Lower Bounds on the Bottleneck Complexity of Secure Multiparty Computation

Reo Eriguchi[1 [0000−0002−0019−6934]] and Keitaro Hiwatashi[1]

National Institute of Advanced Industrial Science and Technology, Japan
{eriguchi-reo,hiwatashi-keitaro}@aist.go.jp

Abstract. Secure multiparty computation (MPC) is a cryptographic primitive which enables multiple parties to jointly compute a function without revealing any extra information on their private inputs. Bottleneck complexity is an efficiency measure that captures the load-balancing aspect of MPC protocols, defined as the maximum amount of communication required by any party. In this work, we study the problem of establishing lower bounds on the bottleneck complexity of MPC protocols. While the previously known techniques for lower bounding total communication complexity can also be applied to bottleneck complexity, they do not provide nontrivial bounds in the correlated randomness model, which is commonly assumed by existing protocols achieving low bottleneck complexity, or they are applied only to functions of limited practical interest. We propose several novel techniques for lower bounding the bottleneck complexity of MPC protocols. Our methods derive nontrivial lower bounds even in the correlated randomness model and apply to practically relevant functions including the sum function and threshold functions. Furthermore, our lower bounds demonstrate the optimality of some existing MPC protocols in terms of bottleneck complexity or the amount of correlated randomness.

Keywords: Secure computation, Bottleneck complexity

1 Introduction

Secure multiparty computation (MPC) [31] is a fundamental cryptographic primitive which enables mutually distrustful n parties to jointly compute a function $f(x_1, \ldots, x_n)$ without revealing information on their private inputs x_i to adversaries corrupting at most t parties. Traditionally, total communication complexity, which counts the total number of bits transmitted between parties, has been considered as the most fundamental metric to measure the efficiency of MPC protocols. A number of works have made significant progresses to characterize the minimum total communication complexity (e.g., upper bounds were obtained in [20, 3, 5, 7, 8, 6, 23, 25, 22] and lower bounds were obtained in [17, 14, 11, 10, 9]).

However, in practical applications where lightweight devices perform MPC via peer-to-peer communication, the per-party communication cost is a more effective measure than the total cost. For example, consider an MPC protocol

on a star communication pattern, in which a central party interacts with all the other parties and computes an output. Then, while total communication cost is possibly scalable (i.e., $O(n)$), the central party must bear communication proportional to the total number of parties. In large-scale MPC, these costs quickly become prohibitive. To address these concerns, Boyle et al. [4] introduced a more fine-grained efficiency measure, called *bottleneck complexity*, which is defined as the *maximum* amount of communication required by any party during the execution of the protocol. In this paper, we focus on the bottleneck complexity of MPC protocols.

On the positive side, Boyle et al. [4] showed that if fully homomorphic encryption [19, 29] is assumed, then any (possibly insecure) protocol computing a function can be compiled into a secure MPC protocol for the same function preserving bottleneck complexity. As long as computational security is concerned, their results reduce the goal of minimizing the bottleneck complexity of MPC to constructing protocols with low bottleneck complexity without any privacy requirements, which is a purely complexity-theoretic question. Subsequently, a series of works [24, 15, 16] obtained information-theoretically secure MPC protocols with low bottleneck complexity in the *correlated randomness* model [1]. In this model, parties receive correlated randomness from a trusted dealer before inputs are known and then consume the randomness to perform input-dependent computation. Note that without correlated randomness, MPC protocols even for very simple functions such as addition require bottleneck complexity proportional to n due to the lower bounds in [17, 10].

On the negative side, existing lower bounds on the total communication complexity of MPC also lower bound the bottleneck complexity: By definition, if the total communication complexity is lower bounded by C, then the bottleneck complexity is lower bounded by C/n. For example, lower bounds on the total communication complexity in [17, 10] imply $\Omega(n)$ lower bounds on the bottleneck complexity for the sum function and threshold functions. However, these lower bounds do not hold in the correlated randomness model. Indeed, as shown in [27], the sum function can be computed with $O(1)$ bottleneck complexity if parties receive additive shares of zero as correlated randomness. To the best of our knowledge, the only lower bound on bottleneck complexity that still holds in the correlated randomness model was obtained via the results in [9]. The authors of [9] introduced a special function related to private information retrieval and showed that any MPC protocol computing it has roughly at least $\Omega(nt)$ total communication complexity and hence $\Omega(t)$ bottleneck complexity, where t is the number of corrupted parties. However, [9] mainly aimed at showing the existence of hard functions for which secure computation must incur large communication overheads, and as such, the function considered there was not directly related to those of practical interest (e.g., the sum function or threshold functions). Motivated by the above considerations, we ask the following question:

Can we show a lower bound on the bottleneck complexity of MPC protocols for functions of practical interest in the correlated randomness model?

1.1 Our Results

In this paper, we propose three novel methods to provide lower bounds on the bottleneck complexity of information-theoretic (i.e., perfectly correct and perfectly private) MPC protocols. Our methods can be applied even if parties are allowed to receive correlated randomness and provide lower bounds for functions of practical use including the sum function and threshold functions. Furthermore, as a corollary, our lower bounds demonstrate the optimality of some existing MPC protocols. Below, we elaborate on our results.

Lower Bounds. Our first method can be applied to functions that have large ranges even if the inputs of some parties are fixed. Specifically, we show a trade-off between bottleneck complexity and the amount of correlated randomness for such functions. As a remarkable implication, any MPC protocol for computing a special class of low-degree polynomials, including the sum function, must have $\Omega(t)$ bottleneck complexity if the amount of correlated randomness is $o(\log |\mathbb{F}|)$, where $\mathbb{F}$ is the underlying field.

Secondly, we show that the bottleneck complexity is lower bounded if the communication pattern of an MPC protocol satisfies a special property that either the number of corrupted parties interacting with honest parties or the number of honest parties interacting with corrupted parties is limited. For example, any MPC protocol for a threshold function with a threshold $k = \Omega(n)$ must have $\Omega(\log n)$ bottleneck complexity no matter how much correlated randomness is assumed, as long as its communication pattern satisfies the above property. Note that these communication patterns include a cycle (where the i-th party interacts only with the $(i-1)$-th and $(i+1)$-th parties), and a tree (where each party interacts only with parties corresponding to its parent or children nodes). These are common communication patterns assumed by all the existing MPC protocols achieving low bottleneck complexity [4, 27, 24, 15, 16].

Thirdly, we focus on MPC protocols satisfying a stronger security requirement, *function privacy*, where the protocol is associated with a set $\mathcal{F}$ of functions instead of a single function, and corrupted parties learn no information not only on honet parties' inputs but on a function $f \in \mathcal{F}$ to be evaluated. This property is actually satisfied by several MPC protocols achieving low bottleneck complexity [24, 15, 16][1]. We show that any function-private MPC protocols for a set $\mathcal{F}$ of functions must have roughly $\Omega(\log\log|\mathcal{F}|) - O(\log(n-t))$ bottleneck complexity. Note that function-private MPC protocols for $\mathcal{F}$ with $|\mathcal{F}| > 1$ exist only in the correlated randomness model since parties' computation in the online phase is independent of $f \in \mathcal{F}$ and correlated randomness must "encode" f. We provide a quantitative analysis of this: Any function-private MPC protocol for $\mathcal{F}$ must assume correlated randomness of size $\Omega((\log|\mathcal{F}|)/(n-t))$.

We note that our proof techniques rely on the properties of perfect privacy. For instance, we use the fact that for two inputs x, y satisfying $f(x) = f(y)$, any transcript m appearing in the protocol execution on x must also appear

[1] This fact was not explicitly stated in their original works. For completeness, we prove it in Appendix A.

during execution on y. This does not generally hold without perfect privacy. It remains an open question whether our techniques can be extended to non-perfect protocols.

Optimality. Our lower bounds show the optimality of existing MPC protocols for several functions including the sum function and threshold functions.

First, let $f : \mathbb{F}^n \to \mathbb{F}$ be the sum function over a finite field $\mathbb{F}$, that is, $f(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i$. Orlandi et al. [27] showed an MPC protocol for f in the correlated randomness model such that its bottleneck complexity is $O(\log |\mathbb{F}|)$ and the amount of correlated randomness is $O(\log |\mathbb{F}|)$ per party. Clearly, the bottleneck complexity cannot be further reduced since the bit-length of each input $x_i \in \mathbb{F}$ is $\Omega(\log |\mathbb{F}|)$. Our results ensure that if $|\mathbb{F}| = 2^{o(n)}$, the amount of correlated randomness cannot be further reduced either. Indeed, if each party only receives correlated randomness of size $o(\log |\mathbb{F}|)$, then the bottleneck complexity of such a protocol must be $\Omega(n)$.

Secondly, let T_k be a threshold function with a threshold $k = \Theta(n)$. According to a series of results in [24, 15, 16], there exist MPC protocols for T_k in the correlated randomness model achieving $O(\log n)$ bottleneck complexity. All of these protocols assume a communication pattern represented by a cycle. Our lower bound $\Omega(\log n)$ ensures that the bottleneck complexity cannot be further reduced as long as a cycle-like communication pattern is assumed. Note that it remains unknown whether one can circumvent our lower bound and achieve $o(\log n)$ bottleneck complexity by considering communication patterns other than cycles.

Thirdly, let $\mathcal{S}_d$ be the set of all symmetric functions $f : X^n \to \{0, 1\}$ with $|X| = d$.[2] Consider the case of $t = n - 1$ corruptions. According to the results in [24, 15], there exists a function-private MPC protocol for $\mathcal{S}_d$ such that the bottleneck complexity is $O(d \log n)$ and the amount of correlated randomness is $O(n^{d-1})$ per party. On the other hand, our lower bounds for function-private protocols imply that the bottleneck complexity must be $\Omega(d \log n)$ and the amount of correlated randomness must be $\Omega(n^{d-1})$, since $|\mathcal{S}_d| = 2^{\Theta(n^{d-1})}$ if $d = O(1)$. This shows the optimality of the protocol in [24, 15] as a function-private protocol for $\mathcal{S}_d$. We show a detailed comparison between upper and lower bounds in Table 1.

Comparison to Related Works. Franklin and Yung [17] showed that the total communication complexity for the sum function must be at least $\Omega(n^2)$ and hence the bottleneck complexity must be $\Omega(n)$. This implies that sublinear bottleneck complexity cannot be achieved if we do not assume correlated randomness. Our lower bound (Corollary 1) strengthens the negative result in [17] in the sense that the sum function over $\mathbb{F}$ cannot be computed with sublinear bottleneck complexity if the amount of correlated randomness is not sufficient (specifically, if it is less than $o(\log |\mathbb{F}|)$).

[2] A function f is called *symmetric* if $f(x_{\sigma(1)}, \ldots, x_{\sigma(n)}) = f(x_1, \ldots, x_n)$ for any input $(x_1, \ldots, x_n) \in X^n$ and any permutation σ.

Table 1. Comparison between upper and lower bounds on bottleneck complexity and the amount of correlated randomness

Function	BC	CR	Corruption	Reference
Sum over $\mathbb{F}$	$O(\log \lvert\mathbb{F}\rvert)$	$O(\log \lvert\mathbb{F}\rvert)$	$t = n - 1$	[27]
	BC $= \Omega(t)$ or CR $= \Omega(\log \lvert\mathbb{F}\rvert)$		any	Cor. 1
T_k	$O(\log n)$	$O(n)$	$t = n - 1$	[24, 15]
	$O(\log n)$	$O(\epsilon^{-1} \log n)$	$t = n - \epsilon n$	[16]
	$\Omega(\log \min\{k, n-k, t'\})$	–	any	Cor. 2[a]
$\mathcal{S}_d$ with $d = O(1)$	$O(d \log n)$	$O(n^{d-1})$	$t = n - 1$	[24, 15]
	$O(d \log n)$	$O(n^{d-2}\epsilon^{-1} \log n)$	$t = n - \epsilon n$	[16]
	$\Omega(d \log n)$	$\Omega(n^{d-2}\epsilon^{-1})$	$t = n - \epsilon n$	Cors. 3, 4

"BC" stands for bottleneck complexity and "CR" stands for the amount of correlated randomness per party. $\mathbb{F}$ is a finite field, T_k is a threshold function with a threshold k, $\mathcal{S}_d$ is the set of all symmetric functions $f : X^n \to \{0,1\}$ with $|X| = d$, and $t' = \min\{t, \lfloor n/2 \rfloor\}$.

[a] The communication pattern is a cycle or a tree.

The lower bound in [10] implies that the bottleneck complexity for threshold functions must be $\Omega(n)$ if correlated randomness is not assumed. Our results in Corollary 2 show that the lower bound of $\Omega(\log n)$ still holds even if parties have access to an unbounded amount of correlated randomness. It is impossible to obtain a super-logarithmic lower bound $\omega(\log n)$ in the correlated randomness model since threshold functions can be computed with $O(\log n)$ bottleneck complexity assuming correlated randomness of size $O(n)$ [24, 15].

Damgård et al. [9] introduced a special function $\mathrm{PIR}_{n,k} : (\{0,1\}^k \times \{0,1\}^{\log(nk)} \times \{0,1\})^n \to \{0,1\}^n$, which on input $(x_i, z_i, b_i)_{i\in[n]}$, outputs $b_i \cdot x[z]$ as the i-th outputs, where $x \in \{0,1\}^{nk}$ is the concatenation of the x_i's, $z \in \{0,1\}^{\log(nk)}$ is the bit-wise XOR of the z_i's, and $x[z] \in \{0,1\}$ is the z-th bit of x. They showed a lower bound $\Omega(tn)$ for this function, which in turn implies a lower bound $\Omega(t)$ on bottleneck complexity. Prior to our work, this was the first lower bound on bottleneck complexity that holds in the correlated randomness model. However, the function $\mathrm{PIR}_{n,k}$ was introduced to show the existence of functions that are hard to securely compute, and as such, it was a somewhat "artificial" example. We are the first to provide lower bounds on bottleneck complexity for functions that are directly related to real-world applications.

Boyle et al. [4] showed the existence of a function for which any protocol cannot compute with sublinear bottleneck complexity even without any security considerations. However, their proof was based on the counting argument and cannot provide lower bounds on explicit functions. Damgård et al. [12] showed a general method to lower bound the total communication complexity of MPC but did not provide nontrivial lower bounds in the correlated randomness model. Data et al. [14] gave lower bounds on the total communication complexity of MPC in the three-party setting and Damgård et al. [11] gave lower bounds

applicable to a class of MPC protocols following a specific design, which they termed the "gate-by-gate" design.

On the upper-bound side, there are non-perfect MPC protocols achieving low bottleneck complexity in the plain model [13, 26, 18]. Concretely, if a function is computed by an arithmetic circuit with m gates, then the protocol in [13] achieves bottleneck complexity of roughly $O(m/n + \sqrt{n})$ without assuming correlated randomness. It might seem contradictory since lower bounds in [17, 10] and ours imply that the bottleneck complexity must be large unless a sufficient amount of correlated randomness is assumed. This gap comes from the fact that these protocols only achieve non-perfect security while our lower bounds are applied to MPC protocols with perfect correctness and perfect privacy.

2 Preliminaries

2.1 Notations

For a natural number $n \in \mathbb{N}$, we define $[n] = \{1, 2, \ldots, n\}$. Let $X_1, \ldots, X_n$ be sets and $(x_i)_{i\in[n]} \in \prod_{i\in[n]} X_i$. For $A \subseteq [n]$, we denote the Cartesian product $\prod_{i\in A} X_i$ by X_A and $(x_i)_{i\in[n]}$ by x_A. For a set X, let $\bar{X}$ denote its complement. For two random variables X, Y, we denote $X \equiv Y$ if they are identically distributed. For a bit string $x = (x_i)_{i\in[n]} \in \{0,1\}^n$, we denote the Hamming weight of x by $\mathrm{wt}(x)$, i.e., $\mathrm{wt}(x) = |\{i \in [n] : x_i = 1\}|$. Let $\log x$ denote the base-2 logarithm of x.

2.2 Secure Multiparty Computation

We recall the definition of secure multiparty computation (or MPC for short) [21]. We assume that there are n parties $\mathsf{P}_1, \ldots, \mathsf{P}_n$ and that they are connected with a synchronous point-to-point network with authenticated private channels. We also assume the correlated randomness model in which there is a trusted dealer D, who samples and distributes input-independent correlated randomness among the parties. Note that it includes the notion of MPC in the plain model if the dealer D outputs an empty string. Roughly speaking, the execution of a protocol Π among the n parties proceeds as follows. Each party P_i has a private input x_i and the dealer D has a function f. In the offline phase, D computes correlated randomness $(r_1, \ldots, r_n)$ and distributes r_i to P_i for each $i \in [n]$. In the online phase, the parties compute and send messages to each other in the way specified by the protocol. Once all messages are computed, each party P_i obtains a local output y_i.

Formally, a protocol Π among the n parties is specified by a pair of algorithms Gen and $\mathsf{NextMsg}$ of the following syntax:

- Gen is a randomized algorithm that takes a function $f \in \mathcal{F}$ as input and outputs an n-tuple $(r_1, \ldots, r_n)$ of correlated randomness. We say that a tuple $(r_1, \ldots, r_n)$ is *valid* correlated randomness for f if it appears as an output of $\mathsf{Gen}(f)$ with non-zero probability.

– $\mathsf{NextMsg}$ is a deterministic algorithm specifying how a party computes a next message from the messages that the party received so far. Concretely, the input of $\mathsf{NextMsg}$ consists of a party index $i \in [n]$, a number ℓ specifying the current round, the correlated randomness r_i, the input $x_i \in X_i$ of P_i, local randomness sampled by P_i, and all the messages that P_i received so far. The output of $\mathsf{NextMsg}$ is a party index $j \in [n]$ and an outgoing message which P_i should send to a party P_j.

In this paper, we assume that the communication pattern of Π is fixed before the protocol starts. In other words, the party index j outputted by $\mathsf{NextMsg}$ depends only on the party index $i \in [n]$ and the number ℓ of the current round, independent of the input of P_i or his randomness. Furthermore, without loss of generality, we may assume that in each round, one party sends a message to another party and the others send no messages. This is because if we obtain a lower bound on the communication complexity of Π under the above assumption, then it also applies to a protocol Π' where parties send messages in parallel as much as possible. We can thus define the communication pattern of Π as a directed labeled graph $G_\Pi = (V, E, L)$ where V is the set of parties and a pair $(\mathsf{P}_i, \mathsf{P}_j)$ of parties is connected by an edge $e \in E$ with a label $\ell = L(e)$ if and only if P_i sends a message P_j in round ℓ. Note that G_Π possibly has multiple edges.

First, we define the correctness of MPC protocols.

Definition 1 (Correctness). *Let $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_n$ be finite sets. Let $\mathcal{F}$ be a set of functions from $\prod_{i\in[n]} X_i$ to $\prod_{i\in[n]} Y_i$. A protocol Π is correct with respect to $\mathcal{F}$ if for any $f \in \mathcal{F}$ and any $(x_i)_{i\in[n]} \in \prod_{i\in[n]} X_i$, it holds that*

$$\Pr[(y_1, \ldots, y_n) = f(x_1, \ldots, x_n)] = 1,$$

where $(y_1, \ldots, y_n)$ are local outputs of parties obtained as a result of executing Π on f and $(x_i)_{i\in[n]}$.

Next, we define the input privacy of MPC protocols. Define the view of P_i during the execution of Π, as the one including the input x_i of P_i, correlated randomness r_i, local randomness sampled by P_i and messages received by P_i. We say that a protocol is input-private if the views of corrupted parties are independent of the inputs of honest parties.

Definition 2 (Input privacy). *A protocol Π is said to satisfy input privacy under t corruptions (or t-input-privacy for short) with respect to $\mathcal{F}$ if for any $T \subseteq [n]$ of size t, any $f \in \mathcal{F}$ and any pair of inputs $(x_i)_{i\in[n]}, (x'_i)_{i\in[n]} \in \prod_{i\in[n]} X_i$ such that $y_i = y'_i$ and $x_i = x'_i$ for all $i \in T$, where $(y_i)_{i\in[n]} = f((x_i)_{i\in[n]})$ and $(y'_i)_{i\in[n]} = f((x'_i)_{i\in[n]})$, it holds that*

$$(\mathsf{view}_i)_{i\in T} \equiv (\mathsf{view}'_i)_{i\in T},$$

where view_i (resp. view'_i) is the view of P_i during the execution of Π on f and $(x_i)_{i\in[n]}$ (resp. f and $(x'_i)_{i\in[n]}$).

We also introduce the function privacy of Π, which is a stronger property that the views of corrupted parties are not only independent of honest parties' inputs but also of a function to be computed.

Definition 3 (Function privacy). *A protocol Π is said to satisfy input and function privacy under t corruptions (or t-input-function-privacy for short) with respect to $\mathcal{F}$ if for any $T \subseteq [n]$ of size t, any pair of functions $f, f' \in \mathcal{F}$ and any pair of inputs $(x_i)_{i\in[n]}, (x'_i)_{i\in[n]} \in \prod_{i\in[n]} X_i$ such that $y_i = y'_i$ and $x_i = x'_i$ for all $i \in T$, where $(y_i)_{i\in[n]} = f((x_i)_{i\in[n]})$ and $(y'_i)_{i\in[n]} = f'((x'_i)_{i\in[n]})$, it holds that*

$$(\mathsf{view}_i)_{i\in T} \equiv (\mathsf{view}'_i)_{i\in T},$$

where view_i (resp. view'_i) is the view of P_i during the execution of Π on f and $(x_i)_{i\in[n]}$ (resp. f' and $(x'_i)_{i\in[n]}$).

We say that a protocol Π is a t-input-private (resp. t-input-function-private) MPC protocol for a set of functions $\mathcal{F}$ if it is correct and t-input-private (resp. t-input-function-private) with respect to $\mathcal{F}$. If $\mathcal{F}$ consists only of a single function $f : \prod_{i\in[n]} X_i \to \prod_{i\in[n]} Y_i$, i.e., $\mathcal{F} = \{f\}$, we simply say that Π is an MPC protocol for f. Furthermore, if f gives the same outcome of a single-output function f' to all parties, i.e., $f(x_1, \ldots, x_n) = (y, \ldots, y)$, where $y = f'(x_1, \ldots, x_n)$, then by abuse of terminology, we say that Π is an MPC protocol for f'.

We denote by $\mathrm{Comm}_i(\Pi)$ the total number of bits sent or received by the i-th party P_i during the execution of a protocol Π with worst-case inputs. We define the bottleneck complexity of Π as $\mathrm{BC}(\Pi) = \max_{i\in[n]}\{\mathrm{Comm}_i(\Pi)\}$. We denote by $\mathrm{Rand}_i(\Pi)$ the size of correlated randomness for P_i, i.e., the total number of bits that P_i receives from the dealer D, and define $\mathrm{CR}(\Pi) = \max_{i\in[n]}\{\mathrm{Rand}_i(\Pi)\}$.

3 Lower Bounds for Functions with Large Ranges

In this section, we show our first method to prove trade-offs between bottleneck complexity and the amount of correlated randomness. This method can be applied to functions which have large ranges even if the inputs of some parties are fixed. First, we provide the formal description of this property.

Definition 4. *Let f be a function from $\prod_{i\in[n]} X_i$ to Y. For a subset $A \subseteq [n]$ and $x_A \in X_A$, let f_{x_A} denote a function $f_{x_A} : X_{\bar{A}} \to Y$ defined as $f_{x_A}(x_{\bar{A}}) = f(x_A, x_{\bar{A}})$ for all $x_{\bar{A}} \in X_{\bar{A}}$. Define $V_f(t)$ as*

$$V_f(t) = \min_{A\subseteq[n],|A|=t} \max_{x_A\in X_A} \left|\mathrm{Range}(f_{x_A})\right|,$$

where $\mathrm{Range}(f_{x_A}) = \{y \in Y : \text{there exists } x_{\bar{A}} \in X_{\bar{A}} \text{ such that } f_{x_A}(x_{\bar{A}}) = y\}$.

For example, for the sum function $f : \mathbb{G}^n \to \mathbb{G}$ over an abelian group $\mathbb{G}$, it holds that $V_f(t) = |\mathbb{G}|$ for any $t = 0, 1, \ldots, n-2$. Indeed, for any t-sized subset

A and any $x_A \in \mathbb{G}^{|A|}$, the function $f_{x_A} : \mathbb{G}^{|\bar{A}|} \to \mathbb{G}$ is surjective. More generally, $V_f(t)$ can be lower bounded if f is a low-degree polynomial that is nontrivial in the following sense: Let $\mathbb{F}$ be a finite field and $f \in \mathbb{F}[T_1, \ldots, T_n]$ be an n-variate polynomial over $\mathbb{F}$. We say that f is *k-trivial* if there exists a subset $A \subseteq [n]$ of size k such that for any $x_A = (x_i)_{i \in A} \in \mathbb{F}^k$, the $(n-k)$-variate polynomial $f_{x_A}((T_i)_{i \in \bar{A}})$ is a constant polynomial, i.e., f is a function on $(T_i)_{i \in A}$ for any A of size k. We say that f is *k-nontrivial* if it is not k-trivial. Then, the following proposition holds.

Proposition 1. *Let $f \in \mathbb{F}[T_1, \ldots, T_n]$ be a t-nontrivial polynomial of total degree at most d. Then, it holds that $V_f(t) \geq |\mathbb{F}|/d$.*

Proof. Let $A \subseteq [n]$ be a subset of size t. Since f is t-nontrivial, there exist t elements $x_A = (x_i)_{i \in A} \in \mathbb{F}^t$ such that $f_{x_A}((T_i)_{i \in \bar{A}})$ is an $(n-t)$-variate degree-d polynomial that is not a constant. Let $S = \mathrm{Range}(f_{x_A})$ and $s = |S|$. For any $a \in S$, define $f_{x_A}^{-1}(a) = \{x_{\bar{A}} \in \mathbb{F}^{n-t} : f_{x_A}(x_{\bar{A}}) = a\}$ and $m_a = |f_{x_A}^{-1}(a)|$. Since $\sum_{a \in S} m_a = |\mathbb{F}|^{n-t}$, there exists $a \in S$ such that $m_a \geq |\mathbb{F}|^{n-t}/s$. For this a, we have that $q(x_{\bar{A}}) = 0$ for any $x_{\bar{A}} \in f_{x_A}^{-1}(a)$, where $q((T_i)_{i \in \bar{A}}) := f_{x_A}((T_i)_{i \in \bar{A}}) - a$. Since f_{x_A} is not a constant, q is not a zero polynomial. Therefore, we have from Schwartz–Zippel lemma (e.g., [28]) that $m_a \leq d|\mathbb{F}|^{n-t-1}$ and that $|\mathrm{Range}(f_{x_A})| = s \geq |\mathbb{F}|^{n-t}/m_a \geq |\mathbb{F}|/d$. □

Now, we show a trade-off between bottleneck complexity and the amount of correlated randomness for MPC protocols that give the output of a function to only one party and nothing to others.

Theorem 1. *Let $f : \prod_{i \in [n]} X_i \to Y$ be any function. Let Π be a t-input-private MPC protocol for the function g that takes $(x_i)_{i \in [n]} \in \prod_{i \in [n]} X_i$ as input and gives $f(x_1, \ldots, x_n)$ to the n-th party and nothing to others, that is,*

$$g((x_i)_{i \in [n]}) = (\bot, \ldots, \bot, f((x_i)_{i \in [n]})),\ \forall (x_i)_{i \in [n]} \in \prod_{i \in [n]} X_i.$$

Then, it holds that either $\mathrm{CR}(\Pi) \geq \log V_f(t)$ or $\mathrm{BC}(\Pi) \geq t$.

Proof. Let $A = \{\mathsf{P}_n\}$ denote the set of parties receiving the output, which is a singleton in this case. Let B denote the set of parties who communicate with A, and let $C = \overline{A \cup B}$. Let $m_{A \to B}$ and $m_{B \to A}$ denote respectively the messages from A to B and from B to A. Similarly, we define $m_{C \to B}$ and $m_{B \to C}$. Note that there exists no direct message between A and C. Let x_A, x_B and x_C denote the inputs of parties in A, B and C, respectively. Let r_A, r_B and r_C denote the correlated randomness received by parties in A, B and C, respectively. Let R_A, R_B and R_C denote the local randomness generated in the online phase by parties in A, B and C, respectively. We call them *online randomness* to distinguish them from correlated randomness.

Before proving the statement, we note a key feature of messages between parties: The message outgoing from A (i.e., $m_{A \to B}$) is uniquely determined

by incoming message to A (i.e., $m_{B\to A}$), x_A, r_A, and R_A. Similarly, a tuple $(m_{B\to A}, m_{B\to C})$ is uniquely determined by $(m_{A\to B}, m_{C\to B})$, x_B, r_B and R_B, and $m_{C\to B}$ is uniquely determined by $m_{B\to C}$, x_C, r_C, and R_C. Thus, if we fix all the inputs x_S and randomness r_S, R_S for $S \in \{A, B, C\}$, we can define the functions $\Psi^A_{x_A,r_A,R_A}, \Psi^B_{x_B,r_B,R_B}, \Psi^C_{x_C,r_C,R_C}$ such that

- $m_{A\to B} = \Psi^A_{x_A,r_A,R_A}(m_{B\to A})$.
- $(m_{B\to A}, m_{B\to C}) = \Psi^B_{x_B,r_B,R_B}(m_{A\to B}, m_{C\to B})$.
- $m_{C\to B} = \Psi^C_{x_C,r_C,R_C}(m_{B\to C})$.

Then, we define the set of possible messages sent or received by A with (x_A, r_A, R_A), denoted by $M_A^{(x_A,r_A,R_A)}$, as follows:

$$M_A^{(x_A,r_A,R_A)} = \left\{ (m_{A\to B}, m_{B\to A}) : \begin{array}{c} m_{B\to A} \in \mathrm{Domain}(\Psi^A_{x_A,r_A,R_A}), \\ m_{A\to B} = \Psi^A_{x_A,r_A,R_A}(m_{B\to A}) \end{array} \right\},$$

where $\mathrm{Domain}(\Psi^A_{x_A,r_A,R_A})$ is the domain of $\Psi^A_{x_A,r_A,R_A}$, i.e., the set of all possible messages that may appear as $m_{B\to A}$. Similarly, we define $M_B^{(x_B,r_B,R_B)}$ and $M_C^{(x_C,r_C,R_C)}$:

$$M_B^{(x_B,r_B,R_B)} = \left\{ ((m_{A\to B}, m_{B\to A}), (m_{C\to B}, m_{B\to C})) : \begin{array}{c} (m_{A\to B}, m_{C\to B}) \in \mathrm{Domain}(\Psi^B_{x_B,r_B,R_B}), \\ (m_{B\to A}, m_{B\to C}) = \Psi^B_{x_B,r_B,R_B}(m_{A\to B}, m_{C\to B}) \end{array} \right\},$$

$$M_C^{(x_C,r_C,R_C)} = \left\{ (m_{C\to B}, m_{B\to C}) : \begin{array}{c} m_{B\to C} \in \mathrm{Domain}(\Psi^C_{x_C,r_C,R_C}), \\ m_{C\to B} = \Psi^C_{x_C,r_C,R_C}(m_{B\to C}) \end{array} \right\}.$$

We show the following lemma.

Lemma 1. *For any valid correlated randomness (r_A, r_B, r_C), inputs (x_A, x_B, x_C), and online randomness (R_A, R_B, R_C), the intersection $(M_A^{(x_A,r_A,R_A)} \times M_C^{(x_C,r_C,R_C)}) \cap M_B^{(x_B,r_B,R_B)}$ is a singleton and the only element is the actual transcript of Π executed on the correlated randomness (r_A, r_B, r_C), the inputs (x_A, x_B, x_C), and the online randomness (R_A, R_B, R_C).*

Proof (of Lemma 1). Let $\mathsf{trans} = ((m^{\mathrm{t}}_{A\to B}, m^{\mathrm{t}}_{B\to A}), (m^{\mathrm{t}}_{C\to B}, m^{\mathrm{t}}_{B\to C}))$ be the actual transcript of Π executed on the correlated randomness (r_A, r_B, r_C), the inputs (x_A, x_B, x_C), and the randomness (R_A, R_B, R_C). By definition, we have

- $(m^{\mathrm{t}}_{A\to B}, m^{\mathrm{t}}_{B\to A}) \in M_A^{x_A,r_A,R_A}$,
- $((m^{\mathrm{t}}_{A\to B}, m^{\mathrm{t}}_{B\to A}), (m^{\mathrm{t}}_{C\to B}, m^{\mathrm{t}}_{B\to C})) \in M_B^{x_B,r_B,R_B}$,
- $(m^{\mathrm{t}}_{C\to B}, m^{\mathrm{t}}_{B\to C}) \in M_C^{x_C,r_C,R_C}$,

and therefore, we have

$$((m^{\mathrm{t}}_{A\to B}, m^{\mathrm{t}}_{B\to A}), (m^{\mathrm{t}}_{C\to B}, m^{\mathrm{t}}_{B\to C})) \in (M_A^{x_A,r_A,R_A} \times M_C^{x_C,r_C,R_C}) \cap M_B^{x_B,r_B,R_B}.$$

Suppose on the contrary that $\mathsf{trans}' = ((m'_{A\to B}, m'_{B\to A}), (m'_{C\to B}, m'_{B\to C})) \neq \mathsf{trans}$ belongs to $(M_A^{x_A,r_A,R_A} \times M_C^{x_C,r_C,R_C}) \cap M_B^{x_B,r_B,R_B}$. Let i be the first round where trans' and trans differ and P be a party who sends a message in the i-th round. We derive the contradiction in the case that P belongs to A. By the definition of $M_A^{(x_A,r_A,R_A)}$, we have

- $m_{A\to B} = \Psi^A_{x_A,r_A,R_A}(m_{B\to A})$,
- $m'_{A\to B} = \Psi^A_{x_A,r_A,R_A}(m'_{B\to A})$.

Since the $\Psi^A_{x_A,r_A,R_A}$ is a function that maps the incoming messages to A into the outgoing messages from A, the message in $m_{A\to B}$ sent in the i-th round is uniquely determined by the incoming messages in $m_{B\to A}$ before the i-th round. Also, the message in $m'_{A\to B}$ sent in the i-th round is uniquely determined by the incoming messages in $m'_{B\to A}$ before the i-th round. By the definition of i, messages in $m_{B\to A}$ before the i-th round are equal to messages in $m'_{B\to A}$ before the i-th round, and therefore, the message in $m_{A\to B}$ sent in the i-th round is also equal to the message in $m'_{A\to B}$ sent in the i-th round. This contradicts the definition of i. Similarly, we can derive the contradiction in the case that P belongs to B or C. Therefore, the statement holds. □

Then, we prove the statement of Theorem 1. Suppose on the contrary that $\mathrm{BC}(\Pi) < t$ and $\mathrm{CR}(\Pi) < \log V_f(t)$. Since the party P_n in A interchanges at most $|\mathrm{BC}(\Pi)|$ messages, we have that $|B| < t$. From the definition of $V_f(t)$ and the fact that $|A| + |B| \leq t$, there exist x'_A, x'_B such that $\mathrm{Range}(f_{x'_A,x'_B}) \geq V_f(t) =: v$. We thus have the inputs $x_C^1, \ldots, x_C^v$ of C such that $\{f(x'_A, x'_B, x_C^1), \ldots, f(x'_A, x'_B, x_C^v)\}$ are pairwise disjoint. In the rest of the proof, we fix the inputs of A and B to x'_A and x'_B, respectively. Also, we fix the correlated randomness and the online randomness of B to r'_B and R'_B. Since Π is MPC for the function g, the adversary corrupting all the parties in B cannot obtain any information on the input of A, the input of C, and the output received by A. That is, the distribution D of $(m_{A\to B}, m_{B\to A}, m_{C\to B}, m_{B\to C})$ only depends on x'_B, r'_B and R'_B. Let $(m'_{A\to B}, m'_{B\to A}, m'_{C\to B}, m'_{B\to C})$ be an element of the support of D. Then, for any x_C, there exists r_A, r_C, R_A, R_C such that (r_A, r'_B, r_C) is valid correlated randomness and the transcript of Π executed on $((x'_A, r_A, R_A), (x'_B, r'_B, R'_B), (x_C, r_C, R_C))$ is equal to $(m'_{A\to B}, m'_{B\to A}, m'_{C\to B}, m'_{B\to C})$. Thus, for each $i = 1, \ldots, v$, we have correlated randomness and online randomness of A and C, denoted by $r_A^i, r_C^i, R_A^i, R_C^i$, such that (r_A^i, r'_B, r_C^i) is valid correlated randomness and the transcript of Π executed on $((x'_A, r_A^i, R_A^i), (x'_B, r'_B, R'_B), (x_C^i, r_C^i, R_C^i))$ is equal to $(m'_{A\to B}, m'_{B\to A}, m'_{C\to B}, m'_{B\to C})$. Since $\mathrm{CR}(\Pi) < \log v$, there exists $i \neq j \in \{1, \ldots, v\}$ such that $r_A^i = r_A^j$. Without loss of generality, we can assume that $r_A^1 = r_A^2 =: r'_A$. From Lemma 1, for $i = 1, 2, \ldots, N$, we have

$$(m'_{A\to B}, m'_{B\to A}) \in M_A^{(x'_A, r_A^i, R_A^i)} \text{ and } (m'_{C\to B}, m'_{B\to C}) \in M_C^{(x_C^i, r_C^i, R_C^i)}.$$

In particular, this implies that $((m'_{A\to B}, m'_{B\to A}), (m'_{C\to B}, m'_{B\to C})) \in M_A^{(x'_A, r'_A, R_A^1)} \times M_C^{(x_C^2, r_C^2, R_C^2)}$. Therefore, from Lemma 1, the transcript of Π executed on

$((x'_A, r'_A, R^1_A), (x'_B, r'_B, R'_B), (x^2_C, r^2_C, R^2_C))$ is also equal to $((m'_{A\to B}, m'_{B\to A}), (m'_{C\to B}, m'_{B\to C}))$[3]. Since the output that A receives is determined by x'_A, r'_A, R^1_A, $m_{A\to B}$ and $m_{B\to A}$, this implies that $f(x'_A, x'_B, x^1_C) = f(x'_A, x'_B, x^2_C)$. This contradicts the definition of x^1_C and x^2_C. □

Since the value of $V_f(t)$ can be lower bounded for nontrivial low-degree polynomials f, we have the following corollary.

Corollary 1. *Let $\mathbb{F}$ be a finite field. Let p be an $(n-1)$-variate polynomial of total degree at most d that is $(t-1)$-nontrivial. Let $f : \mathbb{F}^n \to \mathbb{F}$ be a polynomial defined as*

$$f(x_1, \ldots, x_n) = p(x_1, \ldots, x_{n-1}) + x_n,\ \forall (x_i)_{i\in[n]} \in \mathbb{F}^n.$$

Then, for any t-input-private protocol Π for f, it holds that either $\mathrm{CR}(\Pi) \geq \log(|\mathbb{F}|/d))$ or $\mathrm{BC}(\Pi) \geq t$.

Proof. Let Π' be a protocol where the n-th party P_n samples a uniformly random element $r \in \mathbb{F}$, runs Π on input $(x_1, \ldots, x_{n-1}, x_n + r)$, and after receiving an output y, P_n outputs $y-r$. Since $f(x_1, \ldots, x_{n-1}, x_n+r) = f(x_1, \ldots, x_{n-1}, x_n)+r$, Π' is a t-input-private protocol Π for the functionality g defined as

$$g((x_i)_{i\in[n]}) = (\bot, \ldots, \bot, f((x_i)_{i\in[n]})),\ \forall (x_i)_{i\in[n]} \in \mathbb{F}^n.$$

Furthermore, $\mathrm{CR}(\Pi') = \mathrm{CR}(\Pi)$ and $\mathrm{BC}(\Pi') = \mathrm{BC}(\Pi)$.

We show that f is t-nontrivial and hence $V_f(t) \geq |\mathbb{F}|/d$ from Proposition 1. Assume otherwise that f is t-trivial. Then, there would exist $A \subseteq [n]$ of size t such that $f_{x_A}(\cdot)$ is a constant polynomial for any $x_A \in \mathbb{F}^t$. We then have that $n \in A$. This is because otherwise, $f_{x_A}((T_i)_{i\in[n]\setminus A}) = p((x_i)_{i\in A}, (T_i)_{i\in[n-1]\setminus A}) + T_n$ is never a constant polynomial due to the term T_n. Let $A' = A \cap [n-1]$, which is of size $t-1$, and $x_{A'} = (x_i)_{i\in A'} \in \mathbb{F}^t$ be any elements. Set $x_n \in \mathbb{F}$ as any element, say $x_n = 0$. Then, $p_{x_{A'}}((T_i)_{i\in[n-1]\setminus A'}) = f((x_{A'}, x_n), (T_i)_{i\in[n-1]\setminus A'}) - x_n = f_{x_A}((T_i)_{i\in[n]\setminus A})$ is a constant polynomial. This implies that p is $(t-1)$-trivial, which is a contradiction.

By applying Theorem 1, we have that if $\mathrm{CR}(\Pi) < \log(|\mathbb{F}|/d))$ then $\mathrm{CR}(\Pi') < \log(|\mathbb{F}|/d))$ and hence $\mathrm{BC}(\Pi) = \mathrm{BC}(\Pi') \geq t$. □

On the upper-bound side, it was shown in [27] that there is an $(n-1)$-input-private MPC protocol Π for the sum function $f(x_1, \ldots, x_n) = \sum_{i\in[n]} x_i$ over $\mathbb{F}$ such that $\mathrm{BC}(\Pi) = O(\log|\mathbb{F}|)$ and $\mathrm{CR}(\Pi) = O(\log|\mathbb{F}|)$. Due to Corollary 1, the amount of correlated randomness cannot be further reduced if $|\mathbb{F}| = 2^{o(n)}$ since otherwise, the bottleneck complexity would be $\Omega(n)$.

[3] Note that (r'_A, r'_B, r^2_C) is valid correlated randomness, and therefore the the protocol execution is valid.

4 Lower Bounds for MPC with Special Communication Patterns

In this section, we show our second method to prove lower bounds on bottleneck complexity. This method is well applied if the communication pattern of a protocol satisfies a property that either the number of corrupted parties interacting with honest parties or the number of honest parties interacting with corrupted parties is limited. To put it formally, we introduce the notion of the *boundary* of a subset of parties in the communication pattern.

Definition 5 (Boundary). *Let Π be an MPC protocol whose communication pattern is $G_\Pi = (V, E, L)$ and $A \subseteq V$ be a subset of parties. We define the boundary* $\mathrm{Bod}_\Pi(A)$ *of A as*

$$\{\mathsf{P} \in A : \textit{there exists a party } \mathsf{P}' \in \bar{A} \textit{ such that } (\mathsf{P}, \mathsf{P}') \in E \textit{ or } (\mathsf{P}', \mathsf{P}) \in E\}.$$

There are two examples where we can choose a subset A of parties such that $|\mathrm{Bod}_\Pi(A)|$ is small. The first one is the case where the communication pattern of Π is a *cycle*, i.e., each party P_i interacts only with P_{i-1} or P_{i+1}, where indices are modulo n. The other case is that the communication pattern is a *tree*, i.e., parties correspond to the nodes of a k-ary tree and each party interacts only with parties corresponding to its parent or children nodes. Then, the following lemma holds.

Lemma 2. *If the communication pattern of a protocol Π is a cycle, then for any $1 \leq t < n$, there exists a subset A of parties such that $|A| = t$ and $|\mathrm{Bod}_\Pi(A)| \leq 2$. If the communication pattern of Π is a tree, then for any $1 \leq t < n$, we can choose a subset A of parties such that $t/2 \leq |A| \leq t$ and $|\mathrm{Bod}_\Pi(A)| \leq 1$.*

Proof. The case of a cycle is straightforward. A subset $A = \{\mathsf{P}_i, \mathsf{P}_{i+1}, \ldots, \mathsf{P}_{i+t-1}\}$ of t consecutive parties in the cycle satisfies the requirement.

Next, assume that the communication pattern of Π is a k-ary tree T. Let P be a party corresponding to the root of T. For $j \in [k]$, let A_j be the sub-tree whose root is the j-th children of P, and $n_j = |A_j|$. If there exists $j \in [k]$ such that $t/2 \leq n_j \leq t$, then the set $A = A_j$ satisfies the requirement as the root of A_j is the only party who may interact with parties not in A_j. Otherwise, either of the following two cases occurs: (1) $n_j < t/2$ for all $j \in [k]$ or (2) $n_j > t$ for some $j \in [k]$.

If the case (1) occurs, since $\sum_{j=1}^{k} n_j = n - 1 \geq t > t/2$, there exists the smallest integer $\ell \in [k]$ such that $\sum_{j=1}^{\ell} n_j > t/2$. For this ℓ, we have that $\sum_{j=1}^{\ell} n_j < t$ since $\sum_{j=1}^{\ell} n_j = \sum_{j=1}^{\ell-1} n_j + n_\ell < t/2 + t/2 = t$. Then, the set $A = \{\mathsf{P}\} \cup \bigcup_{j=1}^{\ell} A_j$ satisfies the requirement as $|A| = 1 + \sum_{j=1}^{\ell} n_j$ is between $t/2$ and t, and P is the only party who may interact with parties not in A.

If the case (2) occurs, then we may assume without loss of generality that $n_1 > t$. Then, we apply the above procedure to a new tree $T' = A_1$. Concretely, let P' be the party corresponding to the root of the tree $T' = A_1$. Let A'_j be

the sub-tree whose root is the j-th children of P', and $n'_j = |A'_j|$. While we have to ensure the existence of the smallest integer ℓ such that $\sum_{j=1}^{\ell} n'_j > t/2$ to continue the procedure, it is indeed the case since $\sum_{j=1}^{k} n'_j = |A_1 \setminus \{\mathsf{P}'\}| = n_1 - 1 \geq t > t/2$. Since the size of sub-trees eventually becomes less than or equal to t, the case (1) will occur at some point during the procedure and then we get a desired set A. □

Furthermore, we introduce an equivalence relation associated with a boolean function f.

Definition 6 (f-equivalence). *Let f be a function from $\prod_{i\in[n]} X_i$ to $\{0,1\}$. For a subset $A \subseteq [n]$, we define an equivalence relation $\sim_f$ on X_A as follows: For any $x_A, x'_A \in X_A$,*

$$x_A \sim_f x'_A \iff \forall x_{\bar{A}} \in X_{\bar{A}},\ f(x_A, x_{\bar{A}}) = f(x'_A, x_{\bar{A}}).$$

We let $S_f(A)$ denote the size of the quotient set $X_A/\sim_f$.

Now, we show a main theorem in this section.

Theorem 2. *Let Π be a t-input-private MPC protocol Π for a function $f : \prod_{i\in[n]} X_i \to \{0,1\}$. Then, it holds that*

$$\mathrm{BC}(\Pi) \geq \max_{A\subseteq[n],|A|\leq t} \frac{\log S_f(A)}{\min\{|\mathrm{Bod}(\Pi, A)|, |\mathrm{Bod}(\Pi, \bar{A})|\}},$$

no matter how much correlated randomness is assumed.

Proof. Without loss of generality, we may assume that the online-phase of Π is deterministic by including randomness generated in the online phase in correlated randomness. Let A be a set of at most t corrupted parties and let $B = \bar{A}$. We denote the inputs of the parties in A and B by $x_A \in X_A$ and $x_B \in X_B$, respectively. Also, we denote correlated randomness distributed to the parties in A and B by r_A and r_B, respectively. Let M be the set of all possible transcripts between A and B. For each r_A, r_B, x_B such that (r_A, r_B) is valid correlated randomness, we define $\phi_{r_A,r_B,x_B} \colon X_A \to M$ as a deterministic function that maps $x_A \in X_A$ into the transcript of Π executed on $((x_A, r_A), (x_B, r_B))$.

We show the following lemma.

Lemma 3. *Let $p_A, p'_A \in X_A$. If there exist r'_A, r'_B, x'_B such that (r'_A, r'_B) is valid correlated randomness and $\phi_{r'_A,r'_B,x'_B}(p_A) = \phi_{r'_A,r'_B,x'_B}(p'_A)$, then $f(p_A, x_B) = f(p'_A, x_B)$ for any $x_B \in X_B$, that is, $p_A \sim_f p'_A$.*

Proof (of Lemma 3). Let $m = \phi_{r'_A,r'_B,x'_B}(p_A) = \phi_{r'_A,r'_B,x'_B}(p'_A)$. First, since the output that B receives in Π is determined by the transcript m, input x'_B and correlated randomness r'_B, it takes the same value both in Π executed on $((p_A, r'_A), (x'_B, r'_B))$ and in Π executed on $((p'_A, r'_A), (x'_B, r'_B))$. We thus have that $f(p_A, x'_B) = f(p'_A, x'_B)$. Without loss of generality, we may assume that

$f(p_A, x'_B) = f(p'_A, x'_B) = 0$. Since Π securely computes f, for any $x_B \in X_B$ such that $f(p_A, x_B) = 0$, there exists r_B such that (r'_A, r_B) is valid correlated randomness and the transcript of Π executed on $((p_A, r'_A), (x_B, r_B))$ is equal to m.

Then, we can prove that the transcript of Π executed on $((p'_A, r'_A), (x_B, r_B))$ is also equal to m by the following argument. Note that the transcripts of Π executed on $((p_A, r'_A), (x_B, r_B))$, $((p_A, r'_A), (x'_B, r'_B))$ and $((p'_A, r'_A), (x'_B, r'_B))$ are equal to m from the definition of m and r_B. Towards a contradiction, assume that the transcript m' of Π executed on $((p'_A, r'_A), (x_B, r_B))$ is not equal to m. Let i be the index of the first round where m' differs from m, i.e., the first $i-1$ messages of m and m' are the same and the i-th message of m' differs from the i-th message of m. Let $m_{<i}$ denote the first $i-1$ messages of m, m_i denote the i-th message of m, and m'_i denote the i-th message of m'. Let P be a party who sends a message in the i-th round in Π. If the party P belongs to A, then m'_i is determined by $m_{<i}$, p'_A and r'_A. We then have that $m'_i = m_i$ since the first $i-1$ messages, the inputs of A and the correlated randomness of A are common in the executions of Π on $((p'_A, r'_A), (x_B, r_B))$ and $((p'_A, r'_A), (x'_B, r'_B))$. If the party P belongs to B, then m'_i is determined by $m_{<i}$, x_B and r_B. Again, we have that $m'_i = m_i$ since the first $i-1$ messages, the inputs of B and the correlated randomness of B are common in the executions of Π on $((p'_A, r'_A), (x_B, r_B))$ and $((p_A, r'_A), (x_B, r_B))$. Therefore, $m_i = m'_i$ in both cases, which contradicts the definition of i.

Now, the output of Π is uniquely determined by the transcript, the inputs of B, and the randomness of B. Therefore, for any $x_B \in X_B$ such that $f(p_A, x_B) = 0$, we have $f(p'_A, x_B) = f(p_A, x_B) = 0$ since $f(p'_A, x_B)$ is determined by (m, x_B, r_B) considering the execution of Π on $((p'_A, r'_A), (x_B, r_B))$ and $f(p_A, x_B)$ is also determined by (m, x_B, r_B) considering the execution of Π on $((p_A, r'_A), (x_B, r_B))$. Note that the above argument in the previous paragraph was based on the assumption that $f(p_A, x_B) = 0$, and therefore we proved $f(p'_A, x_B) = f(p_A, x_B) = 0$ for any $x_B \in X_B$ such that $f(p_A, x_B) = 0$. By the same argument, we have $f(p'_A, x_B) = f(p_A, x_B) = 0$ for any $x_B \in X_B$ such that $f(p'_A, x_B) = 0$. These imply $f(p_A, x_B) = f(p'_A, x_B)$ for any $x_B \in X_B$ since the range of f is $\{0, 1\}$, and therefore we have $p_A \sim_{\mathcal{F}} p'_A$. □

Finally, we show that

$$\mathrm{BC}(\Pi) \geq \frac{\log S_f(A)}{k},$$

where $k = \min\{|\mathrm{Bod}(\Pi, A)|, |\mathrm{Bod}(\Pi, B)|\}$. Suppose on the contrary that $\mathrm{BC}(\Pi) < (\log S_f(A))/k$. Then, the size of M is less than $S_f(A)$ since M consists of messages between $\mathrm{Bod}(\Pi, A)$ and $\mathrm{Bod}(\Pi, B)$. Thus, for any valid correlated randomness (r_A, r_B) and the input x_B of B, there exist $p_A, p'_A \in X_A$ such that $p_A \nsim_f p'_A$ and $\phi_{r_A, r_B, x_B}(p_A) = \phi_{r_A, r_B, x_B}(p'_A)$. However, as shown above, $\phi_{r_A, r_B, x_B}(p_A) = \phi_{r_A, r_B, x_B}(p'_A)$ implies that $p_A \sim_f p'_A$, which is a contradiction. □

We apply Theorem 2 to obtain lower bounds for threshold functions. Let $T_k : \{0,1\}^n \to \{0,1\}$ be a threshold function defined as

$$T_k(x_1, \ldots, x_n) = \mathbf{1}(\mathrm{wt}(x_1, \ldots, x_n) \geq k) = \begin{cases} 1, & \text{if } \mathrm{wt}(x_1, \ldots, x_n) \geq k, \\ 0, & \text{otherwise.} \end{cases}$$

The following lemma shows a lower bound on $S_{T_k}(A)$.

Lemma 4. *Let $A \subseteq [n]$ be a set such that $|A| \leq n/2$. Then, it holds that*

$$S_{T_k}(A) \geq \min\{k, n-k, |A|\}.$$

Proof. Let $a = |A| \leq n/2$ and $f_{n,k}(a) = \min\{k, n-k, a\}$. Let $x_A, x'_A \in X_A = \{0,1\}^a$. Clearly, if $\mathrm{wt}(x_A) = \mathrm{wt}(x'_A)$, then $x_A \sim_{T_k} x'_A$. Suppose that $\mathrm{wt}(x_A) < \mathrm{wt}(x'_A)$. Since $T_k(x_A, x_{\bar{A}}) = \mathbf{1}(\mathrm{wt}(x_A) + w \geq k)$ where $w = \mathrm{wt}(x_{\bar{A}})$, it holds that $x_A \sim_{T_k} x'_A$ if and only if $\mathrm{wt}(x'_A) + w < k$ for all $w \in \{0, 1, \ldots, n-a\}$ or $\mathrm{wt}(x_A) + w \geq k$ for all $w \in \{0, 1, \ldots, n-a\}$. This condition is equivalent to the condition that $\mathrm{wt}(x'_A) < k - n + a$ or $\mathrm{wt}(x_A) \geq k$.

First, assume that $k \geq n/2$. If $a \leq n-k$, then $f_{n,k}(a) = a$. Furthermore, since $k - n + a \leq 0$ and $a \leq n/2 \leq k$, all the equivalence classes under $\sim_{T_k}$ are

$$\{x_A : \mathrm{wt}(x_A) = i\} \ (0 \leq i \leq a)$$

and hence $S_{T_k}(A) = a + 1 \geq f_{n,k}(a)$. On the other hand, if $a > n-k$, then $f_{n,k}(a) = n-k$. Furthermore, since $k - n + a \geq 1$ and $a \leq k$, all the equivalence classes are

$$\{x_A : 0 \leq \mathrm{wt}(x_A) < k - n + a\} \text{ and } \{x_A : \mathrm{wt}(x_A) = i\} \ (k - n + a \leq i \leq a),$$

and hence $S_{T_k}(A) = n - k + 2 \geq f_{n,k}(a)$.

Second, assume that $k < n/2$. If $a \leq k$, then $f_{n,k}(a) = a$. Furthermore, since $k - n + a < 0$ and $a \leq k$, all the equivalence classes are

$$\{x_A : \mathrm{wt}(x_A) = i\} \ (0 \leq i \leq a)$$

and hence $S_{T_k}(A) = a + 1 \geq f_{n,k}(a)$. On the other hand, if $a > k$, then $f_{n,k}(a) = k$. Furthermore, since $k - n + a < 0$, all the equivalence classes are

$$\{x_A : \mathrm{wt}(x_A) = i\} \ (0 \leq i \leq k) \text{ and } \{x_A : k < \mathrm{wt}(x_A) \leq a\},$$

and hence $S_{T_k}(A) = k + 2 \geq f_{n,k}(a)$. □

Corollary 2. *Let Π be a t-input-private MPC protocol Π for a threshold function T_k. Let $t' = \min\{t, \lfloor n/2 \rfloor\}$. If the communication pattern of Π is a cycle, it holds that $\mathrm{BC}(\Pi) \geq \log \min\{k, n-k, t'\}/2$. If the communication pattern of Π is a tree, it holds that $\mathrm{BC}(\Pi) \geq \log \min\{k, n-k, t'/2\}$.*

Proof. First, assume that the communication pattern of Π is a cycle. From Lemma 2, there exists a set A of parties such that $|A| = t'$ and $|\mathrm{Bod}_\Pi(A)| \leq 2$. Then, since $|A| \leq n/2$, it follows from Lemma 4 that $S_{T_k}(A) \geq \min\{k, n-k, t'\}$. Therefore, since $|A| \leq t$, we obtain the first statement by applying Theorem 2.

Second, assume that the communication pattern is a tree. From Lemma 2, there exists a set A of parties such that $t'/2 \leq |A| \leq t'$ and $|\mathrm{Bod}_\Pi(A)| \leq 1$. Then, since $|A| \leq n/2$, it follows from Lemma 4 that $S_{T_k}(A) \geq \min\{k, n-k, |A|\} \geq \min\{k, n-k, t'/2\}$. Therefore, since $|A| \leq t$, we obtain the second statement by applying Theorem 2. □

On the upper-bound side, [24, 15] showed a t-input-private MPC protocol Π for T_k such that $\mathrm{BC}(\Pi) = O(\log n)$ and $\mathrm{CR}(\Pi) = O(n)$, assuming a cycle communication pattern. The lower bound in Corollary 2 implies that the bottleneck complexity cannot be further improved as long as a cycle or tree communication pattern is assumed. In other words, achieving $o(\log n)$ bottleneck complexity must adopt a different communication pattern than a cycle or a tree.

5 Lower Bounds for MPC with Function Privacy

In this section, we show a method to prove lower bounds on the bottleneck complexity and on the amount of correlated randomness for function-private MPC protocols. First, we provide a lower bound on bottleneck complexity.

Theorem 3. *Let $\mathcal{F}$ be a set of functions from $\prod_{i\in[n]} X_i$ to Y. For each $f \in \mathcal{F}$, let $g_f : \prod_{i\in[n]} X_i \to Y^n$ denote a function that takes $(x_i)_{i\in[n]} \in \prod_{i\in[n]} X_i$ as input and distributes $y = f((x_i)_{i\in[n]})$ to all parties, i.e., $g_f(x_1, \ldots, x_n) = (y, \ldots, y)$, and let $\mathcal{F}' = \{g_f : f \in \mathcal{F}\}$. Let Π be a t-input-function-private MPC protocol for $\mathcal{F}'$. Then, it holds that*

$$\mathrm{BC}(\Pi) \geq \frac{1}{2}\left(\log \frac{\log|\mathcal{F}|}{n-t} - \ell_{\mathrm{in}}\right),$$

no matter how much correlated randomness is assumed, where $\ell_{\mathrm{in}} = \max_{i\in[n]} \log|X_i|$.

Proof. Let A be a set of t corrupted parties and let $B = \bar{A}$. We denote the inputs of the parties in A and B by $x_A \in X_A$ and $x_B \in X_B$, respectively. Also, we denote correlated randomness distributed to the parties in A and B by r_A and r_B, respectively. By fixing randomness generated in the online phase, the correlated randomness r_i of the i-th party P_i determines a function $M_i^{r_i}$ that maps a tuple of his input and incoming messages into outgoing messages.

First, we show that

$$\max_{i\in[n]} \log|\mathcal{R}_i| \geq \frac{\log|\mathcal{F}|}{n-t}, \tag{1}$$

where $\mathcal{R}_i = \{M_i^{r_i} : r_i \text{ is possible correlated randomness distributed to } \mathsf{P}_i\}$. Suppose on the contrary that $\max_i \log|\mathcal{R}_i| < (\log|\mathcal{F}|)/(n-t)$. For any correlated

randomness r_B of B, let $M_B^{r_B} = (M_i^{r_i})_{i\in B}$ denote the function that maps a tuple of x_B and incoming messages of B into outgoing messages of B. Since $\max_i \log|\mathcal{R}_i| < (\log|\mathcal{F}|)/(n-t)$, the function $M_i^{r_i}$ can be represented with less than $(\log|\mathcal{F}|)/(n-t)$ bits, and therefore $M_B^{r_B}$ can be represented with $(\log|\mathcal{F}|)/(n-t)\times|B| < \log|\mathcal{F}|$ bits. Fix r_A arbitrarily. From the security requirement, for any function $f\in\mathcal{F}$, there exists r_B such that (r_A, r_B) is valid correlated randomness for f. We let $r_{B|f}$ denote such r_B.[4] Hence, there exist distinct $f, f'\in\mathcal{F}$ such that $M_B^{r_{B|f}} = M_B^{r_{B|f'}}$. Since the output of Π is determined by x_A, r_A, x_B and $M_B^{r_B}$, this implies that $f(x_A, x_B) = f'(x_A, x_B)$ for all x_A and x_B, and contradicts that f and f' are distinct.

Then, we show the lower bound for $\mathrm{BC}(\Pi)$ given in the statement. Since $M_i^{r_i}$ is the function whose input is of at most $\mathrm{BC}(\Pi)+\log|X_i|$ bits and whose output is of at most $\mathrm{BC}(\Pi)$ bits, the cardinality of $\mathcal{R}_i$ is upper-bounded by the number of functions from $\{0,1\}^{\mathrm{BC}(\Pi)+\log|X_i|}$ to $\{0,1\}^{\mathrm{BC}(\Pi)}$. Therefore, we have

$$\log\left|\mathcal{R}_i\right| \le \mathrm{BC}(\Pi)2^{\mathrm{BC}(\Pi)+\ell_{\mathrm{in}}},$$

and combining it with (1), we have

$$\log\frac{\log|\mathcal{F}|}{n-t} \le \mathrm{BC}(\Pi)+\ell_{\mathrm{in}}+\log\mathrm{BC}(\Pi) \le 2\mathrm{BC}(\Pi)+\ell_{\mathrm{in}}.$$

This completes the proof. □

We apply Theorem 3 to a special class of symmetric functions and obtain lower bounds matching the upper bounds in [24, 15, 16]. We call a function $f: X^n\to Y$ *symmetric* if $f(x_{\sigma(1)},\dots,x_{\sigma(n)}) = f(x_1,\dots,x_n)$ for any $(x_1,\dots,x_n)\in X^n$ and any permutation σ over $[n]$. Let $\mathcal{S}_d$ denote the set of all symmetric functions from $[d]^n$ to $\{0,1\}$. Let $\mathcal{S}'_d = \{g_f : f\in\mathcal{S}_d\}$, where g_f is a function that takes $(x_i)_{i\in[n]}\in X^n$ as input and distributes $y = f(x_1,\dots,x_n)$ to all parties. Since $|\mathcal{S}_d| = 2^{\binom{n+d-1}{d-1}}$, we have the following corollary.

Corollary 3. *For any t-input-function-private protocol Π for $\mathcal{S}'_d$, it holds that*

$$\mathrm{BC}(\Pi) \ge \frac{1}{2}\left(\log\binom{n+d-1}{d-1} - \log(n-t) - \log d\right).$$

On the upper-bound side, a series of works [24, 15, 16] imply that there exists an $(n-1)$-input-function-private MPC protocol Π for $\mathcal{S}'_d$ such that $\mathrm{BC}(\Pi) = O(d\log n)$. If $d = O(1)$, Corollary 3 implies $\mathrm{BC}(\Pi) = \Omega(d\log n)$ matching the upper bound in [24, 15, 16].

Next, we provide a lower bound on the amount of correlated randomness.

Theorem 4. *Continuing the notations in Theorem 3, it holds that*

$$\mathrm{CR}(\Pi) \ge \frac{\log|\mathcal{F}|}{n-t}.$$

[4] If several r_B satisfy that (r_A, r_B) is valid correlated randomness for f, we choose one of them arbitrarily as $r_{B|f}$.

Proof. Let A be a set of t corrupted parties and let $B = \bar{A}$. We denote the inputs of the parties in A and B by $x_A \in X_A$ and $x_B \in X_B$, respectively. Also, we denote correlated randomness distributed to the parties in A and B by r_A and r_B, respectively.

Suppose on the contrary that $\mathrm{CR}(\Pi) \leq (\log|\mathcal{F}|)/(n-t)$. Then, the bit-length of r_B is less than $\log|\mathcal{F}|)/(n-t) \times |B| < \log|\mathcal{F}|$. Furthermore, from the input-function-privacy, for any correlated randomness r_A and $f \in \mathcal{F}$, there exists r_B such that (r_A, r_B) is valid correlated randomness for f. This implies that for any r_A, we can define a function ϕ_{r_A} that maps each function $f \in \mathcal{F}$ into part of correlated randomness $\phi_{r_A}(f)$ such that $(r_A, \phi_{r_A}(f))$ is valid correlated randomness for f. Since the size of the set of all possible r_B's is less than $2^{\log|\mathcal{F}|} = |\mathcal{F}|$, there exist r_A and $f \neq f' \in \mathcal{F}$ such that $\phi_{r_A}(f) = \phi_{r_A}(f')(=: r_B)$. The output of Π with input (x_A, x_B) and correlated randomness (r_A, r_B) is equal to $f(x_A, x_B)$ from the correctness since $(r_A, r_B) = (r_A, \phi_{r_A}(f))$ is valid correlated randomness for f. It is also equal to $f'(x_A, x_B)$ since $(r_A, r_B) = (r_A, \phi_{r_A}(f'))$ is valid correlated randomness for f. This implies f is equal to f', contradicting $f \neq f'$. Hence, we have $\mathrm{CR}(\Pi) \geq (\log|\mathcal{F}|)/(n-t)$. □

Again, applying Theorem 4 to the class of symmetric functions, we obtain the following corollary.

Corollary 4. *For any t-input-function-private protocol Π for $\mathcal{S}'_d$, it holds that*

$$\mathrm{CR}(\Pi) \geq \frac{1}{n-t}\binom{n+d-1}{d-1}.$$

On the upper-bound side, the $(n-1)$-input-function-private MPC protocol Π for $\mathcal{S}'_d$ with $\mathrm{BC}(\Pi) = O(d \log n)$ [24, 15] uses per-party correlated randomness of size $O(n^{d-1})$. Corollary 4 implies that the amount of correlated randomness of [24, 15] cannot be further reduced if $d = O(1)$. In the case of $t = n - \epsilon n$ for $0 < \epsilon < 1$, [16] showed a t-input-function-private MPC protocol Π for $\mathcal{S}'_d$ such that $\mathrm{CR}(\Pi) = O(n^{d-2}\epsilon^{-1} \log n)$.[5] Corollary 4 gives an almost matching lower bound $\mathrm{CR}(\Pi) = \Omega(n^{d-2}\epsilon^{-1})$ if $d = O(1)$.

References

1. Beaver, D.: Efficient multiparty protocols using circuit randomization. In: Advances in Cryptology – CRYPTO '91. pp. 420–432 (1992)
2. Beimel, A., Gabizon, A., Ishai, Y., Kushilevitz, E., Meldgaard, S., Paskin-Cherniavsky, A.: Non-interactive secure multiparty computation. In: Advances in Cryptology – CRYPTO 2014, Part II. pp. 387–404 (2014)
3. Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation. In: Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing. pp. 1–10 (1988)

[5] The original protocol in [16] assumed binary inputs (i.e., $d = 2$) but the generalization to $d > 2$ is straightforward. For completeness, we prove a generalized protocol in Appendix A.

4. Boyle, E., Jain, A., Prabhakaran, M., Yu, C.H.: The Bottleneck Complexity of Secure Multiparty Computation. In: 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018). Leibniz International Proceedings in Informatics (LIPIcs), vol. 107, pp. 24:1–24:16 (2018)
5. Chaum, D., Crépeau, C., Damgard, I.: Multiparty unconditionally secure protocols. In: Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing. pp. 11–19. STOC '88 (1988)
6. Chida, K., Genkin, D., Hamada, K., Ikarashi, D., Kikuchi, R., Lindell, Y., Nof, A.: Fast large-scale honest-majority MPC for malicious adversaries. In: Advances in Cryptology – CRYPTO 2018, Part III. pp. 34–64 (2018)
7. Damgård, I., Nielsen, J.B.: Scalable and unconditionally secure multiparty computation. In: Advances in Cryptology – CRYPTO 2007. pp. 572–590 (2007)
8. Damgård, I., Pastro, V., Smart, N., Zakarias, S.: Multiparty computation from somewhat homomorphic encryption. In: Advances in Cryptology – CRYPTO 2012. pp. 643–662 (2012)
9. Damgård, I., Larsen, K.G., Nielsen, J.B.: Communication lower bounds for statistically secure mpc, with or without preprocessing. In: Advances in Cryptology – CRYPTO 2019. pp. 61–84 (2019)
10. Damgård, I., Nielsen, J.B., Ostrovsky, R., Rosén, A.: Unconditionally secure computation with reduced interaction. In: Advances in Cryptology – EUROCRYPT 2016. pp. 420–447 (2016)
11. Damgård, I., Nielsen, J.B., Polychroniadou, A., Raskin, M.: On the communication required for unconditionally secure multiplication. In: Advances in Cryptology — CRYPTO 2016. pp. 459–488 (2016)
12. Damgård, I.B., Li, B., Schwartzbach, N.I.: More Communication Lower Bounds for Information-Theoretic MPC. In: 2nd Conference on Information-Theoretic Cryptography (ITC 2021). Leibniz International Proceedings in Informatics (LIPIcs), vol. 199, pp. 2:1–2:18 (2021)
13. Dani, V., King, V., Movahedi, M., Saia, J., Zamani, M.: Secure multi-party computation in large networks. Distributed Computing **30**, 193–229 (2017)
14. Data, D., Prabhakaran, M.M., Prabhakaran, V.M.: On the communication complexity of secure computation. In: Advances in Cryptology – CRYPTO 2014. pp. 199–216 (2014)
15. Eriguchi, R.: Unconditionally secure multiparty computation for symmetric functions with low bottleneck complexity. In: Advances in Cryptology – ASIACRYPT 2023. pp. 335–368 (2023)
16. Eriguchi, R.: Secure multiparty computation of symmetric functions with polylogarithmic bottleneck complexity and correlated randomness. In: 5th Conference on Information-Theoretic Cryptography (ITC 2024). Leibniz International Proceedings in Informatics (LIPIcs), vol. 304, pp. 10:1–10:22 (2024)
17. Franklin, M., Yung, M.: Communication complexity of secure computation (extended abstract). In: Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing. pp. 699–710. STOC '92 (1992)
18. Gelles, Y., Komargodski, I.: Optimal load-balanced scalable distributed agreement. Cryptology ePrint Archive, Paper 2023/1139 (2023), `https://eprint.iacr.org/2023/1139`
19. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing. pp. 169–0178. STOC '09 (2009)

20. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game. In: Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing. pp. 218–229. STOC '87 (1987)
21. Goldreich, O.: Foundations of cryptography: volume 2, basic applications. Cambridge University Press (2009)
22. Goyal, V., Li, H., Ostrovsky, R., Polychroniadou, A., Song, Y.: ATLAS: Efficient and scalable MPC in the honest majority setting. In: Advances in Cryptology – CRYPTO 2021, Part II. pp. 244–274 (2021)
23. Goyal, V., Song, Y., Zhu, C.: Guaranteed output delivery comes free in honest majority MPC. In: Advances in Cryptology – CRYPTO 2020, Part II. pp. 618–646 (2020)
24. Keller, H., Orlandi, C., Paskin-Cherniavsky, A., Ravi, D.: MPC with low bottleneck-complexity: Information-theoretic security and more. In: 4th Information-Theoretic Cryptography (ITC) Conference (2023), `https://eprint.iacr.org/2023/683`
25. Keller, M.: MP-SPDZ: A versatile framework for multi-party computation. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. pp. 1575–1590. CCS '20 (2020)
26. King, V., Saia, J., Sanwalani, V., Vee, E.: Scalable leader election. In: SODA. vol. 6, pp. 990–999 (2006)
27. Orlandi, C., Ravi, D., Scholl, P.: On the bottleneck complexity of mpc with correlated randomness. In: Public-Key Cryptography – PKC 2022, Part I. pp. 194–220 (2022)
28. Schwartz, J.T.: Fast probabilistic algorithms for verification of polynomial identities. Journal of the ACM **27**(4), 701–717 (1980)
29. van Dijk, M., Gentry, C., Halevi, S., Vaikuntanathan, V.: Fully homomorphic encryption over the integers. In: Advances in Cryptology – EUROCRYPT 2010. pp. 24–43 (2010)
30. Yamamoto, H.: Secret sharing system using (k, L, n) threshold scheme. Electronics and Communications in Japan (Part I: Communications) **69**(9), 46–54 (1986)
31. Yao, A.C.: Protocols for secure computations. In: Proceedings of the 23rd Annual Symposium on Foundations of Computer Science. pp. 160–164. SFCS '82 (1982)

A A Note on the Protocols in [24, 15, 16]

The authors of [24, 15] showed an $(n-1)$-input-private MPC protocol Π for the class of symmetric functions $\mathcal{S}'_d$ such that $\mathrm{BC}(\Pi) = O(d \log n)$ and $\mathrm{CR}(\Pi) = O(n^{d-1})$. Furthermore, for $t = n - \epsilon n$ for $0 < \epsilon < 1$, [16] showed a t-input-private protocol Π for $\mathcal{S}'_2$ such that $\mathrm{BC}(\Pi) = O(\log n)$ and $\mathrm{CR}(\Pi) = O(\epsilon^{-1} \log n)$. In this section, we prove that the protocol in [16] can be generalized into the case of $\mathcal{S}'_d$ for $d > 2$ and all the above-mentioned protocols satisfy function privacy. Much of this section is taken almost verbatim from [16]. We assume basic terminologies and facts on MPC including the composition theorem (see, e.g., [21] for the details).

A.1 Basic Protocols

Let $\mathbb{G}$ be an abelian group (e.g., a finite field or a ring of integers modulo m). Define $\mathsf{Additive}_{\mathbb{G}}(s)$ as an algorithm to generate additive shares over $\mathbb{G}$ for a

secret $s \in \mathbb{G}$. Formally, on input $s \in \mathbb{G}$, $\mathsf{Additive}_{\mathbb{G}}(s)$ chooses $(s_1, \ldots, s_n) \in \mathbb{G}^n$ uniformly at random conditioned on $s = \sum_{i \in [n]} s_i$, and outputs it.

The functionality $\mathcal{F}_{\mathsf{Sum},\mathbb{G}}$ receives group elements $x_1, \ldots, x_n \in \mathbb{G}$, each from P_i, and gives $s := \sum_{i \in [n]} x_i$ to all parties. As mentioned in [27], it is straightforward to obtain an $(n-1)$-input-private protocol $\Pi_{\mathsf{Sum},\mathbb{G}}$ realizing $\mathcal{F}_{\mathsf{Sum},\mathbb{G}}$ such that $\mathrm{CR}(\Pi_{\mathsf{Sum},\mathbb{G}}) = O(\log |\mathbb{G}|)$ and $\mathrm{BC}(\Pi_{\mathsf{Sum},\mathbb{G}}) = O(\log |\mathbb{G}|)$.

A.2 Ramp Secret Sharing

Let $\mathbb{K}$ be the minimum finite field such that $|\mathbb{K}| \geq 2n$ and fix $2n$ pairwise distinct elements $\beta_0, \beta_1, \ldots, \beta_{n-1}, \alpha_1, \ldots, \alpha_n \in \mathbb{K}$. Let ℓ be a positive integer such that $\ell \leq n$. Define $\mathsf{RSS}_\ell(\mathbf{s})$ as an algorithm to generate shares of the (t, ℓ, n)-ramp secret sharing scheme [30, 17] for a secret vector $\mathbf{s} \in \mathbb{K}^\ell$. Formally, for $\mathbf{s} \in \mathbb{K}^\ell$, we define a set $\mathcal{R}_{\mathbf{s}}$ of polynomials as

$$\mathcal{R}_{\mathbf{s}} := \{\varphi \in \mathbb{K}[X] : \deg \varphi \leq t + \ell,\ (\varphi(\beta_0), \ldots, \varphi(\beta_{\ell-1})) = \mathbf{s}\}$$

On input $\mathbf{s} \in \mathbb{K}^\ell$, $\mathsf{RSS}_\ell(\mathbf{s})$ chooses a polynomial φ uniformly at random from $\mathcal{R}_{\mathbf{s}}$, and then outputs $(\varphi(\alpha_1), \ldots, \varphi(\alpha_n))$.

Basic mathematical facts regarding RSS_ℓ were shown in [16].

Lemma 5. *Let $T \subseteq [n]$ be any set of size at most t and $\mathbf{s} \in \mathbb{K}^\ell$. Then, there is a polynomial $\Delta_{\mathbf{s}} \in \mathcal{R}_{\mathbf{s}}$ such that $\Delta_{\mathbf{s}}(\alpha_i) = 0$ for all $i \in T$.*

Lemma 6. *Let $\mathbf{s}, \mathbf{u} \in \mathbb{K}^\ell$ and $\varphi_{\mathbf{s}} \in \mathcal{R}_{\mathbf{s}}$. If $\varphi_{\mathbf{u}}$ is uniformly distributed over $\mathcal{R}_{\mathbf{u}}$, then $\varphi_{\mathbf{s}} + \varphi_{\mathbf{u}}$ is uniformly distributed over $\mathcal{R}_{\mathbf{s}+\mathbf{u}}$.*

Lemma 7. *Let $\mathbf{s} = (s_0, \ldots, s_{\ell-1}) \in \mathbb{K}^\ell$. Then, there is an algorithm $\mathsf{Reconst}_\ell$ such that*

$$\sum_{i \in [n]} \mathsf{Reconst}_\ell(j, i; v_i) = s_j,\ \forall j = 0, 1, \ldots, \ell - 1$$

for any possible shares $(v_1, \ldots, v_n) \leftarrow \mathsf{RSS}_\ell(\mathbf{s})$. Furthermore, $\mathsf{Reconst}_\ell$ is linear in the sense that $\mathsf{Reconst}_\ell(j, i; v) + \mathsf{Reconst}_\ell(j, i; v') = \mathsf{Reconst}_\ell(j, i; v + v')$ for any $v, v' \in \mathbb{K}$.

We can construct a deterministic algorithm $\mathsf{FixedShare}_\ell$ that outputs predetermined shares consistent with a given secret vector. Formally, we fix a deterministic algorithm $\mathsf{FixedSample}_\ell$ which on input $\mathbf{s} \in \mathbb{K}^\ell$, computes a polynomial $\psi_{\mathbf{s}} \in \mathcal{R}_{\mathbf{s}}$. It can be implemented efficiently, e.g., with Gaussian elimination. Define $\mathsf{FixedShare}_\ell$ as follows: On input $i \in [n]$ and $\mathbf{s} \in \mathbb{K}^\ell$, $\mathsf{FixedShare}_\ell(i, \mathbf{s})$ computes $\psi_{\mathbf{s}} = \mathsf{FixedSample}_\ell(\mathbf{s})$ and outputs $\psi_{\mathbf{s}}(\alpha_i)$. Note that $(\mathsf{FixedShare}_\ell(i, \mathbf{s}))_{i \in [n]}$ is a tuple of possible shares of a secret vector $\mathbf{s}$.

A.3 A Function-private MPC Protocol for $\mathcal{S}'_d$

A function $g : \mathbb{G}^n \to \{0,1\}$ is called an abelian program over an abelian group $\mathbb{G}$ if there exists a function $f : \mathbb{G} \to \{0,1\}$ such that $g(x_1,\dots,x_n) = f(x_1+\cdots+x_n)$ for all $(x_1,\dots,x_n) \in \mathbb{G}^n$, where addition is taken over $\mathbb{G}$ [2]. Let $\mathcal{A}_{\mathbb{G}}$ be the set of all abelian programs over $\mathbb{G}$ and let $\mathcal{A}'_{\mathbb{G}} = \{g_h : h \in \mathcal{A}_{\mathbb{G}}\}$, where g_h is a function that takes $(x_i)_{i\in[n]} \in X^n$ as input and distributes $y = h(x_1,\dots,x_n)$ to all parties. As pointed out in [2], abelian programs can compute a symmetric function $h : [d]^n \to \{0,1\}$ by setting $\mathbb{G} = \mathbb{Z}_{n+1}^{d-1}$ and encoding each input $x_i \in [d]$ into $x'_i \in \mathbb{G}$ as

$$x'_i = \begin{cases} \mathbf{e}_{x_i}, & \text{if } 1 \le x_i < d, \\ \mathbf{0}, & \text{if } x_i = d, \end{cases}$$

where $\mathbf{e}_j$ is the vector whose entry is 1 at position j and 0 otherwise, and $\mathbf{0}$ is the zero vector. From this point of view, we focus on showing an MPC protocol for $\mathcal{A}'_{\mathbb{G}}$.

Theorem 5. *Let ℓ be any integer such that $\ell \le |\mathbb{G}|$, and suppose that $t \le n - \lceil |\mathbb{G}|/\ell \rceil$. The protocol Π_{Sym} described in Fig. 1 is a t-input-function-private MPC protocol for $\mathcal{A}'_{\mathbb{G}}$ in the $(\mathcal{F}_{\mathsf{Sum},\mathbb{G}}, \mathcal{F}_{\mathsf{Sum},\mathbb{K}})$-hybrid model.*

Proof. First, we prove the correctness of Π_{Sym}. Let $\mathbf{x} = (x_i)_{i\in[n]} \in \mathbb{G}^n$ be any input. Since $r = \sum_{i\in[n]} r_i$, it holds that $a_{y'} = y = r + \sum_{i\in[n]} x_i$. Since $(v_i^{(j)})_{i\in[n]}$ are shares of RSS_k for a secret vector $\mathbf{U}^{(j)}$, it also holds that

$$z = \sum_{i\in[n]} \mathsf{Reconst}_k(\tau, i; v_i^{(\sigma)}) = \mathbf{U}^{(\sigma)}[\tau] = \mathbf{S}[\sigma k + \tau] = \mathbf{S}[y'] = f(a_{y'} - r) = f(y - r)$$

where $\mathbf{U}^{(\sigma)}[\tau]$ is the τ-th element of $\mathbf{U}^{(\sigma)}$. Therefore, we have that $z = f(\sum_{i\in[n]} x_i) = h(x_1,\dots,x_n)$.

Next, we prove the input-function-privacy of Π_{Sym}. Let $T \subseteq [n]$ be the set of t corrupted parties. Let $H = [n] \setminus T$ be the set of honest parties. Let $\mathbf{x} = (x_i)_{i\in[n]}, \widetilde{\mathbf{x}} = (\widetilde{x}_i)_{i\in[n]} \in \mathbb{G}^n$ be any pair of inputs and $h, \widetilde{h}$ be any pair of abelian programs such that $x_i = \widetilde{x}_i$ $(\forall i \in T)$ and $h(x_1,\dots,x_n) = \widetilde{h}(\widetilde{x}_1,\dots,\widetilde{x}_n)$. Let $f, \widetilde{f} : \mathbb{G} \to \{0,1\}$ be the functions corresponding to the abelian programs $h, \widetilde{h}$, respectively.

Note that in the $(\mathcal{F}_{\mathsf{Sum},\mathbb{G}}, \mathcal{F}_{\mathsf{Sum},\mathbb{K}})$-hybrid model, corrupted parties' view can be simulated from the following elements since the other elements are locally computed from them:

Correlated randomness. $(r_i, v_i^{(0)}, \dots, v_i^{(\ell-1)})$ for all $i \in T$;
Online messages. $y = \sum_{i\in[n]} x_i + r$ and z.

It is sufficient to prove that the distribution of the above elements during the execution of Π_{Sym} on input $\mathbf{x}$ and h is identical to that on input $\widetilde{\mathbf{x}}$ and $\widetilde{h}$.

Protocol Π_{Sym}

Notations.
- Let $\mathbb{G} = \{a_0, \ldots, a_{N-1}\}$ be an abelian group.
- Let $\ell \leq N$, $k := \lceil N/\ell \rceil$ and $m := \ell k$.

Input. Each party P_i has $x_i \in \mathbb{G}$.

Output. Every party obtains $z = h(x_1, \ldots, x_n)$.

Correlated randomness. Given an abelian program $h : \mathbb{G}^n \to \{0,1\}$,

1. Let $f : \mathbb{G} \to \{0,1\}$ be a function such that $h(x_1, \ldots, x_n) = f(\sum_{i\in[n]} x_i)$ for all $(x_1, \ldots, x_n) \in \mathbb{G}^n$.
2. Let $r \leftarrow\!\!\$\, \mathbb{G}$ and $(r_i)_{i\in[n]} \leftarrow \mathsf{Additive}_{\mathbb{G}}(r)$.
3. Define $\mathbf{S} = (\mathbf{S}[i])_{0\leq i\leq m-1} \in \mathbb{K}^m$ as
$$\mathbf{S}[i] = \begin{cases} f(a_i - r), & \text{if } 0 \leq i < N, \\ 0, & \text{otherwise.} \end{cases}$$
4. Decompose $\mathbf{S}$ into ℓ vectors $\mathbf{U}^{(0)}, \ldots, \mathbf{U}^{(\ell-1)}$ of dimension k, i.e., $\mathbf{S} = (\mathbf{U}^{(0)}, \ldots, \mathbf{U}^{(\ell-1)})$.
5. For each $j = 0, 1, \ldots, \ell-1$, let $(v_i^{(j)})_{i\in[n]} \leftarrow \mathsf{RSS}_k(\mathbf{U}^{(j)})$.
6. Each party P_i receives $(r_i, v_i^{(0)}, \ldots, v_i^{(\ell-1)})$.

Protocol.

1. Each party P_i computes $y_i = x_i + r_i$ over $\mathbb{G}$.
2. Parties obtain $y = \mathcal{F}_{\mathsf{Sum},\mathbb{G}}((y_i)_{i\in[n]})$.
3. Each party computes $y' \in \{0, 1, \ldots, N-1\}$ such that $a_{y'} = y$.
4. Each party computes σ, τ such that $\sigma \in \{0, 1, \ldots, \ell-1\}$, $\tau \in \{0, 1, \ldots, k-1\}$, and $y' = \sigma k + \tau$.
5. Each party P_i computes $z_i = \mathsf{Reconst}_k(\tau, i; v_i^{(\sigma)})$.
6. Parties obtain $z = \mathcal{F}_{\mathsf{Sum},\mathbb{K}}((z_i)_{i\in[n]})$.
7. Each party P_i outputs z.

Fig. 1. A protocol Π_{Sym}

To show the equivalence of the distributions, we show a bijection between the random strings used by Π_{Sym} on input $\mathbf{x}$ and h and the random strings used by Π_{Sym} on input $\widetilde{\mathbf{x}}$ and $\widetilde{h}$ such that the correlated randomness and the online messages received by T are the same under this bijection. The set of all random strings is

$$\mathcal{Q}_f = \left\{ \left((r_i)_{i\in[n]}, \phi^{(0)}, \ldots, \phi^{(\ell-1)} \right) : r_i \in \mathbb{G},\ \phi^{(j)} \in \mathcal{R}_{\mathbf{U}^{(j)}} \right\},$$

where $r = \sum_{i\in[n]} r_i$, $(\mathbf{U}^{(0)}, \ldots, \mathbf{U}^{(\ell-1)}) = \mathbf{S}_{f,r}$ and

$$\mathbf{S}_{f,r}[i] = \begin{cases} f(a_i - r), & \text{if } 0 \le i < N, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

We denote the randomness of Π_{Sym} on input $\mathbf{x}$ and h by $R = ((r_i)_{i\in[n]}, \phi^{(0)}, \ldots, \phi^{(\ell-1)})$ and that on input $\widetilde{\mathbf{x}}$ and $\widetilde{h}$ by $\widetilde{R} = ((\widetilde{r}_i)_{i\in[n]}, \widetilde{\phi}^{(0)}, \ldots, \widetilde{\phi}^{(\ell-1)})$. We consider a bijection that maps the randomness $R \in \mathcal{Q}_f$ to $\widetilde{R} \in \mathcal{Q}_{\widetilde{f}}$ in such a way that

$$\begin{aligned} \widetilde{r}_i &= \begin{cases} r_i, & \text{if } i \in T, \\ r_i + x_i - \widetilde{x}_i, & \text{if } i \in H, \end{cases} \\ \widetilde{\phi}^{(j)} &= \phi^{(j)} + \Delta_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}} \end{aligned}$$

where

$$r := \sum_{i\in[n]} r_i,\ (\mathbf{U}^{(0)}, \ldots, \mathbf{U}^{(\ell-1)}) := \mathbf{S}_{f,r},\ \widetilde{r} := \sum_{i\in[n]} \widetilde{r}_i,\ (\widetilde{\mathbf{U}}^{(0)}, \ldots, \widetilde{\mathbf{U}}^{(\ell-1)}) := \mathbf{S}_{\widetilde{f},\widetilde{r}},$$

and $\Delta_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}} \in \mathcal{R}_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}}$ is a polynomial such that $\Delta_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}}(\alpha_i) = 0$ for all $i \in T$, whose existence is guaranteed by Lemma 5. The image is indeed a consistent random string, i.e., $((\widetilde{r}_i)_{i\in[n]}, \widetilde{\phi}^{(0)}, \ldots, \widetilde{\phi}^{(\ell-1)}) \in \mathcal{Q}_{\widetilde{f}}$, since $\phi^{(j)} \in \mathcal{R}_{\mathbf{U}^{(j)}}$ implies that $\widetilde{\phi}^{(j)} = \phi^{(j)} + \Delta_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}} \in \mathcal{R}_{\widetilde{\mathbf{U}}^{(j)}}$. The above map is indeed a bijection since it has the inverse

$$\begin{aligned} r_i &= \begin{cases} \widetilde{r}_i, & \text{if } i \in T, \\ \widetilde{r}_i + \widetilde{x}_i - x_i, & \text{if } i \in H, \end{cases} \\ \phi^{(j)} &= \widetilde{\phi}^{(j)} - \Delta_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}}. \end{aligned}$$

This bijection does not change the correlated randomness $(r_i, v_i^{(0)}, \ldots, v_i^{(\ell-1)})_{i\in T}$ of T since

$$\widetilde{v}_i^{(j)} = \widetilde{\phi}^{(j)}(\alpha_i) = \phi^{(j)}(\alpha_i) + \Delta_{\widetilde{\mathbf{U}}^{(j)} - \mathbf{U}^{(j)}}(\alpha_i) = \phi^{(j)}(\alpha_i) = v_i^{(j)}$$

for all $i \in T$. It can be seen that $\widetilde{x}_i + \widetilde{r}_i = \widetilde{x}_i + (r_i + x_i - \widetilde{x}_i) = x_i + r_i$ for $i \in H$. In particular, the message y is the same in both executions. Since $h(x_1, \ldots, x_n) = \widetilde{h}(\widetilde{x}_1, \ldots, \widetilde{x}_n)$, the message z is also the same in both executions, which implies that the bijection does not change online messages seen by corrupted parties. $\square$

Recall that $\mathcal{F}_{\mathsf{Sum},\mathbb{G}}$ can be realized by a protocol with bottleneck complexity $O(\log|\mathbb{G}|)$ and correlated randomness of size $O(\log|\mathbb{G}|)$. Thus, we have the following corollary.

Corollary 5. *Let ℓ be any integer such that $\ell \leq |\mathbb{G}|$, and suppose that $t \leq n - \lceil|\mathbb{G}|/\ell\rceil$. Then, there exists a t-input-function-private MPC protocol Π for $\mathcal{A}'_{\mathbb{G}}$ such that $\mathrm{BC}(\Pi) = O(\log|\mathbb{G}| + \log n)$ and $\mathrm{CR}(\Pi) = O(\log|\mathbb{G}| + \ell\log n)$.*

To compute $\mathcal{S}'_d$, we set $\mathbb{G} = \mathbb{Z}_{n+1}^{d-1}$ and hence $\log|\mathbb{G}| = O(d\log n)$.

Corollary 6. *Let $t = n - \epsilon n$, where $0 < \epsilon < 1$. Then, there exists a t-input-function-private MPC protocol Π for $\mathcal{S}'_d$ such that $\mathrm{BC}(\Pi) = O(d\log n)$ and $\mathrm{CR}(\Pi) = O(n^{d-2}\epsilon^{-1}\log n)$.*

In the case of $t = n-1$, we can choose additive secret sharing over the binary field when distributing shares of a function, instead of ramp secret sharing. This reduces the size of correlated randomness by a factor of $\log n$.

Corollary 7. *Let $t = n-1$. Then, there exists a t-input-function-private MPC protocol Π for $\mathcal{S}'_d$ such that $\mathrm{BC}(\Pi) = O(d\log n)$ and $\mathrm{CR}(\Pi) = O(n^{d-1})$.*