


On the (In)Efficiency of Non-Interactive Secure Multiparty Computation

Maki Yoshida¹ and Satoshi Obana²

¹ NICT, Tokyo, Japan

maki-yos@nict.go.jp

² Hosei University, Tokyo, Japan

obana@hosei.ac.jp

Abstract. Secure multi-party computation (MPC) enables multiple players to cooperatively evaluate various functions in the presence of adversaries. In this paper, we consider *non-interactive* MPC (NIMPC) against honest-but-curious adversaries in the information-theoretic setting, which was introduced by Beimel et al. in CRYPTO 2014. Their main focus is to realize stronger security while completely avoiding interaction, and succeeded to show that every function admits a fully robust NIMPC protocol. A drawback of this positive result is the communication complexity, which is linear in the size of the input domain (i.e., exponential in the input length). We first prove that this inefficiency is essentially unavoidable by deriving a lower bound on the communication complexity. However, there is an exponential gap between the derived lower bound and the previous construction. We then reduce the gap between the lower and upper bounds to quadratic in the input length by presenting a much more efficient construction of an important building block, which is an NIMPC protocol for indicator functions.

Keywords: Multiparty computation · Information theoretical setting · Non-interactive · Communication complexity · Lower bound · Upper bound

1 Introduction

Secure multi-party computation (MPC) aims to enable multiple players to cooperatively compute various functions in the presence of adversaries. MPC was first introduced by Yao [10] and because of its importance in cryptography, there have been presented many variants so far [3–5, 7–9]. In CRYPTO 2014 [2], Beimel et al. have introduced a novel type of MPC, called *non-interactive* MPC (NIMPC), against honest-but-curious adversaries in the information theoretical setting, which completely avoids interaction while realizing as strong security as possible. They have succeeded to obtain unconditional positive results for some special cases of interest. In particular, they have presented fully robust protocols for various classes of functions including the class of arbitrary functions. The fully robustness here means that any set of corrupted players cannot obtain

Table 1. The communication complexity of n -player NIMPC protocols for a family of functions $h : \mathcal{X} \rightarrow \{0, 1\}^m$ where $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ and $d' \leq |\mathcal{X}_i| \leq d$ for all $1 \leq i \leq n$.

| | Arbitrary functions | Indicator functions ($m = 1$) |
|---------------------------|---|---------------------------------------|
| Previous protocols in [2] | $ \mathcal{X} \cdot m \cdot d^2 \cdot n$ | $d^2 \cdot n$ |
| Lower bound (Sect. 3) | $ \mathcal{X} \cdot m$ | $\log_2 d' \cdot n$ |
| Our protocols (Sect. 4) | $ \mathcal{X} \cdot m \cdot \lceil \log_2(d+1) \rceil^2 \cdot n$ | $\lceil \log_2(d+1) \rceil^2 \cdot n$ |

any information other than those obtained by an oracle access to the function restricted to the input values of uncorrupted players. However, except for special functions like the summation in an abelian group, the communication complexity is not less than polynomial in the size of the input domain (i.e., exponential in the input length) (Table 1).

The question we ask is whether there is a room to reduce the communication complexity of NIMPC. Unfortunately, relatively less has been known about limitations on the communication complexity of MPC. Recently, the research to tackle the difficult problem of lower bounds for communication in MPC becomes active like Data et al. in CRYPTO 2014 [6]. They have developed novel information-theoretic tools to prove lower bounds on the communication complexity in the traditional (i.e., *interactive*) model involving 3-parties.

In this paper, we study the communication complexity of NIMPC defined in [2]. As a result, we show that the inefficiency on communication of NIMPC is essentially unavoidable except for special classes of functions. The contributions of this paper are as follows.

Communication complexity of NIMPC for the set of any functions:

We derive the first lower bound on the communication complexity of NIMPC for any set of functions. The derived lower bound is the logarithm of the size of the function set. In particular, for the set of arbitrary functions $f : \mathcal{X} \rightarrow \{0, 1\}^m$ where \mathcal{X} is the input domain and m is the output length, the lower bound is $|\mathcal{X}| \cdot m$, i.e., exponential in the input length.

Communication complexity for the set of indicator functions: On the other hand, for the set of indicator functions, where the number of functions is linear in the input and output length, we have a significantly small lower bound. However, the communication complexity of the previous NIMPC protocol for indicator functions in [2] is exponential in the input length. This gap implies an exponential gap between the lower and upper bounds of NIMPC protocols for arbitrary functions because the NIMPC protocol for indicator functions is used as a building block.

Efficient NIMPC protocol for indicator functions: We then reduce the exponential gap between the lower and upper bounds on the communication complexity to quadratic by constructing a much more efficient NIMPC protocol for indicator functions. Specifically, we present a construction of NIMPC protocols for indicator functions whose communication complexity is quadratic in the input length.

Our technique for deriving lower bounds is quite simple and useful for approximating the amount of communication. For the target class of functions, we first assume the existence of a *correct* NIMPC protocol with some communication complexity and show a method for a server to send data to a client by encoding data into a function and evaluating the function with the use of the NIMPC protocol. Thus, the communication complexity is bounded by the size of target class. If the assumed communication complexity is smaller than the logarithm of the size of the target class, the contradiction is implied. Thus, the communication complexity is lower bounded by the logarithm of the size of the target class. A similar technique is used in [1] for proving *impossibility* of multiplicative secret sharing rather than derivation of lower bounds.

2 Preliminaries

We recall the notations and definitions of NIMPC introduced in [2]. For an integer n , let $[n]$ be the set $\{1, 2, \dots, n\}$. For a set $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_n$ and $T \subseteq [n]$, we denote $\mathcal{X}_T \triangleq \prod_{i \in T} \mathcal{X}_i$. For $x \in \mathcal{X}$, we denote by x_T the restriction of x to \mathcal{X}_T , and for a function $h : \mathcal{X} \rightarrow \Omega$, a subset $T \subseteq [n]$, and $x_{\overline{T}} \in \mathcal{X}_{\overline{T}}$, we denote by $h|_{\overline{T}, x_{\overline{T}}} : \mathcal{X} \rightarrow \Omega$ the function h where the inputs in $\mathcal{X}_{\overline{T}}$ are fixed to $x_{\overline{T}}$. For a set S , let $|S|$ denote its size (i.e., cardinality of S).

An NIMPC protocol for a family of functions \mathcal{H} is defined by three algorithms: (1) a randomness generation function GEN, which given a description of a function $h \in \mathcal{H}$ generates n correlated random inputs R_1, \dots, R_n , (2) a local encoding function ENC _{i} ($1 \leq i \leq n$), which takes an input x_i and a random input R_i and outputs a message, and (3) a decoding algorithm DEC that reconstructs $h(x_1, \dots, x_n)$ from the n messages. The formal definition is given as follows:

Definition 1 (NIMPC: Syntax and Correctness). *Let $\mathcal{X}_1, \dots, \mathcal{X}_n, \mathcal{R}_1, \dots, \mathcal{R}_n, \mathcal{M}_1, \dots, \mathcal{M}_n$ and Ω be finite domains. Let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$ and let \mathcal{H} be a family of functions $h : \mathcal{X} \rightarrow \Omega$. A non-interactive secure multi-party computation (NIMPC) protocol for \mathcal{H} is a triplet $\Pi = (\text{GEN}, \text{ENC}, \text{DEC})$ where*

- GEN : $\mathcal{H} \rightarrow \mathcal{R}_1 \times \dots \times \mathcal{R}_n$ is a random function,
- ENC is an n -tuple deterministic functions $(\text{ENC}_1, \dots, \text{ENC}_n)$, where $\text{ENC}_i : \mathcal{X}_i \times \mathcal{R}_i \rightarrow \mathcal{M}_i$,
- DEC : $\mathcal{M}_1 \times \dots \times \mathcal{M}_n \rightarrow \Omega$ is a deterministic function satisfying the following correctness requirement: for any $x = (x_1, \dots, x_n) \in \mathcal{X}$ and $h \in \mathcal{H}$,

$$\Pr[R = (R_1, \dots, R_n) \leftarrow \text{GEN}(h) : \text{DEC}(\text{ENC}(x, R)) = h(x)] = 1, \quad (1)$$

where $\text{ENC}(x, R) \triangleq (\text{ENC}_1(x_1, R_1), \dots, \text{ENC}_n(x_n, R_n))$.

The individual communication complexity of Π is the maximum of $\log |\mathcal{R}_1|, \dots, \log |\mathcal{R}_n|, \log |\mathcal{M}_1|, \dots, \log |\mathcal{M}_n|$. The total communication complexity of Π is the summation of $\log |\mathcal{R}_1|, \dots, \log |\mathcal{R}_n|, \log |\mathcal{M}_1|, \dots, \log |\mathcal{M}_n|$.

We next show the definition of robustness for NIMPC, which states that a coalition can only learn the information they should. In the above setting, a coalition T can repeatedly encode any inputs for T and decode h with the new encoded inputs and the original encoded inputs of \bar{T} . Thus, the following robustness requires that they learn no other information than the information obtained from oracle access to $h|_{\bar{T}, x_{\bar{T}}}$.

Definition 2 (NIMPC: Robustness). *For a subset $T \subseteq [n]$, we say that an NIMPC protocol Π for \mathcal{H} is T -robust if there exists a randomized function Sim_T (a “simulator”) such that, for every $h \in \mathcal{H}$ and $x_{\bar{T}} \in \mathcal{X}_{\bar{T}}$, we have $Sim_T(h|_{\bar{T}, x_{\bar{T}}}) \equiv (M_{\bar{T}}, R_T)$, where R and M are the joint randomness and messages defined by $R \leftarrow \text{GEN}(h)$ and $M_i \leftarrow \text{ENC}_i(x_i, R_i)$.*

For an integer $0 \leq t \leq n$, we say that Π is t -robust if it is T -robust for every $T \subseteq [n]$ of size $|T| \leq t$. We say that Π is fully robust (or simply refer to Π as an NIMPC for \mathcal{H}) if Π is n -robust. Finally, given a concrete function $h : \mathcal{X} \rightarrow \Omega$, we say that Π is a (t -robust) NIMPC protocol for h if it is a (t -robust) NIMPC for $\mathcal{H} = \{h\}$.

As the same simulator Sim_T is used for every $h \in \mathcal{H}$ and the simulator has only access to $h|_{\bar{T}, x_{\bar{T}}}$, NIMPC hides both h and the inputs of \bar{T} . An NIMPC protocol is 0-robust if it is \emptyset -robust. In this case, the only requirement is that the messages (M_1, \dots, M_n) reveal $h(x)$ and nothing else.

An NIMPC protocol is also described in the language of protocols in [2]. Such a protocol involves n players P_1, \dots, P_n , each holding an input $x_i \in \mathcal{X}_i$, and an external “output server,” a player P_0 with no input. The protocol may have an additional input, a function $h \in \mathcal{H}$.

Definition 3 (NIMPC: Protocol Description). *For an NIMPC protocol Π for \mathcal{H} , let $P(\Pi)$ denote the protocol that may have an additional input, a function $h \in \mathcal{H}$, and proceeds as follows.*

Protocol $P(\Pi)(h)$

- **Offline preprocessing:** Each player P_i , $1 \leq i \leq n$, receives the random input $R_i \triangleq \text{GEN}(h)_i \in \mathcal{R}_i$.
- **Online messages:** On input R_i , each player P_i , $1 \leq i \leq n$, sends the message $M_i \triangleq \text{ENC}_i(x_i, R_i) \in \mathcal{M}_i$ to P_0 .
- **Output:** P_0 computes and outputs $\text{DEC}(M_1, \dots, M_n)$.

Informally, the relevant properties of protocol $P(\Pi)$ are given as follows:

- For any $h \in \mathcal{H}$ and $x \in \mathcal{X}$, the output server P_0 outputs, with probability 1, the value $h(x_1, \dots, x_n)$.
- Fix $T \subseteq [n]$. Then, Π is T -robust if in $P(\Pi)$ the set of players $\{P_i\}_{i \in T} \cup \{P_0\}$ can simulate their view of the protocol (i.e., the random inputs $\{R_i\}_{i \in T}$ and the messages $\{M_i\}_{i \in \bar{T}}$) given oracle access to the function h restricted by the other inputs (i.e., $h|_{\bar{T}, x_{\bar{T}}}$).

- Π is 0-robust if and only if in $P(\Pi)$ the output server P_0 learns nothing but $h(x_1, \dots, x_n)$.

We show a claim in [2] stating that for functions outputting more than one bit, we can compute each output bit separately. Based on this fact, in [2], a fully robust NIMPC protocol for the set of indicator functions was first constructed, and then NIMPC protocols for the set of arbitrary functions are constructed based on it.

Proposition 1 (Claim 7 in [2]). *Let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$, where $\mathcal{X}_1, \dots, \mathcal{X}_n$ are some finite domains. Fix an integer $m > 1$. Suppose \mathcal{H} is a family of boolean functions $h : \mathcal{X} \rightarrow \{0, 1\}$ admitting an NIMPC protocol with communication complexity δ . Then, the family of functions $\mathcal{H}^m = \{h : \mathcal{X} \rightarrow \{0, 1\}^m \mid h = h_1 \circ \dots \circ h_m, h_i \in \mathcal{H}\}$ admits an NIMPC protocol with communication complexity $\delta \cdot m$.*

Definition 4 (Indicator Functions). *Let \mathcal{X} be a finite domain. For n -tuple $a = (a_1, \dots, a_n) \in \mathcal{X}$, let $h_a : \mathcal{X} \rightarrow \{0, 1\}$ be the function defined by $h_a(a) = 1$, and $h_a(x) = 0$ for all $a \neq x \in \mathcal{X}$. Let $h_0 : \mathcal{X} \rightarrow \{0, 1\}$ be the function that is identically zero on \mathcal{X} . Let $\mathcal{H}_{\text{ind}} \triangleq \{h_a\}_{a \in \mathcal{X}} \cup \{h_0\}$ be the set of all indicator functions together with h_0 .*

Note that every function $h : \mathcal{X} \rightarrow \{0, 1\}$ can be expressed as the sum of indicator functions, namely, $h = \sum_{a \in \mathcal{X}, h(a)=1} h_a$.

We review the previous results on upper bounds on the *individual* communication complexity of NIMPC. As described above, the NIMPC protocols in [2] are constructed from NIMPC for \mathcal{H}_{ind} . Thus, the previous upper bounds depend on the upper bound for \mathcal{H}_{ind} . This means we have a better upper bound if we obtain a more efficient NIMPC protocol for \mathcal{H}_{ind} .

Proposition 2 (Arbitrary Functions \mathcal{H}_{all} , Proof of Theorem 10 in [2]). *Fix finite domains $\mathcal{X}_1, \dots, \mathcal{X}_n$ such that $|\mathcal{X}_i| \leq d$ for all $1 \leq i \leq n$ and let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$. Let \mathcal{H}_{all} be the set of all functions $h : \mathcal{X} \rightarrow \{0, 1\}^m$. If there exists an NIMPC protocol for \mathcal{H}_{ind} with individual communication complexity δ , then there exists an NIMPC protocol for \mathcal{H} with individual (resp. total) communication complexity $|\mathcal{X}| \cdot m \cdot \delta$ (resp. $|\mathcal{X}| \cdot m \cdot \delta \cdot n$).*

3 Lower Bounds on the Communication Complexity

We derive a lower bound on the *total* communication complexity for any finite set of functions, \mathcal{H}_{all} , and \mathcal{H}_{ind} , respectively.

As described in the introduction, the total communication complexity is bounded by the size of target class. In other words, the total communication complexity cannot be smaller than the logarithm of the size of the target class.

Theorem 1 (Lower bound for any Finite Set of Functions). *Fix finite domains $\mathcal{X}_1, \dots, \mathcal{X}_n$ and Ω . Let $\mathcal{X} \triangleq \mathcal{X}_1, \dots, \mathcal{X}_n$ and \mathcal{H} a set of functions $h : \mathcal{X} \rightarrow \Omega$. Then, any fully robust NIMPC protocol Π for \mathcal{H} satisfies*

$$\sum_{i=1}^n \log |\mathcal{R}_i| \geq \log |\mathcal{H}|, \quad (2)$$

$$\sum_{i=1}^n \log |\mathcal{M}_i| \geq \log |\Omega|. \quad (3)$$

Proof. We first prove Eq. (2). Let $H = |\mathcal{H}|$. Let φ be a one-to-one mapping from \mathcal{H} to $\{0, 1, \dots, H-1\}$. (That is, all functions in \mathcal{H} are numbered on some rule.) Suppose a server holding a random number $a \in \{0, \dots, H-1\}$ aims to send a to a client. Suppose also that there is a NIMPC protocol (GEN, ENC, DEC) for \mathcal{H} that satisfies $\sum_{i=1}^n \log |\mathcal{R}_i| < \log H$. For the function $h = \varphi(a)$, the server executes $R \leftarrow \text{GEN}(h)$ and sends R to the client. The client obtains a by executing ENC and DEC for all possible inputs $x \in \mathcal{X}$ and identifying the function h . We conclude that the server can communicate any $a \in \{0, \dots, H-1\}$ to the client using $R = (R_1, \dots, R_n)$ of which domain size $\prod_{i=1}^n |\mathcal{R}_i|$ is smaller than H , that is impossible. Thus, we have $\sum_{i=1}^n \log |\mathcal{R}_i| \geq \log H$.

In a similar way, we next prove Eq. (3). Suppose a server holding a random element $b \in \Omega$ and aiming to send b to a client and that there is a NIMPC protocol (GEN, ENC, DEC) for \mathcal{H} that satisfies $\sum_{i=1}^n \log |\mathcal{M}_i| < \log |\Omega|$. For a function $h \in \mathcal{H}$ and an element $a \in \mathcal{X}$ such that $h(a) = b$, the server executes $R \leftarrow \text{GEN}(h)$ and $M \leftarrow \text{ENC}(a, R)$, and sends M to the client. The client obtains b by executing DEC. We conclude that the server can communicate any $b \in \Omega$ to the client using $M = (M_1, \dots, M_n)$ of which domain size $\prod_{i=1}^n |\mathcal{M}_i|$ is smaller than $|\Omega|$, that is impossible. Thus, we have $\sum_{i=1}^n \log |\mathcal{M}_i| \geq \log |\Omega|$. \square

The following corollary shows a lower bound on the *total* communication complexity of NIMPC for the set of arbitrary functions. The lower bounds indicate the impossibility of reducing the communication complexity to polynomial in the input length.

Corollary 1 (Lower bound for Arbitrary Functions). *Fix finite domains $\mathcal{X}_1, \dots, \mathcal{X}_n$ such that $|\mathcal{X}_i| \geq d$ for all $1 \leq i \leq n$. Let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$ and \mathcal{H}_{all} the set of all functions $h : \mathcal{X} \rightarrow \{0, 1\}^m$. Any NIMPC protocol Π for \mathcal{H}_{all} satisfies*

$$\sum_{i=1}^n \log |\mathcal{R}_i| \geq m \cdot |\mathcal{X}| \geq d^n \cdot m, \quad (4)$$

$$\sum_{i=1}^n \log |\mathcal{M}_i| \geq m. \quad (5)$$

Proof. The proof is obvious from Theorem 1 by setting $\mathcal{H} = \mathcal{H}_{\text{all}}$. A function maps each input value to some output value. Thus, $|\mathcal{H}|$ is given by multiplying the number of all possible input values by the number of all possible output values, i.e., $2^{m \cdot |\mathcal{X}|}$. Then, $\sum_{i=1}^n \log |\mathcal{R}_i| \geq \log |\mathcal{H}| = m \cdot |\mathcal{X}|$. \square

The following corollary shows a lower bounds on the *total* communication complexity of NIMPC for \mathcal{H}_{ind} . The gap between this lower bound (linear in the

input length) and the previous upper bound (exponential in the input length) is large. In the next section, we will present an efficient NIMPC protocol for \mathcal{H}_{ind} with individual (resp. total) communication complexity $O(n \cdot \log^2 d)$ (resp. $O(n^2 \cdot \log^2 d)$).

Corollary 2 (Lower bound for Indicator Functions). *Fix finite domains $\mathcal{X}_1, \dots, \mathcal{X}_n$ such that $|\mathcal{X}_i| \geq d$ for all $1 \leq i \leq n$ and let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$. Then, any NIMPC protocol Π_{ind} for \mathcal{H}_{ind} satisfies*

$$\sum_{i=1}^n \log |\mathcal{R}_i| \geq \log |\mathcal{X}| \geq n \cdot \log d. \tag{6}$$

Though the proof is obvious from Theorem 1, we give a more constructive proof, which need not to assume an existence of a one-to-one mapping ϕ .

Proof. Suppose a server holding a random vector $a = (a_1, \dots, a_n) \in \mathcal{X}$ and aiming to send a to a client. Suppose that there is an NIMPC protocol (GEN, ENC, DEC) for \mathcal{H}_{ind} that satisfies $\sum_{i=1}^n \log |\mathcal{R}_i| < \log |\mathcal{X}|$. The server executes $R \leftarrow \text{GEN}(h_a)$ and sends R to the client. The client obtains a by executing ENC and DEC for all possible inputs $a' \in \mathcal{X}$ and checking whether the output is 1 or not. The input a' for which the output is 1 is considered as a . We conclude that the server can communicate any $a \in \mathcal{X}$ to the client using $R = (R_1, \dots, R_n)$ of which domain size $\prod_{i=1}^n |\mathcal{R}_i|$ is smaller than $|\mathcal{X}|$, that is impossible. Thus, we have $\sum_{i=1}^n \log |\mathcal{R}_i| \geq \log |\mathcal{X}|$. \square

4 Efficient Constructions

We now present an efficient construction of NIMPC for \mathcal{H}_{ind} . In the previous construction in [2], all the possible input values are encoded in a *unary* way, and thus the communication complexity depends on the size of the input domain. Specifically, each possible input value is represented by a single vector over \mathbb{F}_2 so that the summation of vectors corresponding to $a = (a_1, \dots, a_n)$ is equal to the zero vector while the other combination is linearly independent to satisfy the robustness. Our idea to reduce the communication complexity is to encode all the possible input values in a *binary* way. Specifically, for each bit in the binary representation, two vectors representing “0” and “1” are generated so that the summation of all vectors over the binary representation of a is equal to zero. Since the proposed encoding reduces the required dimension of vectors, the communication complexity of resulting NIMPC is greatly reduced, too.

The detailed description of the protocol is as follows. For $i \in [n]$, let $d_i = |\mathcal{X}_i|$ and ϕ_i a one-to-one mapping from \mathcal{X}_i to $[d_i]$. Let $l_i = \lceil \log_2(d_i + 1) \rceil$ and $s = \sum_{i=1}^n l_i$. Fix a function $h \in \mathcal{H}_{\text{ind}}$ that we want to compute.

The proposed NIMPC $P(\Pi_{\text{ind}})(h)$

- **Offline preprocessing:** If $h = h_0$, then choose s linearly independent random vectors $\{m_{i,j}\}_{i \in [n], j \in [l_i]}$ in \mathbb{F}_2^s . If $h = h_a$ for some $a = (a_1, \dots, a_n) \in \mathcal{X}$,

denote the binary representation of $\phi_i(a_i)$ by $b_i = (b_{i,1}, \dots, b_{i,l_i})$ and define a set of indices I_i by $I_i = \{j \in [l_i] \mid b_{i,j} = 1\}$. Choose s random vectors $\{m_{i,j}\}_{i \in [n], j \in [l_i]}$ in \mathbb{F}_2^s under the constraint that $\sum_{i=1}^n \sum_{j \in I_j} m_{i,j} = 0$ and there are no other linear relations between them (that is, choose all the vectors $m_{i,j}$ except $m_{n, \max I_n}$, as random linear independent vectors and set $m_{n, \max I_n} = -\sum_{i=1}^{n-1} \sum_{j \in I_i} m_{i,j} - \sum_{j \in I_n \setminus \{\max I_n\}} m_{n,j}$). Define $\text{GEN}(h) = R = (R_1, \dots, R_n)$, where $R_i = \{m_{i,j}\}_{j \in [l_i]}$.

- **Online messages:** For an input x_i , let $\hat{b}_i = (\hat{b}_{i,1}, \dots, \hat{b}_{i,l_i})$ be the binary representation of $\phi_i(x_i)$. Let \hat{I}_i be the set of indices defined by $\hat{I}_i = \{j \in [l_i] \mid \hat{b}_{i,j} = 1\}$. $\text{ENC}(x, R) = (M_1, \dots, M_n)$ where $M_i = \sum_{j \in \hat{I}_i} m_{i,j}$.
- **Output** $h(x_1, \dots, x_n)$: $\text{DEC}(M_1, \dots, M_n) = 1$ if $\sum_{i=1}^n M_i = \mathbf{0}$.

Theorem 2. *Fix finite domains $\mathcal{X}_1, \dots, \mathcal{X}_n$ such that $|\mathcal{X}_i| \leq d$ for all $1 \leq i \leq n$ and let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$. Then, there is an NIMPC protocol Π_{ind} for \mathcal{H}_{ind} with individual (resp. total) communication complexity at most $\lceil \log_2(d+1) \rceil^2 \cdot n$ (resp. $\lceil \log_2(d+1) \rceil^2 \cdot n^2$).*

Proof. For the correctness, note that $\sum_{i=1}^n M_i = \sum_{i=1}^n \sum_{j \in \hat{I}_i} m_{i,j}$. If $h = h_a$ for $a \in \mathcal{X}$, this sum equals 0 if and only if $I_i = \hat{I}_i$ for all $i \in [n]$, i.e., $a = x$. If $h = h_0$, this sum is never zero, as all vectors were chosen to be linearly independent in this case.

To prove robustness, fix a subset $T \subset [n]$ and $x_{\bar{T}} \in \mathcal{X}_{\bar{T}}$. The encodings $M_{\bar{T}}$ of \bar{T} consist of the vectors $\{M_i\}_{i \in \bar{T}}$. The randomness R_T consists of the vectors $\{m_{i,j}\}_{i \in [n], j \in [l_i]}$. If $h|_{\bar{T}, x_{\bar{T}}} \equiv 0$, then these vectors are uniformly distributed in \mathbb{F}_2^s under the constraint that they are linearly independent. If $h|_{\bar{T}, x_{\bar{T}}}(x_T) = 1$ for some $x_T \in \mathcal{X}_T$, then $\sum_{i \in \bar{T}} M_i + \sum_{i \in T} \sum_{j \in \hat{I}_i} m_{i,j} = 0$ and there are no other linear relations between them. Formally, to prove the robustness, we describe a simulator Sim_T : the simulator queries $h|_{\bar{T}, x_{\bar{T}}}$ on all possible inputs in \mathcal{X}_T . If all answers are zero, this simulator generates random independent vectors. Otherwise, there is an $x_T \in \mathcal{X}_T$ such that $h|_{\bar{T}, x_{\bar{T}}}(x_T) = 1$, and the simulator outputs random vectors under the constraints described above, that is, all vectors are independent with the exception that $\sum_{i \in T} M_i + \sum_{i \in \bar{T}} \sum_{j \in \hat{I}_i} m_{i,j} = 0$.

The correlated randomness R_i is composed of $l_i \leq \lceil \log_2(d+1) \rceil$ binary vectors of length $s \leq \lceil \log_2(d+1) \rceil \cdot n$ and the encoding is the summation of some of them. Hence, the communication complexity is at most $\lceil \log_2(d+1) \rceil^2 \cdot n$. \square

Corollary 3. *Fix finite domains $\mathcal{X}_1, \dots, \mathcal{X}_n$ such that $|\mathcal{X}_i| \leq d$ for all $1 \leq i \leq n$ and let $\mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$. Then, there is an NIMPC protocol for \mathcal{H}_{all} with individual (resp. total) communication complexity at most $|\mathcal{X}| \cdot m \cdot \lceil \log_2(d+1) \rceil^2 \cdot n$ (resp. $|\mathcal{X}| \cdot m \cdot \lceil \log_2(d+1) \rceil^2 \cdot n^2$).*

From Proposition 2 and Theorem 1, it is obvious.

5 Conclusion

We have presented the first lower bound on the communication complexity of n -player NIMPC protocols for any set of functions including the set of arbitrary

functions and the set of indicator functions. We have constructed novel NIMPC protocols for the set of arbitrary functions and the set of indicator functions. The proposed protocols are much more efficient than the previous protocols. For example, for the set of arbitrary functions, while the previous best known protocol in [2] requires $|\mathcal{X}| \cdot m \cdot d^2 \cdot n$ communication complexity, the communication complexity of the proposed construction is only $|\mathcal{X}| \cdot m \cdot \lceil \log_2(d+1) \rceil^2 \cdot n$, where \mathcal{X} denote the (total) input domain, d is the maximum domain size of a player, and m is the output length. By this result, the gap between the lower and upper bounds on the communication complexity is significantly reduced from $d^2 \cdot n$ to $\lceil \log_2(d+1) \rceil^2 \cdot n$, that is, from the exponential in the input length to the quadratic.

The lower bounds in this paper are derived from the correctness property of NIMPC. While this approach is useful for approximating the communication complexity, there may be a room to improve the lower bounds by taking the robustness property into account. Thus, a possible future work is to derive a tighter lower bound and present an optimum construction of NIMPC.

References

1. Barkol, O., Ishai, Y., Weinreb, E.: On d -multiplicative secret sharing. *J. cryptol.* **23**(4), 580–593 (2010)
2. Beimel, A., Gabizon, A., Ishai, Y., Kushilevitz, E., Meldgaard, S., Paskin-Cherniavsky, A.: Non-interactive secure multiparty computation. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 387–404. Springer, Heidelberg (2014)
3. Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation. In: Proceedings of the 20th Annual ACM Symposium on Theory of Computing, STOC 1988, pp. 1–10 (1988)
4. Chaum, D., Crèpeau, C., Damgård, I.: Multiparty unconditionally secure protocols. In: Proceedings of the 20th Annual ACM Symposium on Theory of Computing, STOC 1988, pp. 11–19 (1988)
5. Cramer, R., Damgård, I.B., Maurer, U.M.: General secure multi-party computation from any linear secret-sharing scheme. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 316–335. Springer, Heidelberg (2000)
6. Data, D., Prabhakaran, M.M., Prabhakaran, V.M.: On the communication complexity of secure computation. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 199–216. Springer, Heidelberg (2014)
7. Hirt, M., Maurer, U.: Player Simulation and General Adversary Structures in Perfect Multiparty Computation. *J. Cryptology* **13**(1), 31–60 (2000)
8. Maurer, U.M.: Secure multi-party computation made simple. In: Cimato, S., Galdi, C., Persiano, G. (eds.) SCN 2002. LNCS, vol. 2576, pp. 14–28. Springer, Heidelberg (2003)
9. Rabin, T., Ben-Or, M.: Verifiable secret sharing and multiparty protocols with honest majority. In: Proceedings of the 21st Annual ACM Symposium on Theory of Computing, STOC 1989, pp. 73–85 (1989)
10. Yao, A.C.: Protocols for secure computations. In: Proceedings of the 23rd Annual Symposium on Foundations of Computer Science, FOCS 1982, pp. 160–164 (1982)