Towards Optimal and Efficient Perfectly Secure Message Transmission

Matthias Fitzi[1⋆], Matthew Franklin[2], Juan Garay[3], and S. Harsha Vardhan[4⋆⋆]

[1] Department of Computer Science, ETH Zürich, Switzerland.
fitzi@inf.ethz.ch.
[2] Department of Computer Science, UC Davis, CA 95016.
franklin@cs.ucdavis.edu.
[3] Bell Labs, 600 Mountain Ave., Murray Hill, NJ 07974.
garay@research.bell-labs.com.
[4] Department of Computer Science and Engineering, IIT Madras, India.
harshas@cse.iitm.ernet.in.

Abstract. Perfectly secure message transmission (PSMT), a problem formulated by Dolev, Dwork, Waarts and Yung, involves a sender $\mathcal{S}$ and a recipient $\mathcal{R}$ who are connected by n synchronous channels of which up to t may be corrupted by an active adversary. The goal is to transmit, with perfect security, a message from $\mathcal{S}$ to $\mathcal{R}$. PSMT is achievable if and only if $n > 2t$.

For the case $n > 2t$, the lower bound on the number of communication rounds between $\mathcal{S}$ and $\mathcal{R}$ required for PSMT is 2, and the only known efficient (i.e., polynomial in n) two-round protocol involves a communication complexity of $O(n^3\ell)$ bits, where ℓ is the length of the message. A recent solution by Agarwal, Cramer and de Haan is provably communication-optimal by achieving an asymptotic communication complexity of $O(n\ell)$ bits; however, it requires the messages to be exponentially large, i.e., $\ell = \Omega(2^n)$.

In this paper we present an efficient communication-optimal two-round PSMT protocol for messages of length polynomial in n that is almost optimally resilient in that it requires a number of channels $n \geq (2+\varepsilon)t$, for any arbitrarily small constant $\varepsilon > 0$. In this case, optimal communication complexity is $O(\ell)$ bits.

1 Introduction

In the problem of *perfectly secure message transmission* (PSMT) a sender $\mathcal{S}$ and a recipient $\mathcal{R}$ are connected by n distinct, synchronous communication channels. Of these channels, an active adversary may be corrupting any selection of up to t. The goal is to have $\mathcal{S}$ transmit a message to $\mathcal{R}$ perfectly securely, i.e., in such a way that (1) the adversary gets no information about the message, and (2) that $\mathcal{R}$ receives the correct message with probability 1. In general, a protocol for

⋆ Work partly done while at Aarhus University.
⋆⋆ Work partly done at Bell Labs Research, Bangalore, India.

PSMT requires multiple communication exchanges—*rounds*—between $\mathcal{S}$ and $\mathcal{R}$, for example, to first agree on a one-time pad before having the padded message transmitted from $\mathcal{S}$ to $\mathcal{R}$.

PSMT was introduced by Dolev, Dwork, Waarts and Yung in [8]. Their main result is that PSMT is achievable if and only if $n > 2t$. For this particular bound, they also showed that two communication rounds are necessary and sufficient in order to achieve PSMT (i.e., a communication flow from $\mathcal{R}$ to $\mathcal{S}$, and then a flow from $\mathcal{S}$ to $\mathcal{R}$). However, their protocol to achieve this bound is inefficient as it involves an exponential (in n) computation and communication overhead. In [17], Sayeed and Abu-Amara gave polynomial-time two-round protocol that requires a communication complexity of $O(n^3\ell)$ bits, where ℓ is the length of the message to be transmitted. More recently, Srinathan, Narayanan and Rangan [18] showed that, in order to achieve two-round PSMT, $\Omega(n\ell)$ bits must be communicated. This lower bound has been matched by the protocol by Agarwal, Cramer and de Haan [1], at the price, however, of requiring messages of length exponential in n. In [16], Patra, Choudhary, Srinathan and Rangan show that by using one additional round (i.e., three rounds in total), this communication bound can be achieved with polynomial message length.

Our contributions. In this paper, we present an efficient two-round protocol for PSMT with optimal communication complexity that works for messages of length polynomial in n. The protocol works for any parameterization of $n \geq (2+\varepsilon)t$, where $\varepsilon > 0$ is a fixed but arbitrarily small constant—i.e., the protocol is almost optimally resilient. Note, however, that our protocol is optimally resilient with respect to the communication complexity we achieve: $O(\ell)$, where ℓ is the length of the message—as it follows from the lower bound in [18] that $n = 2t + \Omega(t)$ is necessary in order to achieve communication complexity $O(\ell)$ (in contrast to $\Omega(n\ell)$ for the general case $n > 2t$).

Our protocol is derived from a modification of the communication-optimal one-round PSMT protocol for $n > 3t$ in [17], and by applying a technique that we call *player virtualization*, which can be viewed as a very simple and constructive instantiation of so-called *Bracha assignments* [6], which are used to "amplify" the resilience of a distributed computation protocol while preserving some of its other properties. (We describe this technique in more detail below.)

Additionally, we also show a tight bound on the communication complexity of one-round PSMT for $n > 3t$.

The "player virtualization" technique. The idea of creating *virtual players* whose behavior is simulated by the actions of groups of real players was introduced by Bracha in [6] in the context of Byzantine agreement [14], in order to prove the existence of a randomized protocol for the problem for any $n > (3+\delta)t$, where n is the total number of players, t is the number of faulty players, and $\delta > 0$ is an arbitrary constant, running in expected $O(\log n)$ rounds. The goal was to simulate Ben-Or's randomized distributed coin-flipping protocol [2], which required, for good performance, that the number of faulty players be at most $O(\sqrt{n})$—i.e., the effect of the simulation is to obtain a set of virtual players with

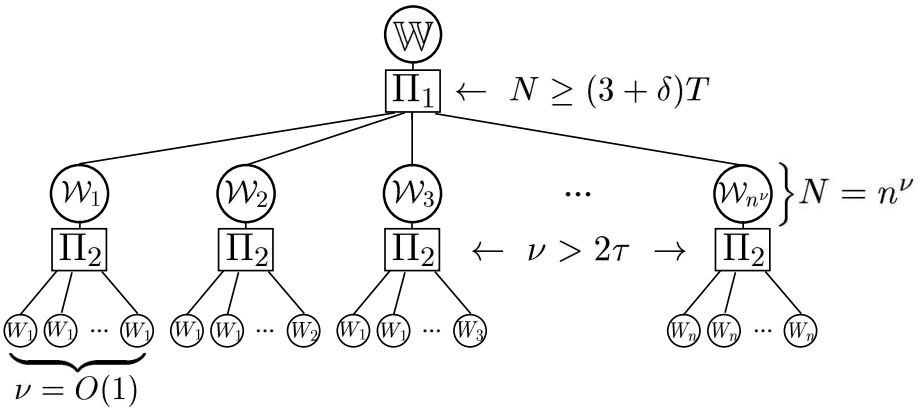

Fig. 1. *The wire virtualization scheme for PSMT.*

a lower corruption rate than in the original player set. While Bracha was able to prove the existence of such a protocol, the result is non-constructive.

A similar—but perhaps simpler—idea, also applied in the context of Byzantine agreement, is to partition the player set into smaller, non-overlapping "committees" (e.g., [5, 7]), with the goal of obtaining at least one out of the several subsets of players that maintains the global corruption ratio (t/n). This approach, however, typically has the converse effect of the set of committees having a higher corruption rate than the original player set.

In the context of secure multi-party computation [19, 10], Hirt and Maurer [11] essentially applied player virtualization in order to reduce a generalized-adversary computation to threshold-adversary computations of a small size. Their construction, however, generally yields protocols with exponential (in n) computation and communication complexities.

Constructive Bracha assignments have also been used for the leader election problem in the full information model [15, 20], and recently in order to reduce the communication (to polylogarithmic in n) required for the task [13] ([13] also studies "almost-everywhere agreement" [9] under reduced communication). At a high level, these constructions are based on expander graphs, and typically carry a probability of error. We elaborate more on this type of approach in Section 5.

We now give a high-level description of how we apply player (more precisely, "wire") virtualization to PSMT. Recall that we are given $\mathcal{S}$ and $\mathcal{R}$ who are connected by n wires of which t might be corrupted by the adversary. We first observe the following facts about PSMT:

1. For any $N \geq 3(T+\delta)$, where $\delta > 0$ is a constant and N denotes the total number of wires and T the number of possibly corrupted wires, there is a one-round PSMT protocol Π_1 with constant communication overhead. Such a protocol is described in Section 3.1.
2. For any $\nu > 2\tau$, where ν denotes the total number of wires and τ the number of possibly corrupted wires, there is a two-round PSMT protocol Π_2 that is

communication-optimal but requires messages of exponential size in ν. This is the protocol in [1].

The basic idea now is to run an instance of protocol Π_1 wherein the N wires are simulated by instances of protocol Π_2 among different selections of ν wires. In particular, we can apply protocol Π_2 to any subset of $\nu < n$ physical wires. If a strict minority of the wires happens to be corrupted then the resulting protocol will simulate an uncorrupted "virtual wire;" if not, then the virtual wire will behave like a corrupted physical wire. As a result, such a virtual wire can now be abstractly used as an additional wire by a "higher-level" PSMT protocol.

Our goal is to generate N virtual wires with the help of protocol Π_2 such that, independently of which t physical wires are corrupted, at most $T \leq N/(3+\delta)$ of the virtual wires can act as if they were corrupted. Once we achieve this, we can simply apply protocol Π_1 on the set of N virtual wires. As can be easily seen, this construction preserves round complexity 2. However, in order to also maintain $\mathrm{poly}(n)$ efficiency when running the protocols Π_1 and Π_2, we need the additional constraints $N = \mathrm{poly}(n)$ and $\nu = O(\log n)$.

We meet these constraints by choosing $\nu = O(1)$ and having each possible set of ν physical wires (including repetitions) simulate a different virtual wire, resulting in $N = n^\nu$. The approach is depicted in Figure 1. As we show in the sequel, it turns out that this construction works for any parameterization of $n \geq (2+\varepsilon)t$, where $\varepsilon = \Omega(1)$; i.e., round-optimal, bit-optimal and efficient PSMT with almost optimal resiliency can be achieved in this way.

Organization of the paper. In the next section we present the model and the definition of the PSMT problem. We dedicate Section 3 to the treatment of the one-round case. We first present an efficient PSMT protocol for $n > 3t$ wires which, as we also show, has optimal communication overhead. Design and analysis of the virtualization construction yielding our main result are presented in Section 4. We conclude in Section 5 with some optimization considerations and final remarks.

2 Model and Definitions

Sender $\mathcal{S}$ and recipient $\mathcal{R}$ are connected by n distinct synchronous channels ("wires") $W_1, W_2, \ldots, W_n$. An adversary $\mathcal{A}$ may select up to t of the n wires and corrupt them actively, i.e., $\mathcal{A}$ may eavesdrop on the selected wires as well as change the messages being sent on them. The adversary is assumed to be computationally unbounded. Furthermore, the adversary is assumed to be *adaptive*, i.e., it can adaptively decide on which further wires to corrupt at any point during the protocol — but "non-mobile," i.e., the adversary is not allowed to have corrupted any more than t different wires by the end of the protocol, overall.

Definition 1. *A protocol between $\mathcal{S}$ and $\mathcal{R}$, based on local computation and communication via the network described above, achieves* perfectly secure message transmission (PSMT) *if it transmits a message from $\mathcal{S}$ to $\mathcal{R}$ such that the following two conditions are satisfied:*

Protocol 1-PSMT(n, t, m)

- Given a message $m = [m_1 m_2 \ldots m_k]$ $(k = n - 3t)$, the sender $\mathcal{S}$ randomly forms a polynomial $f(x)$ of degree at most $d = (n - 2t - 1)$ by choosing its coefficients as follows:

$$\text{coeff}(x^i) = \begin{cases} m_{i+1}, & \text{if } 0 \leq i < k, \\ c_{i-k}, & \text{if } k \leq i < (k+t), \end{cases}$$

  where the c_{i-k}'s are chosen uniformly at random from $\mathbb{F}$.
- On wire W_j, $1 \leq j \leq n$, the sender $\mathcal{S}$ sends the share $r_j = f(\alpha^{j-1})$, where α is a generator of the multiplicative group of $\mathbb{F}$.
- The recipient $\mathcal{R}$ uses the Welch-Berlekamp decoding algorithm [4] on the received values in order to obtain the message.

Fig. 2. *One-round PSMT with low communication overhead.*

PRIVACY: *$\mathcal{A}$ does not get any information about the message being transmitted.*
CORRECTNESS: *$\mathcal{R}$ gets full information about the message transmitted by $\mathcal{S}$; i.e., $\mathcal{R}$ learns the message with probability* 1. ◇

In the sequel, and without loss of generality, we assume that the messages are taken from a finite field $\mathbb{F}$ with $|\mathbb{F}| > n$.

We define the *bit-communication complexity* (or, *communication complexity*, for short) of a PSMT protocol to be the total number of bits being communicated between $\mathcal{S}$ and $\mathcal{R}$. For convenience, we also define the *communication overhead*, Λ, as the total number of bits communicated by the protocol divided by the length of the message. The *round complexity* of a PSMT protocol is its number of subsequent communication rounds between $\mathcal{S}$ and $\mathcal{R}$. In particular, a one-round PSMT protocol consists of a synchronous flow of communication on the wires from $\mathcal{S}$ to $\mathcal{R}$, and a two-round PSMT protocol has a synchronous flow from $\mathcal{R}$ to $\mathcal{S}$ followed by a synchronous flow from $\mathcal{S}$ to $\mathcal{R}$.

3 One-Round PSMT with Low Communication Overhead

In this section, we extend the one-round PSMT protocol in [8, 17] for $n = 3t+1$ to handle any $n > 3t$ with low communication overhead—in fact, exactly $\Lambda = \frac{n}{n-3t}$, which, as we also show, is optimal for this case.

3.1 Protocol 1-PSMT

At a high level, the PSMT protocols in [8, 17] hide the message to be transmitted using the approach in [3] of verifiable secret sharing (VSS) over a finite field $\mathbb{F}$ using Reed-Solomon codes. In contrast to their solutions, instead of hiding the message in one single coefficient of the polynomial, we split the message into "pieces" and assign each piece to a separate coefficient, and correspondingly

increase the degree of the polynomial. Effectively, this allows us to hide $n - 3t$ different field elements in one VSS instance.

In more detail, assuming an adequate field size[5], the message is interpreted as a sequence of $k = n - 3t$ field elements, and transmitted using the protocol of Figure 2. We are able to show:

Theorem 1. *Protocol 1-PSMT(n, t, m) is a one-round PSMT protocol for any $n > 3t$ with communication overhead $\Lambda = \frac{n}{n-3t}$.*

Proof (sketch).

CORRECTNESS: Since $n > d + 2t$, $\mathcal{R}$ can decode the complete polynomial, compute the low-degree coefficients m_i, $1 \leq i \leq k$, and extract the full message m.

PRIVACY: Since $f(x)$ is of degree $d = t + (k-1)$, any t shares of the form $f(\alpha^j)$ are independent from the k coefficients m_i. Thus, $\mathcal{A}$ gets no information about m.

COMMUNICATION OVERHEAD: The protocol communicates n field elements in order to transmit a secret message consisting of $k = n - 3t$ field elements. Thus, the communication overhead of the protocol is $\Lambda = \frac{n}{n-3t}$. □

The following corollary will be useful for our main virtualization result in Section 4.

Corollary 1. *One-round PSMT with* constant *communication overhead is possible for $n = (3 + \delta)t$, for any constant $\delta > 0$.*

As we now show, the communication overhead of our one-round PSMT protocol is in fact optimal. The reader intrigued by the use of 1-PSMT in our virtualization scheme is invited to proceed directly to Section 4.

3.2 Communication lower bound for one-round PSMT

In [18], Srinathan, Narayanan and Rangan established a lower bound on the communication overhead (of $\Lambda \geq \frac{n}{n-2t}$) for two-round PSMT. In this section we show a lower bound of $\Lambda \geq \frac{n}{n-3t}$ for one-round PSMT when $n > 3t$. Note that one-round PSMT is impossible if $n \leq 3t$.

Theorem 2. *Any one-round PSMT protocol for $n > 3t$ wires requires communication overhead $\Lambda \geq \frac{n}{n-3t}$.*

Proof. Let $\mathcal{M}$ be the message space from where the sender $\mathcal{S}$'s message is drawn. Let $\mathbf{T}_i^m$ denote the set of all possible transmissions that can occur on wire $W_i \in \{W_1, \ldots, W_n\}$ when $\mathcal{S}$ transmits message m. Furthermore, for $j \geq i$, let $\mathbf{M}_{i,j}^m \subseteq \mathbf{T}_i^m \times \mathbf{T}_{i+1}^m \times \cdots \times \mathbf{T}_j^m$ denote the set of all possible transmissions that can occur on the wires in $\{W_i, W_{i+1}, \ldots, W_j\}$ when $\mathcal{S}$ transmits message m. Finally,

[5] Alternatively, we would first split the message into blocks, and then transmit each block separately.

let $\mathbf{M}_{2t+1,n} = \bigcup_{m\in\mathcal{M}} \mathbf{M}^m_{2t+1,n}$, and $\mathbf{T}_i = \bigcup_{m\in\mathcal{M}} \mathbf{T}^m_i$, and let us call $\mathbf{T}_i$ the *capacity of wire* W_i and $\mathbf{M}_{k,\ell}$ the *capacity of the set of wires* $\{W_k, W_{k+1}, \ldots, W_\ell\}$.

Consider any one-round PSMT protocol for $n > 3t$. Perfect privacy requires that the transmissions on any t wires be independent of the message. Thus, for any two messages $m_1, m_2 \in \mathcal{M}$ it must hold that

$$\mathbf{M}^{m_1}_{2t+1,3t} = \mathbf{M}^{m_2}_{2t+1,3t} \ .$$

(The above must hold for any selection of t wires; we focus on the set $\{W_{2t+1}, \ldots, W_{3t}\}$ for simplicity.) Furthermore, perfect correctness implies that the (uncorrupted) transmissions on any $n-2t$ wires must uniquely determine the message. Thus, it must also hold that

$$\mathbf{M}^{m_1}_{2t+1,n} \cap \mathbf{M}^{m_2}_{2t+1,n} = \emptyset \ .$$

Since $\mathbf{M}^m_{2t+1,3t}$ may be the same for every message m, it follows that

$$\prod_{i=3t+1}^{n} |\mathbf{T}_i| \geq |\mathbf{M}_{3t+1,n}| \geq |\mathcal{M}| \ .$$

Let $d = n - 3t$. More generally, the above inequality holds for any selection of d wires $\mathcal{D} \subset \{W_1, W_2, \ldots, W_n\}$, $|\mathcal{D}| = d$, i.e., $\prod_{W_i\in\mathcal{D}} |\mathbf{T}_i| \geq |\mathcal{M}|$, and in particular it holds for every selection $\mathcal{D}_k = \{W_{(kd+1) \bmod n}, W_{(kd+2) \bmod n}, \ldots, W_{(kd+d) \bmod n}\}$, with $k \in \{0, 1, \ldots, n-1\}$.

If we consider all sets $\mathcal{D}_k$ separately, then each wire is accounted for exactly d times. Thus, the product of the capacities of all $\mathcal{D}_k$ yields the capacity of the full wire set to the d-th power, and since each $\mathcal{D}_k$ has capacity at least $|\mathcal{M}|$, we get

$$|\mathcal{M}|^n \leq \prod_{k=0}^{n-1} \prod_{W_j\in\mathcal{D}_k} |\mathbf{T}_j| = \left(\prod_{i=1}^{n} |\mathbf{T}_i|\right)^d \ ,$$

and therefore

$$\Lambda \geq \frac{\sum_{i=1}^{n} log|\mathbf{T}_i|}{log|\mathcal{M}|} \geq \frac{n}{d} = \frac{n}{n-3t} \ .$$

□

4 Communication-Optimal Two-Round PSMT for $n \geq (2+\varepsilon)t$

In this section, we use wire virtualization and protocol 1-PSMT from the previous section to construct our new two-round PSMT protocol.

4.1 The wire virtualization construction

Let $n \geq (2+\varepsilon)t$ for some $\varepsilon > 0$. Let Π_2 be the communication-optimal (but inefficient) two-round PSMT protocol in [1] (or even the communication-suboptimal protocol in [17]) for ν wires tolerating $\tau = \lfloor \frac{\nu-1}{2} \rfloor$ corrupted wires, where $\nu = O(1)$ (ν will be quantified later, based on the analysis below). Choosing $\nu = O(1)$ implies that protocol Π_2's communication overhead is constant, and thus that Π_2 is communication-optimal.

Further, let Π_1 be 1-PMST, the communication-optimal one-round protocol from Section 3.1 for N wires tolerating $T \leq \frac{N}{3+\delta}$ corrupted wires for some fixed constant $\delta > 0$ (where $N = n^\nu$; see below).

We start by forming all $N = n^\nu$ possible virtual wires $\mathcal{W}_1, \ldots, \mathcal{W}_{n^\nu}$ involving ν wires from the set of real wires $W = \{W_1, \ldots, W_n\}$, allowing repetitions. We call this collection of virtual wires $\mathcal{W}$, $\mathcal{W} = \{\mathcal{W}_1, \ldots, \mathcal{W}_{n^\nu}\}$. We can apply protocol Π_2 to any element of $\mathcal{W}$ with the effect that it will achieve PSMT as long as at most $\tau = \lfloor \frac{\nu-1}{2} \rfloor$ of the involved real wires are actually corrupted. We thus call a virtual wire *correct* when it involves at most τ corrupted real wires and *corrupted* otherwise. Let T be the number of corrupted virtual wires in $\mathcal{W}$.

Our goal now is to find a constant ν such that of all $N = n^\nu$ possible virtual wires out of $\mathcal{W}$, at most $T = \frac{N}{3+\delta}$ are corrupted. This will then allow us to apply one-round protocol Π_1 to the N virtual wires where, in turn, every virtual wire is simulated by the two-round protocol Π_2 (see Figure 1). The analysis in the next section will yield constant ν.

4.2 Virtualization analysis

We consider the following random experiment in order to give a (deterministic) estimation on the ratio of corrupted virtual wires.

Let ν be fixed. Let p be the probability that, picking one of the $N = n^\nu$ possible ν-tuples of n real wires uniformly at random, the respective virtual wire is corrupted. If this probability is at most $\frac{T}{N} = \frac{1}{3+\delta}$ then, clearly, at most $T = \frac{N}{3+\delta}$ virtual wires are corrupted — which is tolerated by protocol Π_1.

For this, we consider random variable $X \in \{0, \ldots, \nu\}$ denoting the number of corrupted wires in the selection. Let P be the probability distribution induced by the following random experiment: pick a wire out of W uniformly at random, repeat this ν times, and let the resulting selection of wires form a tuple of size ν (i.e., a virtual wire).

Our goal is to show that there is a constant ν such that $p = \Pr(X \geq \nu/2) \leq \frac{1}{3+\delta}$, and thus, that the number of actual corrupted virtual wires in Π_1 is at most $T = \frac{N}{3+\delta}$. We achieve this with help of the Chernoff bound (see Appendix A).

According to the process associated with P, let X_i be the 0-1 distributed random variable describing whether the i-th chosen wire is corrupted. Then $X = \sum_{i=1}^{\nu} X_i$. We demand

$$\Pr\left(X \geq \frac{\nu}{2}\right) \leq \frac{1}{3+\delta} \,.$$

Since, clearly, the random variables $X_1, \ldots, X_\nu$ are independent, we can estimate this probability by the Chernoff bound (Equation 2) as

$$\Pr\left(X \geq \frac{\nu}{2} = \lambda\,\mu\nu = \lambda\,\frac{\nu}{2+\varepsilon}\right) \leq e^{-\frac{\nu}{2(2+\varepsilon)}(\lambda-1)^2} \quad \text{where} \quad \lambda = \frac{2+\varepsilon}{2}\ ,$$

and demand

$$e^{-\frac{\nu}{2(2+\varepsilon)}(\lambda-1)^2} \stackrel{!}{\leq} \frac{1}{3+\delta}\ .$$

We thus require that

$$\frac{\nu}{2(2+\varepsilon)}(\lambda-1)^2 = \frac{\nu}{8(2+\varepsilon)}\varepsilon^2 \geq \ln(3+\delta)\ ,$$

which yields

$$\nu \geq \left\lceil \frac{8\ln(3+\delta)(2+\varepsilon)}{\varepsilon^2} \right\rceil\ , \tag{1}$$

obtaining a lower-bound estimation on ν depending on constants ε and δ, where ε is an input parameter and δ is any positive constant of free choice.

Theorem 3. *The construction described above is a two-round PSMT protocol for any $n \geq (2+\varepsilon)t$, $\varepsilon > 0$, and has constant communication overhead, which is optimal.*

Proof (sketch).

CORRECTNESS AND PRIVACY. Correctness and privacy of the protocol follow from the above quantitative analysis and from the respective properties of protocols Π_1 and Π_2.

NUMBER OF ROUNDS. The top-level protocol Π_1 is one-round and operates on virtual wires. Every virtual wire can be independently simulated in parallel by the two-round protocol Π_2. Thus, the resulting protocol involves two communication rounds.

COMMUNICATION OVERHEAD. Protocol Π_2 operates on ν real wires. Since $\nu = O(1)$, the protocol has constant communication overhead. Protocol Π_1 operates on $N = n^\nu$ virtual wires and also has constant communication overhead since we have $T = \frac{N}{3+\delta}$. Thus Π_1 involves N messages of size $\frac{\ell}{N} \cdot O(1)$ which are each transmitted by an instance of protocol Π_2 with constant communication overhead, resulting in the total communication of $N \cdot \frac{\ell}{N} \cdot O(1) = O(\ell)$ bits — or communication overhead $\Lambda = O(1)$ — matching the lower bound for two-round PSMT established in [18]. □

5 Conclusions

In this paper, we presented a communication-optimal two-round PSMT protocol for $n \geq (2+\varepsilon)t$ where $\varepsilon > 0$ is an arbitrary, small constant. For the protocol to be

communication-optimal, messages of length only polynomial in n are required. The communication complexity of the protocol is $O(\ell)$.

As it follows from the lower bound in [18], communication complexity $O(\ell)$ can only be achieved if $n = 2t + \Omega(t)$. Thus, our protocol is optimally resilient under the constraint of communication complexity $O(\ell)$. Our protocol is constructed along the lines of Bracha's player-virtualization technique, systematically extending the player set in order to amplify the resilience of a lower-level protocol.

We also obtained a tight bound on the communication complexity of one-round PSMT for $n > 3t$.

Regarding optimizations to our construction, note that our estimation on ν is rather conservative since it is based on a rough Chernoff-bound estimation. Experiments computing minimal values ν for particular values of ε show that much better results can be achieved. However, depending on the particular value of ε, our construction may still demand the message size to be a polynomial in n of high degree. For example, $\varepsilon = .6$ yields $\nu = 3$, $\varepsilon = .3$ yields $\nu = 11$, while $\varepsilon = .1$ yields $\nu = 83$.

In some cases, variations of the given construction achieve better results — for example, by setting $\nu = 3$ and applying virtualization recursively. Another possibility, at least in order to non-constructively prove the existence of protocols for smaller message sizes, is to have $\nu = \Theta(\log n)$ and proceed along the lines of Bracha [6]. We note that in this case constant communication overhead can still be achieved while requiring low-level protocol Π_2 to be of lower-than-optimal resilience, i.e., $\nu \geq (2 + \alpha)\tau$, where $\alpha > 0$ is a constant. Yet another direction worth investigating in order to achieve a lower number of virtual wires, as suggested by one of the reviewers, would be a "de-randomized" choice of sets obtained from short walks on low-degree expander graphs.

Acknowledgements

We thank the anonymous reviewers for *TCC* for their many useful comments. The work of Matthias Fitzi and Matt Franklin was partly supported by a David and Lucile Packard Fellowship for Science and Engineering.

References

1. S. Agarwal, R. Cramer, and R. de Haan. Asymptotically optimal two-round perfectly secure message transmission. In *Advances in Cryptology: CRYPTO '06*. Springer-Verlag, 2006.
2. M. Ben-Or. Another advantage of free choice: Completely asynchronous agreement protocols. In *Proceedings of the 2nd ACM Symposium on Principles of Distributed Computing (PODC '83)*, pages 17–19. ACM, 1983.
3. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC '88)*, pages 1–10. Springer-Verlag, 1988.

4. E. Berlekamp and L. Welch. Error correction of algebraic block codes. US Patent 4,633,470.
5. P. Berman, J. A. Garay, and K. J. Perry. Bit optimal distributed consensus. In *Computer Science Research*, pages 313–322. Plenum Publishing Corporation, 1992.
6. G. Bracha. An $O(\log n)$ expected rounds randomized Byzantine generals protocol. *Journal of the Association for Computing Machinery*, 34(4):910–920, Oct. 1987.
7. B. A. Coan and J. L. Welch. Modular construction of a Byzantine agreement protocol with optimal message bit complexity. *Information and Computation*, 97(1):61–85, Mar. 1992.
8. D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly secure message transmission. *Journal of the ACM*, 40(1):17–47, Jan. 1993.
9. C. Dwork, D. Peleg, N. Pippinger, and E. Upfal. Fault tolerance in networks of bounded degree. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing (STOC '86)*, pages 370–379, 1986.
10. O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC '87)*, pages 218–229, 1987.
11. M. Hirt and U. Maurer. Player simulation and general adversary structures in perfect multiparty computation. *Journal of Cryptology*, 13(1):31–60, Winter 2000.
12. W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, Mar. 1963.
13. V. King, J. Saia, V. Sanwalani, and E. Vee. Towards secure and scalable computation in Peer-to-Peer networks. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS '06)*, 2006.
14. L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Trans. Prog. Lang. Syst.*, 4(3):382–401, July 1982.
15. R. Ostrovsky, S. Rajagopalan, and U. Vazirani. Simple and efficient leader election in the full information model. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing (STOC '94)*, pages 234–242, 1994.
16. A. Patra, A. Choudhary, K. Srinathan, and C. Pandu Rangan. Constant phase bit optimal protocols for perfectly secure message transmission. In *Indocrypt '06*, 2006.
17. H. Sayeed and H. Abu-Amara. Efficient perfectly secure message transmission in synchronous networks. *Information and Communication*, 126(1):53–61, 1996.
18. K. Srinathan, A. Narayanan, and C. Pandu Rangan. Optimal perfectly secure message transmission. In *Advances in Cryptology: CRYPTO '04*, volume 3152 of *Lecture Notes in Computer Science*, pages 545–561. Springer-Verlag, 2004.
19. A. C. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science (FOCS '82)*, pages 160–164. IEEE, 1982.
20. D. Zuckerman. Randomness-optimal sampling, extractors, and constructive leader election. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*, pages 286–295, 1996.

A Chernoff Bounds

Chernoff bounds [12] give bounds on the probability that of n independent Bernoulli trials the outcome deviates form the expected value by a given fraction. Here we present the "upper tail" version.

Let X_i $(1 \leq i \leq n)$ be a sequence of independent 0-1 distributed random variables with expected value μ. By $\mathcal{C}(\mu, n, \lambda)$ $(\lambda > 1)$ we denote the probability that, out of n trials, the outcome exceeds the expected value $n\mu$ by a given factor depending on λ. The following inequality, which holds for $1 < \lambda < 2e$, bounds this probability.

$$\mathcal{C}(\mu, n, \lambda) = \Pr\left(\sum_{i=1}^{n} X_i \geq \lambda \mu n\right) \leq e^{-\frac{\mu n(\lambda-1)^2}{2}} \tag{2}$$