in $\bar{T}$. Notice that the protocol is identical to the n-private protocol for $t = n-2$. For *sum*, $n-2$-privacy implies n-privacy.

Party	Sends to	Receives from
P_1	$P_2,\ldots,P_{s+1},P_{n-s+1},\ldots,P_n$	$--$
P_2	$P_3,\ldots,P_{s+2},P_{n-s+2},\ldots,P_n$	P_1
$\vdots$	$\vdots$	$\vdots$
P_s	$P_{s+1},\ldots,P_{2s},P_n$	$P_1,\ldots,P_{s-1}$
P_{s+1}	$P_{s+2},\ldots,P_{2s+1}$	$P_1,\ldots,P_s$
$\vdots$	$\vdots$	$\vdots$
P_i	$P_{i+1},\ldots,P_{i+s}$	$P_{i-s},\ldots,P_{i-1}$
$\vdots$	$\vdots$	$\vdots$
P_{n-s}	$P_{n-s+1},\ldots,P_{n-1}$	$P_{n-2s},\ldots,P_{n-s-1}$
P_{n-s+1}	$P_{n-s+2},\ldots,P_n$	$P_1,P_{n-2s+1},\ldots,P_{n-s}$
$\vdots$	$\vdots$	$\vdots$
P_{n-1}	P_n	$P_1,\ldots,P_{s-1},P_{n-s-1},\ldots,P_{n-2}$
P_n	$--$	$P_1,\ldots,P_s,P_{n-s},\ldots,P_{n-1}$

Figure 1: The t-private protocol ($s=(t+1)/2$)

4 The Lower Bound

In this section we prove a lower bound on the number of messages required by any oblivious protocol that computes the sum t–privately. The interesting case is where $t \leq n-2$. This is because any $n-1$ parties can compute from $sum(\vec{x})$ and their input values the input value of the last party.

Theorem 2: *Let $t \leq n-2$. Let $\mathcal{F}$ be an oblivious t–private protocol for computing sum. Then, $\mathcal{F}$ uses at least $\lceil n\cdot(t+1)/2\rceil$ messages.*

Proof: Let $\vec{x}$ be any input vector. Let T_i be the set of parties that communicate (either send or receive a message) with P_i during the execution of the protocol $\mathcal{F}$. As the protocol is oblivious, this set is well defined, and depends neither on the inputs nor on the random inputs. We prove that $|T_i| \geq t+1$, for all i. This implies that the number of links on which messages are sent is at least $n\cdot(t+1)/2$. Since this number is an integer, this will prove the $\lceil n\cdot(t+1)/2\rceil$ lower bound.

Assume, by way of contradiction, that $|T_i| \leq t$, for some i. The intuition behind the argument is that in any correct protocol for computing $sum(\vec{x})$, the parties in T_i together will "know" the input value of P_i, and therefore, such a protocol cannot be t–private. Formally, let D_i be the set of all parties except P_i and the parties in T_i. That is, $D_i=\{1,2,\ldots n\}\setminus T_i\setminus\{i\}$. (Note that since $t\leq n-2$ the set D_i is not empty). Let $\vec{y}$,$\vec{z}$ and $\vec{w}$ be input vectors which satisfy:

$y_i = x_i+1 \bmod m,\quad y_j=x_j,$ for all $j\in T_i,\quad \sum_{j\in D_i} y_j = \sum_{j\in D_i} x_j\ -1 \bmod m$

$z_i = x_i,\quad z_j = x_j,$ for all $j\in T_i,\quad z_j=y_j,$ for all $j\in D_i$

$w_i = x_i+1 \bmod m,\quad w_j=x_j,$ for all $j\in T_i,\quad w_j=x_j,$ for all $j\in D_i$

Clearly, $sum(\vec{y})=sum(\vec{x})$, $sum(\vec{z})=sum(\vec{x})-1 \bmod m$, and $sum(\vec{w})=sum(\vec{x})+1 \bmod m$.

Now, the t–privacy of the protocol implies that the coalition T_i should not be able to distinguish between $\vec{x}$ and $\vec{y}$. Formally, for every choice of random inputs $\{r_j\}_{j\in T_i}$ and for every communication S,

$$Pr(S_{T_i}|\vec{x},\{r_j\}_{j\in T_i}) = Pr(S_{T_i}|\vec{y},\{r_j\}_{j\in T_i}) \ . \qquad (***)$$

We will now show that a protocol which satisfies this condition cannot be correct. In the definition of correct protocols we assume that a certain party, say P_n, is the one that produces the final output. We examine two cases, $n \neq i$, and $n = i$. Assume $n \neq i$, then P_n is either in T_i or in D_i. By (***) there exist $\vec{r^x}$ and $\vec{r^y}$, two vectors of random inputs extending $\{r_j\}_{j\in T_i}$, such that $S_{T_i}(\vec{x},\vec{r^x}) = S_{T_i}(\vec{y},\vec{r^y})$. Let $\vec{r^w}$ be the same as $\vec{r^x}$ except with the random input of P_i being the same as in $\vec{r^y}$ (i.e., $r_i^w = r_i^y$). By induction on the round number, $S(\vec{w},\vec{r^w}) = S(\vec{x},\vec{r^x})$. Therefore, in both cases, P_n sees the same input, the same random input, and the same messages, and so P_n must produce the same output. Therefore the output is wrong on at least one of the two inputs $\vec{x}$ and $\vec{w}$ – contradicting the correctness of $\mathcal{F}$. In the case $n = i$, we use similar argument with respect to the input vector $\vec{z}$. $\square$

It follows from the proof of Theorem 2 that the same lower bound holds even if we only require that the protocol computes the correct sum with probability greater than 1/2. Our lower bound technique can be used for proving a $\lceil n \cdot (t+1)/2 \rceil$ lower bound on the number of messages for many "sufficiently sensitive" functions. For most functions, the known upper bounds [3, 4, 1] are substantially larger than this bound. In addition, according to [5, 7], when $t \geq n/2$, many functions are not t–private at all. Other functions are t–private, but require exponentially many messages (exponential in the length of the inputs) in any t –private computation [7]. Finally, we remark that any deterministic protocol computing *sum* is not even 1–private.

References

[1] Beaver D., and J. Bar-Ilan, "Non-Cryptographic Fault-Tolerant Computing in a Constant Expected Number of Rounds of Interaction" *Proc. of 8th PODC*, 1989, pp. 201-209.

[2] Benaloh (Cohen), J.D., "Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret", *Advances in Cryptography - Crypto86 (proceedings)*, A.M. Odlyzko (ed.), Springer-Verlag, Lecture Notes in Computer Science, Vol. 263, pp. 251-260, 1987.

[3] Ben-or M., S. Goldwasser, and A. Wigderson, "Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation" *Proc. of 20th STOC*, 1988, pp. 1-10.

[4] Chaum, D., C. Crépeau, and I. Damgård, "Multiparty Unconditionally Secure Protocols" *Proc. of 20th STOC*, 1988, pp. 11-19.

[5] Chor, B., and E. Kushilevitz, "A Zero-One Law for Boolean Privacy", *SIAM Jour. on Disc. Math.*, Vol. 4, No. 1, February 1991, pp. 36-47. Early version appears in *Proc. of 21th STOC*, 1989, pp. 62-72.

[6] Franklin M., and M. Yung, "Communication Complexity of Secure Computation", *Proc. of 24th STOC*, 1992, pp. 699-710.

[7] Kushilevitz, E., "Privacy and Communication Complexity", *SIAM Jour. on Disc. Math.* Vol. 5, No. 2, May 1992, pp. 273-284. Early version in *Proc. of 30th FOCS*, 1989, pp. 416–421.