General Perfectly Secure Message Transmission Using Linear Codes

Qiushi Yang[⋆] and Yvo Desmedt[⋆⋆]

Department of Computer Science, University College London, UK
{q.yang, y.desmedt}@cs.ucl.ac.uk

Abstract. We study perfectly secure message transmission (PSMT) from a sender S to a receiver R in the general adversary model. In this model, instead of being bounded by a threshold, the Byzantine adversary in a network is characterized by an adversary structure. By regarding monotone general access structures as linear codes, we introduce some new properties that allow us to design efficient PSMT protocols. We give a number of efficient PSMT protocols in both undirected and directed network graphs. These protocols comprehensively improve the transmission complexity of some previous results in this area. More significantly, as all of our protocols are executed in either 3 or 2 rounds, our result is the first, in the context of PSMT in the general adversary model, to have constant round complexity when using interaction.

Keywords: perfectly secure message transmission, adversary structure, linear codes, transmission complexity, round complexity.

1 Introduction

In most of the communication networks, a *sender* S and a *receiver* R are connected by unreliable and distrusted channels. The distrust of the channels is because of the assumption that there exists a *Byzantine adversary* who, with unbounded computational power, can control some nodes on these channels. The aim of *perfect secure message transmission* (PSMT) is to enable a secret message to be transmitted from S to R with perfect *privacy* and *reliability*. That is, the adversary should learn no information about the message, and the receiver R can output the message correctly.

Initial study by Dolev et al. [9] shows that PSMT is possible by applying secure transmission protocols. It assumes a threshold adversary who can control up to t nodes, and hence can control up to t channels. Extensive studies on the threshold model have been carried out ever since (e.g., [7, 22, 2, 15]).

There are many other studies on a more general adversary model, which allow an adversary to control nodes in a less symmetric way. In many cases, using a threshold to model an adversary makes little sense. Indeed, certain platforms

[⋆] The author would like to acknowledge financial support from UCL PhD Studentship.

[⋆⋆] Part of this work was done while funded by EPSRC EP/C538285/1, by BT as BT Chair of Information Security, and by RCIS, AIST, Japan.

	Network graph	RC	TC over 1	TC over ℓ
Kumar et al. [14]	undirected	$O(n)$	$O(hn^2)$	–
Desmedt et al. [8]	directed-1	1	$O(\lvert\mathcal{A}\rvert n)$	–
Yang-Desmedt [24]	directed-2	expo. in $\lvert\mathcal{A}\rvert$	expo. in $\lvert\mathcal{A}\rvert$	–
Our result	undirected	3 (Section 4.1)	$O(hn^2)$	$O(h\ell)$
		2 (Section 4.2)	$O(hn^2)$	$O(hn\ell)$
	directed-2	3 (Section 5.1)	$O(h^2n^2)$	$O(hn\ell)$
		2 (Section 5.2)	$O(h)$	$O(h\ell)$

* RC denotes round complexity and TC denotes transmission complexity.
"TC over 1" is the TC of the PSMT protocol that transmits a single message and "TC over ℓ" is the TC of the protocol that transmits multiple (ℓ) messages, where each message is a field element.
"directed-1" are the directed graphs without feedback, and "directed-2" are those with feedback. h is the length of a codeword and n is the number of critical paths (see Section 3).

Table 1. PSMT in the general adversary model.

are more vulnerable than the others. Also, more hierarchical structures cannot be described by a single adversary. The general adversary model assumes that the adversary is characterized by an adversary structure [11], which consists of a number of subsets of nodes, and the adversary is able to control one of these subsets, instead of any t nodes.

Notable studies on PSMT tolerating adversary structures have been done by Kumar et al. [14] on bi-direction channels, by Desmedt et al. [8] on one-way forward channels, and by both Patra et al. [19] and Yang and Desmedt [24] on mixed forward and feedback channels. However, due to the generality of the adversary structure, the protocols in the previous studies are, in many cases, inefficient in terms of the number of execution rounds[1] (*round complexity*) and the number of *field elements* transmitted (*transmission complexity*). Also some previous results are yet to be further characterized. We shall describe these issues in more detail in Section 3.

Our contributions. In this paper we show how *linear secret sharing schemes* (LSSS) and *linear codes* can be used to design efficient PSMT protocols in the general adversary model. Before we do that, we first show a basic construction of an LSSS and discuss its properties (see Section 2.1). Then we propose a new generalized linear code (see Section 2.2) for the purpose of error-correcting, and also for the purpose of defining *pseudo-basis* and *pseudo-dimension* (see Section 2.3). This follows the idea of Kurosawa and Suzuki [15]. Our study on LSSS and linear codes is shown in Section 2.

Next, in Section 3, we show a further characterization on the problem of PSMT in the general adversary model. We observe that the transmission complexity of most previous PSMT protocols is determined by the number of the *critical paths.* Thus we shall describe the properties of the critical paths that are effectively used (see Section 3.1). Also in this section, we show how our protocols

[1] A round is a transmission from S to R or vice versa.

improve the previous results in terms of *round complexity* (RC) and *transmission complexity* (TC) (see Table 1, which we discuss in detail in Section 3.2). Indeed, not only do we significantly improve the TC of some previous PSMT protocols, but we are also the first to give interactive protocols that have *constant* RC in the studies of the general adversary model. Furthermore, we are also the first to study PSMT over multiple messages in this context.

Section 4 and 5 give our constant round and communication efficient protocols in different network settings. These protocols show comprehensive improvements to the previous results in this area, as shown in Table 1.

2 LSSS and Linear Codes

Secret sharing schemes are key tools in the study of PSMT. Given a set of n participants $P = \{1, \ldots, n\}$, the extensively studied threshold schemes (e.g., Shamir's scheme [20]) allow any subset of $t+1$ participants to learn a secret s, but do not reveal any information of s to any subset of at most t participants. General non-threshold schemes, which realize secret sharing among general *access structures*, are also presented in literature (e.g., Ito et al. [12] and Benaloh and Leichter [3]). A monotone access structure Γ is a family of the subsets of P such that for any set $A \subseteq P$, if $A \in \Gamma$ and $A \subseteq A'$, then $A' \in \Gamma$. Without loss of generality, we assume that $\Gamma \neq \emptyset$. An adversary structure can be defined as $\mathcal{A} = 2^P \setminus \Gamma$. Thus for any set $A \subseteq P$, if $A \in \mathcal{A}$ and $A \supseteq A'$, then $A' \in \mathcal{A}$. It has been shown that LSSS's can be designed for any monotone access structures, so that any set of participants that is in Γ can learn a secret s but any set in $\mathcal{A}$ cannot. Next we show the construction and the properties of such an LSSS.

2.1 Constructing an LSSS

First, it is well-known that monotone span programs are essentially equivalent to LSSS's [13] (see also [5]).

Definition 1 [13] *A* monotone span program *is a triple* $(\mathbb{F}, M, \psi)$*, where* $\mathbb{F}$ *is a finite field,* M *is an* $h \times d$ *matrix* $(h \geq d)$*, and* $\psi : \{1, \ldots, h\} \rightarrow \{1, \ldots, n\}$ *is a surjective function that assigns a number of rows in* M *to each participant in* P*.*

For later use, we only allow each row of M to be assigned to a unique participant; i.e., if $\psi(i) = j$, then $\psi(i) \neq j'$ for any $j' \neq j$. This is easy to achieve by making duplicates of the rows that are assigned to multiple participants. Thus h can indicate the total number of shares distributed.

As Shamir's scheme, our construction assumes that $\mathbb{F}$ is sufficiently large. We also assume a message space $\mathbb{M} \subseteq \mathbb{F}$, from which the secret is drawn with respect to a certain probability distribution. Now with $(\mathbb{F}, M, \psi)$, one can share a secret using an LSSS.

Definition 2 *Given a monotone span program $(\mathbb{F}, M, \psi)$, a secret $s \in \mathbb{M}$ and a random vector $\mathbf{r} \in \mathbb{F}^{d-1}$. We regard $LS : (\mathbb{M}, \mathbb{F}^{d-1}) \to \mathbb{F}^h$ as a function such that (T denotes transpose)*

$$LS(s, \mathbf{r}) = M \times (s, \mathbf{r})^T = (s_1, \dots, s_h)^T,$$

where $s_1, \dots, s_h$ are the h shares generated by the LSSS, and they are assigned to the n participants by ψ. For any $1 \leq t \leq h$ shares $s_{i_1}, \dots, s_{i_t}$ $(1 \leq i_1 < \dots < i_t \leq h)$, let $\psi(i_1, \dots, i_t)$ be the set of participants to whom these shares are assigned and $s_0 \in \mathbb{M}$ be any possible secret, the LSSS must satisfy the following conditions:

Secrecy: $\Pr[s = s_0 | s_{i_1}, \dots, s_{i_t}] = \Pr[s = s_0]$ *if* $\psi(i_1, \dots, i_t) \in \mathcal{A}$*;*
Reconstruction: $\Pr[s = s_0 | s_{i_1}, \dots, s_{i_t}] = 0$ *or* 1 *if* $\psi(i_1, \dots, i_t) \in \Gamma$*.*

Apparently, if $\psi(i_1, \dots, i_t) \in \Gamma$, then in the linear span of the $i_1, \dots, i_t$-th rows of M, there must exist the *target vector* $tar = (1, 0, \dots, 0)$ [13]. This is to satisfy the reconstruction condition.

In the context of the information rate, the size of the secret shares has been studied in literature (e.g., [6, 23, 4]). However, to the best of our knowledge, there is no results regarding the tight upper bound on the total size of the shares, which is h in our LSSS. In fact, we do not know whether for any access structure, there exists an LSSS with size h polynomial in n. However we can have an exponential size LSSS, which we call *the worst case LSSS*, as follows. The worst case LSSS is defined by a monotone span program $(\mathbb{F}, M_{h \times d}, \psi)$ such that $d = |\mathcal{A}|$ and $h = O(dn)$. h is thus exponential in n because in general $|\mathcal{A}| = O(2^n)$. This construction somehow follows [10] (based on [21]).

The worst case LSSS

Given a set of n participants P and an adversary structure $\mathcal{A}$ on P. Let $\Delta = \{P \setminus A | A \in \mathcal{A}\}$ and $d = |\Delta| = |\mathcal{A}|$. Construct a $d \times d$ matrix M^V, which is an identity matrix except all entries in the first row are changed to 1.
Let $\Delta = \{D_1, ..., D_d\}$, then for each $1 \leq i \leq d$, construct a $|D_i| \times d$ matrix M_i such that each row of M_i is a duplication of the i-th row of M^V. Let $h = \sum_{i=1}^{d} |D_i|$, construct an $h \times d$ matrix M that is filled by $M_1, \dots, M_d$ from top to bottom.
The function ψ assigns the rows in M to each participant in such a manner that if a participant is in $D_i \in \Delta$ $(1 \leq i \leq d)$, then ψ assigns a row of M_i to this participant. **End.**

See the proof of the secrecy and reconstruction properties of the worst case LSSS in the full version of this paper [1].

2.2 Linear Codes

Given an LSSS defined by $(\mathbb{F}, M_{h \times d}, \psi)$. We denote k as the rank of M, thus $k \leq d$. In the rest of the paper, *we let the first k rows of M be linearly independent.* Thus $\psi(1, \dots, k) \in \Gamma$. Indeed, because otherwise $\psi(1, \dots, k) \in \mathcal{A}$ and the participants in $\psi(1,\ldots,k)$ can then recover all the other shares using linear combinations. This contradicts the secrecy condition of Definition 2.

Definition 3 *A linear code C is defined by a $k \times h$ generating matrix G in standard form $G = (I_k|A)$ [16], where I_k denotes the $k \times k$ identity matrix and A is a $k \times (h-k)$ matrix.*

The codewords of code C are determined by an encode *function $EC : \mathbb{F}^k \rightarrow \mathbb{F}^h$ such that given a k-vector $\mathbf{r} \in \mathbb{F}^k$,*

$$EC(\mathbf{r}) = \mathbf{r} \times G = \mathbf{c},$$

where $\mathbf{c}$ is an h-vector, as a codeword of C, and denoted $\mathbf{c} \in C$.

Evidently code C has $|\mathbb{F}|^k$ codewords.

We link an LSSS with a linear code as follows. In the rest of this section, *we let M_k be a $k \times d$ matrix that consists of the first k rows of M*, so the rank of M_k is k. We construct G in such a manner that the i-th column of G, which we call col_i, has the following property: $(col_i)^T \times M_k = row_i$, where row_i is the i-th row of M. This is possible because the rank of M is k, thus row_i is in the linear span of the first k rows of M (M_k). Therefore, the set $\{LS(s,\mathbf{r})|s \in \mathbb{M}, \mathbf{r} = \mathbb{F}^{d-1}\}$ is a subset of a linear code, because for any $s \in \mathbb{M}, \mathbf{r} \in \mathbb{F}^{d-1}$, we have

$$LS(s,\mathbf{r}) = (s_1,\ldots,s_h) = EC(s_1,\ldots,s_k) \in C.$$

Definition 4 *Let $\mathbf{k}$ be a k-vector such that $\mathbf{k} \times M_k = tar$, where $tar = (1,0,\ldots,0) \in \mathbb{F}^d$ is the target vector*[2]*. Let $\mathbf{r} \in \mathbb{F}^k$. We define a* decode *function $DC : \mathbb{F}^k \rightarrow \mathbb{F}$ such that $DC(\mathbf{r}) = \mathbf{r} \times \mathbf{k}^T$. We denote the output of the function, $r = DC(\mathbf{r})$, as* the information of the codeword $\mathbf{c} = EC(\mathbf{r})$.

Theorem 1 *Given any codeword $\mathbf{c} = (c_1,\ldots,c_h) = EC(\mathbf{r}) \in C$. One can decode the information of $\mathbf{c}$ with t entries $c_{i_1},\ldots,c_{i_t}$ $(1 \leq i_1 < \ldots < i_t \leq h)$ of $\mathbf{c}$ if and only if $\psi(i_1,\ldots,i_t) \in \mathit{\Gamma}$.*

Proof. Let $\mathbf{k}$ be a k vector such that the information of $\mathbf{c}$ is $r = DC(\mathbf{r}) = \mathbf{r} \times \mathbf{k}^T$. Remark that C is defined by G, which is derived from M of the LSSS. Let $\mathit{\Lambda}$ be a $k \times t$ matrix such that for each $1 \leq j \leq t$, the j-th column of $\mathit{\Lambda}$ is the i_j-th column of G, then we have

$$\begin{bmatrix} row_{i_1} \\ \vdots \\ row_{i_t} \end{bmatrix} = \mathit{\Lambda}^T \times M_k, \tag{1}$$

where for each $1 \leq j \leq t$, row_{i_j} is the i_j-th row of M.

First we show that if $\psi(i_1,\ldots,i_t) \in \mathcal{A}$, then one cannot decode r with $c_{i_1},\ldots,c_{i_t}$. Assume the opposite, i.e., r can be decoded with $c_{i_1},\ldots,c_{i_t}$. Since $r = \mathbf{r} \times \mathbf{k}^T$, the possibility that r can be decoded by $(c_{i_1},\ldots,c_{i_t})$ means that

[2] Because $\psi(1,\ldots,k) \in \mathit{\Gamma}$ as we showed before, $\mathbf{k}$ must exist.

the column vector $\mathbf{k}^T$ is in the linear span of the columns of Λ. That is, there exists a $\mathbf{t}^T$ such that $\mathbf{k}^T = \Lambda \times \mathbf{t}^T$ so that

$$r = \mathbf{r} \times \mathbf{k}^T = \mathbf{r} \times \Lambda \times \mathbf{t}^T = (c_{i_1}, \ldots, c_{i_t}) \times \mathbf{t}^T.$$

Since $\mathbf{k}^T = \Lambda \times \mathbf{t}^T \Rightarrow \mathbf{k} = \mathbf{t} \times \Lambda^T$, by multiplying $\mathbf{t}$ by both sides of Eq. 1, we have

$$\mathbf{t} \times \begin{bmatrix} row_{i_1} \\ \vdots \\ row_{i_t} \end{bmatrix} = \mathbf{t} \times \Lambda^T \times M_k = \mathbf{k} \times M_k = tar.$$

This means that the target vector tar is in the linear span of the rows assigned to the participants $\psi(i_1, \ldots, i_t) \in \mathcal{A}$, which is not allowed in our LSSS due to the secrecy condition.

Next if $\psi(i_1, \ldots, i_t) \in \Gamma$, then by the reverse of the above proof and the reconstruction condition of the LSSS, we can easily prove that one can decode r with $c_{i_1}, \ldots, c_{i_t}$. □

Given that $\mathbf{c} = (c_1, \ldots, c_h)$ is a codeword at the encoding end, and $\mathbf{x} = (x_1, \ldots, x_h)$ is the input at the decoding end, because of the channel noise, it is possible that $\mathbf{x} \neq \mathbf{c}$. We let $\mathbf{e} = (e_1, \ldots, e_h)$ be an *error vector* such that $\mathbf{e} = \mathbf{x} - \mathbf{c}$. Normally we have the following assumption: let $E = \{i | e_i \neq 0\}$ be an *error locator*, we always have $\psi(E) \in \mathcal{A}$. That is, the errors in a codeword are caused by a set in the adversary structure. With this assumption, it is well-known that

- the decoder can detect that $\mathbf{x}$ is not a codeword if and only if $P \notin 2\mathcal{A}$ (i.e., $P \notin \{A_1 \cup A_2 | A_1, A_2 \in \mathcal{A}\}$), where P is the set of all participants, and
- the decoder can decode the information of $\mathbf{c}$ from $\mathbf{x}$ if and only if $P \notin 3\mathcal{A}$ (i.e., $P \notin \{A_1 \cup A_2 \cup A_3 | A_1, A_2, A_3 \in \mathcal{A}\}$).

See a proof of this result in the full version of this paper [1].

2.3 Pseudo-basis and Pseudo-dimension

In Eurocrypt '08, Kurosawa and Suzuki initiated the idea of pseudo-basis and pseudo-dimension in the threshold model with multiple codewords [15]. A generalization of the pseudo-basis and pseudo-dimension is possible if $P \notin 2\mathcal{A}$ (corresponding to $n \geq 2t + 1$ in the threshold model), thus *we assume that $P \notin 2\mathcal{A}$ in this section.* Next, we let $\psi^{-1} : \{1, \ldots, n\} \to \{1, \ldots, h\}$ be the inverse function of ψ. That is, let $A \subseteq P$, then $\psi^{-1}(A)$ *returns all the locations in a codeword that are assigned to the participants in A by ψ.*

Definition 5 *Let $A \subseteq P$, we define $|A|$ as the* size *of A and $|\psi^{-1}(A)|$ as the* weight *of A. We denote*

$$sz^{\mathcal{A}} = \max\{\text{size of } A | A \in \mathcal{A}\} \textit{ and } wt^{\mathcal{A}} = \max\{\text{weight of } A | A \in \mathcal{A}\}.$$

Evidently $sz^{\mathcal{A}} = O(n)$ and $wt^{\mathcal{A}} = O(h)$. The idea of the generalization is as follows. The encoder sends m codewords $\mathbf{c}_1, \ldots, \mathbf{c}_m$, and the decoder receives m h-vectors $\mathbf{x}_1, \ldots, \mathbf{x}_m$ such that for each $1 \leq i \leq m$, $\mathbf{x}_i = \mathbf{c}_i + \mathbf{e}_i$ where $\mathbf{e}_i = (e_{i1}, \ldots, e_{ih})$ is an error vector. For each $\mathbf{e}_i$, let $E_i = \{j | e_{ij} \neq 0\}$ be an error locator, then E_i has the following two properties: (1) $|E_i| \leq wt^{\mathcal{A}}$ and (2) $\psi(E_i) \in \mathcal{A}$ and hence $|\psi(E_i)| \leq sz^{\mathcal{A}}$. We assume that $\bigcup_{i=1}^{m} E_i \in \mathcal{A}$, i.e., the errors in all the codewords are caused by the same set in $\mathcal{A}$. Now we give our pseudo-basis construction scheme as follows.

Pseudo-basis construction scheme

Set $B := \emptyset$. For each $1 \leq i \leq m$, distinguish the following two cases:

1. $B = \emptyset$: if $x_i \in C$, then do nothing, otherwise, then add $\mathbf{x}_i$ in B.
2. Otherwise: let $B = \{\mathbf{x}_{g_1}, \ldots, \mathbf{x}_{g_b}\}$ where $1 \leq g_1 < \ldots < g_b < i$, if there exist $(a_1, \ldots, a_b) \in \mathbb{F}^b$ such that $\mathbf{x}_i + a_1\mathbf{x}_{g_1} + \ldots + a_b\mathbf{x}_{g_b} \in C$, then do nothing, otherwise, add $\mathbf{x}_i$ in B.

Let B be the pseudo-basis. Thus $|B|$ is the pseudo-dimension. **End.**

It is trivial that the pseudo-dimension of our scheme is at most $wt^{\mathcal{A}} = O(h)$, because there are at most $wt^{\mathcal{A}}$ non-zero entries in each error vector. Thus the pseudo-basis has $O(h^2)$ field elements.

Lemma 1 *For any codeword $\mathbf{c} = (c_1, \ldots, c_h) \in C$, let $D = \{i | c_i \neq 0\}$. If $P \notin 2\mathcal{A}$ and $\psi(D) \in \mathcal{A}$, then the information of $\mathbf{c}$ is 0.*

Proof. Let $O = \{i | c_i = 0\}$. From $P \notin 2\mathcal{A}$ and $\psi(D) \in \mathcal{A}$, we can have $\psi(O) \in \Gamma$. According to Theorem 1, the information of $\mathbf{c}$ can be decoded with all the entries c_i such that $i \in O$. Since all these entries are 0's, the information of $\mathbf{c}$ is 0. □

Given a codeword $\mathbf{c} \in C$ and a vector $\mathbf{x}$, and let $\mathbf{e} = \mathbf{x} - \mathbf{c}$ be an error vector such that $\psi(E) \in \mathcal{A}$. If $\mathbf{e} \in C$, then $\mathbf{x} \in C$. Due to Lemma 1, the information of $\mathbf{e}$ is 0, so the information of $\mathbf{x}$ equals to the information of $\mathbf{c}$. That is, the error vector $\mathbf{e}$ does not actually cause errors, and we call this kind of error vector *invalid*. Evidently, the vector $\mathbf{0} \in \mathbb{F}^h$ is an invalid error vector.

Let $B = \{\mathbf{x}_{g_1}, \ldots, \mathbf{x}_{g_b}\}$ be a pseudo-basis, where $1 \leq g_1 < \ldots < g_b \leq m$, and $E_{g_1}, \ldots, E_{g_b}$ be the respective error locators. we denote $F = \bigcup_{i=1}^{b} E_{g_i}$ as the *final error locator* of B.

Theorem 2 *If the final error locator of a pseudo-basis is known, then the decoder can decode the information of all the codewords.*

Proof. Given the final error locator F of a pseudo-basis $B = \{\mathbf{x}_{g_1}, \ldots, \mathbf{x}_{g_b}\}$, a decoding scheme is as simple as the following:

Decoding scheme from the pseudo-basis

For each $1 \leq i \leq m$, decode the information r_i of $\mathbf{c}_i$ from $\mathbf{x}_i$ such that if $j \in F$, then the j-th entry of $\mathbf{x}_i$ is not used for decoding. **End.**

It is straightforward that if $i \in \{g_1, \ldots, g_b\}$, then the decoded information r_i is correct. Indeed, $P \notin 2\mathcal{A}$ and $\psi(F) \in \mathcal{A}$ imply that $\psi(\{1, \ldots, h\} \setminus F) \in \Gamma$. Thus according to Theorem 1, the entries not indicated by F can be used to decode r_i. Since F contains all the error locations of $\mathbf{x}_i$, all the entries that are used to decode r_i are correct.

Next, if $i \in \{1, \ldots, m\} \setminus \{g_1, \ldots, g_b\}$, then because of the existence of non-zero invalid error vectors, it is possible that $E_i \supsetneq F$. That is, errors may exist in the entries used to decode r_i. Since $\mathbf{x}_i \notin B$, there exist $(a_1, \ldots, a_b) \in \mathbb{F}^b$ such that $\mathbf{x}_i + a_1\mathbf{x}_{g_1} + \ldots + a_b\mathbf{x}_{g_b} \in C$. Thus $\mathbf{e}_i + a_1\mathbf{e}_{g_1} + \ldots + a_b\mathbf{e}_{g_b} \in C$. Let $\mathbf{e}'_i = \mathbf{e}_i + a_1\mathbf{e}_{g_1} + \ldots + a_b\mathbf{e}_{g_b}$, we have that $\mathbf{e}'_i$ is an invalid error vector. Thus one can decode the information r_i of $\mathbf{c}_i$ correctly from the vector $\mathbf{x}'_i = \mathbf{c}_i + \mathbf{e}'_i$. Since $\mathbf{x}_i = \mathbf{c}_i + \mathbf{e}_i$, it is clear that excluding the entries indicated by F, the remaining entries of $\mathbf{x}_i$ are the same as those of $\mathbf{x}'_i$. That is, even though errors may exist in the remaining entries, one can decode the information r_i of $\mathbf{c}_i$ correctly from these entries. □

3 PSMT Preliminaries

We abstract away the concrete network structure and model a network by a graph $G(V, E)$, whose nodes are the parties in the network and edges are point-to-point secure communication channels. We consider two kinds of network graphs in this paper:

1. *Undirected graphs* - in which all the edges are undirected, and allow two-way communication;
2. *Directed graphs* - in which all the edges are one-way directed or bi-directed, and allow mixed communication.

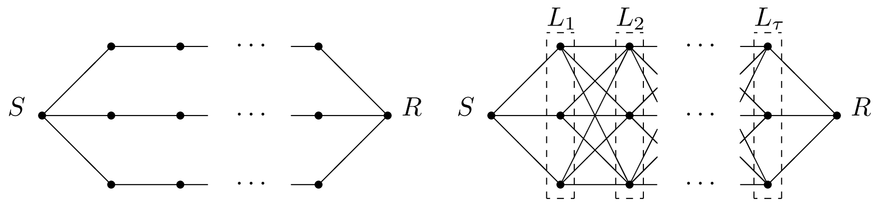
Given an adversary structure $\mathcal{A}$ on the nodes of a graph, we say the sender S and the receiver R are *$d\mathcal{A}$-separated* if there exist d sets $A_1, \ldots, A_d \in \mathcal{A}$ such that all paths between S and R pass through some nodes in $\bigcup_{i=1}^{d} A_i$; otherwise we say they are *$d\mathcal{A}$-connected*.

In the context of PSMT, *perfect security* requires the achievement of *perfect privacy* (i.e., zero probability that the adversary learns the message from the information he gets) and *perfect reliability* (i.e., zero probability that R fails to recover the message correctly). The necessary and sufficient conditions (N&S) for PSMT on different network graphs have been given in previous results:

N&S-undirected: in undirected graphs, S and R are $2\mathcal{A}$-connected [14];

N&S-directed-1: in directed graphs without feedback paths, S and R are $3\mathcal{A}$-connected [8];

N&S-directed-2: in directed graphs with feedback paths, S and R are $2\mathcal{A}$-connected with the forward paths from S to R, and if S and R are $3\mathcal{A}$-separated, then for any three sets $A_1, A_2, A_3 \in \mathcal{A}$ such that $A_1 \cup A_2 \cup A_3$ separates S and R, at most one of these three sets separates S and R on the feedback paths from R to S [19, 24].



(a) G_1: $|W^*| = d + 1 = 3$. **(b)** G_2: $|W^*|$ is exponential in n.

Fig. 1. $2\mathcal{A}$-connectivity in different graphs.

It can be seen that the paths between S and R play an important role in the study of PSMT. Next we show how a characterization of the *critical paths* determines the PSTM protocols and their transmission complexity (TC).

3.1 Critical Paths

Unlike those in the threshold model, the N&S conditions for PSMT in the general adversary model do not require node-disjoint paths. This rises the question of how to transmit messages in a general network graph. The straightforward solution (though somehow less efficient) is to characterize the graph into all possible paths between S and R. To this end, the idea of *critical paths* was introduced by Kumar et al. [14] in their initial study. We extend their study, by firstly giving a formal definition as follows.

Definition 6 *Given a graph $G(V, E)$, in which S and R are $d\mathcal{A}$-connected. A set of paths W is called* critical, *if S and R are $d\mathcal{A}$-connected with all paths in W, but are $d\mathcal{A}$-separated with all paths in any $W' \subsetneq W$. Let $\mathcal{W}$ be the set of all critical sets of paths, we define a* minimal critical set W^* *such that $W^* \in \mathcal{W}$ and $|W^*| = \min\{|W| : W \in \mathcal{W}\}$.*

Without loss of generality, we assume that there does not exist a trusted path between S and R; i.e., $|W^*| > 1$.

Observation 1 *With any graph in which S and R are $d\mathcal{A}$-connected, $|W^*|$ can be as small as $d + 1$ or as large as exponential in the size of the graph.*

We give two examples in Fig. 1. In the examples we assume that S and R are $2\mathcal{A}$-connected. First suppose a graph G_1 is as shown in Fig. 1(a), in which there are only 3 paths between S and R. The adversary structure $\mathcal{A}$ has the following property: all nodes in any set $A \in \mathcal{A}$ are on the same path. Thus it is clear that in G_1, S and R are $2\mathcal{A}$-connected, and all the 3 paths are in W^*.

Next suppose a graph G_2 is as shown in Fig. 1(b). We assume that except S and R, there are 3τ nodes in G_2. We can view S and R as they are connected by τ levels $L_1, \ldots, L_\tau$, where each level L_i ($1 \leq i \leq \tau$) is a set of 3 nodes, and there is an edge between each node in L_i and each node in L_{i+1}. The adversary structure $\mathcal{A}$ has the following property: for each set $A \in 2^P$, if there exist two nodes $v_1, v_2 \in A$ such that $v_1, v_2 \in L_i$ $(1 \leq i \leq \tau)$, then $A \notin \mathcal{A}$; otherwise $A \in \mathcal{A}$. In other words, the adversary E can control at most 1 node of each level.

Obviously S and R are $2\mathcal{A}$-connected in G_2, but if we remove any edge from the graph, then they are $2\mathcal{A}$-separated. Also straightforwardly $|W^*| = 3^\tau$, because the critical paths are all the paths with exactly one node of each level on them. Thus we have that $|W^*|$ is exponential in the size of the network, which is $9\tau - 3$.

Of course our examples can easily be adapted to other connectivity, e.g., $3\mathcal{A}$-connectivity.

Therefore, if a PSMT protocol is executed via the paths in the graphs, then it is impossible to determine its TC in the size of the network, because the number of paths varies remarkably in different graphs with the same connectivity (e.g., G_1 and G_2). Thus we determine TC in the number of critical paths. For this purpose, a re-characterization of the adversary structure is needed.

In general, the participants in an adversary structure are considered to be the nodes in the network graph. We denote this adversary structure as $\mathcal{A}^V$. Given a critical set of paths W, we define a new adversary structure $\mathcal{A}^W$ such that $|\mathcal{A}^W| = |\mathcal{A}^V|$, and for each set $A^V \in \mathcal{A}^V$, there is a corresponding set $A^W \in \mathcal{A}^W$ such that A^W consists of all the paths in W that pass through nodes in A^V.

It is clear that if S and R are $d\mathcal{A}^V$-connected, then they are $d\mathcal{A}^W$-connected with W. In the rest of the paper, we use $\mathcal{A}^W$ as the considered adversary structure. Thus we let $\mathcal{A} = \mathcal{A}^W$ and *the participants of the adversary structure are the critical paths of the network graph.*

3.2 Improvements to the Previous Results

In the rest of the paper, we let $n = |W|$ be the number of critical paths, and $\mathcal{A}$ be an adversary structure over the n paths.

Because the previous protocols use different characterizations for PSMT, it is not straightforward to compare their TC with our result. In fact, we need to compare the three parameters $(n, |\mathcal{A}|, h)$ [3] that determine the TC of the protocols. First we do not know the tight upper bound on h, but our worst case LSSS achieves $h \leq O(|\mathcal{A}|n)$, so h should not be larger. In general $|\mathcal{A}|$ is exponential in n, but due to the way that the critical paths are selected, n can be polynomial in $|\mathcal{A}|$ in some network graph [14]. Either way, our results significantly improve the previous results in terms of round complexity (RC) and transmission complexity (TC) over a single message. We also present some efficient protocols to transmit $\ell > 1$ messages. The problem of multiple message transmission has not been studied before in the general adversary model.

A summary of the results are shown in Table 1 in Section 1. Note that Desmedt et al.'s protocol [8] is executed in directed graphs without feedback,

[3] As shown in the previous section, h is the size of the LSSS as well as the length of the codewords.

which means that the receiver R cannot send messages to the sender S. Thus the protocols in this graph must be non-interactive and can only have 1-round. Their protocol is actually an alternative use of the worst case LSSS that we showed before. Thus the protocol can easily be reformed into a 1-round protocol with TC $O(h)$. The protocol by Yang and Desmedt [24] uses the settings in [19], which require both the RC and TC to be exponential in $|\mathcal{A}|$. As we discussed before, both h and n are at most polynomial in $|\mathcal{A}|$, so our improvements are obvious. We also remark that in the studies of the general adversary model, our results are the first to have constant RC in undirected and directed-2 graphs.

3.3 Other Preliminaries

We assume that each message s is drawn from the message space $\mathbb{M} \subseteq \mathbb{F}$ with respect to a certain probability distribution. Since two different type of graphs are considered, we have the following: in an undirected graph, we denote $W = \{w_1, \ldots, w_n\}$ as a critical set of undirected paths; in a directed graph, we denote $W = \{w_1, \ldots, w_n\}$ as a critical set of the forward paths and $Q = \{q_1, \ldots, q_u\}$ as a critical set of the feedback paths, where $u = O(n)$.

Given that S and R are $2\mathcal{A}$-connected with W, if S sends the same message via all paths in W, then R is able to receive the message perfectly reliably [14]. In our protocols we say "*S broadcasts a message via W*" to indicate this kind of transmission. Thus the TC of the broadcast of 1 field element is $O(n)$.

Note that the linear code is constructed considering the critical paths as the participants. When S sends a codeword $\mathbf{c} = (c_1, \ldots, c_h)$ in such a manner that for each $1 \leq j \leq h$, if $\psi(j) = w_i$ for some $1 \leq i \leq n$, then S sends c_j via w_i, we say "*S sends $\mathbf{c}$ via W with respect to ψ*" to indicate this kind of transmission. Thus the TC of the transmission of 1 codeword is $O(h)$.

In our protocols, we omit some indices for the communication. For example, if S sends a pseudo-basis to R, then generally S should attach a index in the transmission to indicate exactly which codeword each vector in the pseudo-basis corresponds to. Indexing is very cheap in terms of TC. Thus in our protocols, we omit some indices to make the protocols easier to read.

4 PSMT in Undirected Graphs

In this section we show our PSMT protocols in undirected graphs. According to N&S-undirected, S and R must be $2\mathcal{A}$-connected in an undirected graph. We first give 3-round protocols in Section 4.1 for the transmissions of a single message and multiple messages, and then give 2-round protocols in Section 4.2. The protocols given in this section are along the lines of the results in [15].

4.1 3-Round Undirected Protocols

We omit the 3-round protocols in this section due to lack of space, and also because they are relatively simple. However, the TC of our 3-round protocol over a single message is $O(hn^2)$, and the TC of our 3-round protocol over multiple (ℓ) messages is $O(h\ell)$ where $\ell = wt^{\mathcal{A}}h$. Thus the TC of both protocols are about optimal in the context of PSMT in the general adversary model. For the details of the 3-round undirected protocols, see the full version of this paper [1].

4.2 2-Round Undirected Protocols

First we give a 2-round protocol to transmit a single message.

2-round undirected protocol for a single message s

Round 1 - R to S:

1. R chooses n random k-vectors $\mathbf{r}_1, \dots, \mathbf{r}_n \in \mathbb{F}^k$, and for each $1 \le i \le n$, R encodes $\mathbf{r}_i$ to get codeword $\mathbf{c}_i = EC(\mathbf{r}_i) = (c_{i1}, \dots, c_{ih})$.
2. For each $1 \le i \le n$, R sends vector $\mathbf{r}_i$ via path w_i, and sends codeword $\mathbf{c}_i$ via W with respect to ψ.

Round 2 - S to R:

1. S receives n k-vectors $\mathbf{r}'_1, \dots, \mathbf{r}'_n$ and n h-vectors $\mathbf{x}_1, \dots, \mathbf{x}_n$ from W. For each $1 \le i \le n$, let $\mathbf{x}_i = (x_{i1}, \dots, x_{ih})$.
2. For each $1 \le i \le n$, S encodes $\mathbf{r}'_i$ to get codeword $\mathbf{c}'_i = EC(\mathbf{r}'_i) = (c'_{i1}, \dots, c'_{ih})$. S then constructs a set D_i such that for each $1 \le j \le h$, iff $x_{ij} \ne c'_{ij}$, then $(x_{ij}, j) \in D_i$.
3. S finds a k-vector $\mathbf{r}^S$ such that $s = DC(\mathbf{r}^S)$, and then encodes $\mathbf{c}^S = EC(\mathbf{r}^S) = (c^S_1, \dots, c^S_h)$. For each $1 \le j \le h$, if $\psi(j) = w_i$, then S computes $z_j = c^S_j + c'_{ij}$. Finally S sets $\mathbf{z} = (z_1, \dots, z_h)$.
4. S broadcasts $\mathbf{z}$ and $D_1, \dots, D_n$ via W.

Recovery Phase

1. R receives $\mathbf{z}$ and $D_1, \dots, D_n$ from W.
2. R sets $F := \emptyset$. For each $1 \le i \le n$, if there exists a pair $(x_{ij}, j) \in D_i$ such that $x_{ij} = c_{ij}$, then R sets $F := F \cup \{i\}$.
3. For each $1 \le j \le h$, if $\psi(j) = w_i$, then R computes $c^R_j = z_j - c_{ij}$. R then decodes s' as the information of $(c^R_1, \dots, c^R_h)$ such that for any $\psi(j) = w_i$ where $i \in F$, the entry c^R_j is not used for decoding. **End.**

Proof of perfect security. Omitted. See the full version of this paper [1].

TC of the protocol. Let $TC(i)$ be the TC of Round i for $1 \le i \le 3$. In this protocol:

$$TC(1) = hn + kn = O(hn)$$
$$TC(2) = O(n(h + 2hn)) = O(hn^2)$$

We have that the total TC is $O(hn^2)$ field elements.

Next, before we show our 2-round PSMT protocol that transmits multiple messages, we employ a well-known technique in this context: the ***randomness extractor*** [22, 2, 15]. Suppose that the adversary has no knowledge on ℓ out of m random elements $r_1, \dots, r_m \in \mathbb{F}$. Let $f(x)$ be a polynomial of degree $deg\ f(x) \le m - 1$ such that $f(i) = r_i$ for each $1 \le i \le m$, then the adversary has no knowledge on $z_j = f(m+j)$ for each $1 \leq j \leq \ell$. We denote a function $RE : \mathbb{F}^m \rightarrow \mathbb{F}^\ell$ as a randomness extractor such that $RE(r_1, \ldots, r_m) = (z_1, \ldots, z_\ell)$. This function will be used in the following 2-round PSMT protocol.

2-round undirected protocol for $\ell = wt^{\mathcal{A}}(n - sz^{\mathcal{A}} - 1)$ messages $s_1, \ldots, s_\ell$

Round 1 - R to S:

1. R chooses $wt^{\mathcal{A}}n$ random k-vectors $\mathbf{r}_1, \ldots, \mathbf{r}_{wt^{\mathcal{A}}n} \in \mathbb{F}^k$, and for each $1 \leq i \leq wt^{\mathcal{A}}n$, S encodes $\mathbf{r}_i$ to get codeword $\mathbf{c}_i = EC(\mathbf{r}_i) = (c_{i1}, \ldots, c_{ih})$.
2. For each $1 \leq i \leq n$, R sends vectors $\mathbf{r}_{i+0 \cdot wt^{\mathcal{A}}}, \mathbf{r}_{i+1 \cdot wt^{\mathcal{A}}}, \ldots, \mathbf{r}_{i+(wt^{\mathcal{A}}-1)wt^{\mathcal{A}}}$ via path w_i. R also sends codewords $\mathbf{c}_1, \ldots, \mathbf{c}_{wt^{\mathcal{A}}n}$ via W with respect to ψ.

Round 2 - S to R:

1. S receives $wt^{\mathcal{A}}$ k-vectors $\mathbf{r}'_{i+0 \cdot wt^{\mathcal{A}}}, \mathbf{r}'_{i+1 \cdot wt^{\mathcal{A}}}, \ldots, \mathbf{r}'_{i+(wt^{\mathcal{A}}-1)wt^{\mathcal{A}}}$ on each path w_i ($1 \leq i \leq n$), and also receives $wt^{\mathcal{A}}n$ h-vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{wt^{\mathcal{A}}n}$ from W. For each $1 \leq i \leq wt^{\mathcal{A}}n$, let $\mathbf{x}_i = (x_{i1}, \ldots, x_{ih})$.
2. For each $1 \leq i \leq wt^{\mathcal{A}}n$, S uses the pseudo-basis construction scheme to construct a pseudo-basis B from $\mathbf{x}_1, \ldots, \mathbf{x}_{wt^{\mathcal{A}}n}$. Let b be the pseudo-dimension of B, then $b \leq wt^{\mathcal{A}}$.
3. For each $1 \leq i \leq wt^{\mathcal{A}}n$, S encodes $\mathbf{r}'_i$ to get codeword $\mathbf{c}'_i = EC(\mathbf{r}'_i) = (c'_{i1}, \ldots, c'_{ih})$. S then constructs a set D_i such that for each $1 \leq j \leq h$, iff $x_{ij} \neq c'_{ij}$, then $(c'_{ij}, j) \in D_i$.
4. For each $1 \leq i \leq wt^{\mathcal{A}}n$, S decodes $r'_i = DC(\mathbf{r}'_i)$. S then constructs a set T such that iff $|D_i| \leq wt^{\mathcal{A}}$, then $r'_i \in T$. S uses the randomness extractor to get $(z_1, \ldots, z_\ell) = RE(T)$, and for each $1 \leq i \leq \ell$, S computes $\sigma_i = s_i + z_i$.
5. S broadcasts the pseudo-basis B and $\sigma_1, \ldots, \sigma_\ell$. For each $1 \leq i \leq wt^{\mathcal{A}}n$, if $|D_i| > wt^{\mathcal{A}}$, then S broadcasts "ignore i"; else, then S broadcasts D_i.

Recovery Phase

1. R finds the final error locator F from B.
2. For each D_i that R receives on W, R constructs an h-vector $\mathbf{c}''_i = (c''_{i1}, \ldots, c''_{ih})$ such that for each $1 \leq j \leq h$, if $(c'_{ij}, j) \in D_i$, then $c''_{ij} = c'_{ij}$; else, then $c''_{ij} = c_{ij}$. R then decodes the information r''_i of $\mathbf{c}''_i$ such that for any $j \in F$, c''_{ij} is not used for decoding. R puts r''_i in a set T'.
3. R uses the randomness extractor to get $(z'_1, \ldots, z'_\ell) = RE(T')$, and for each $1 \leq i \leq \ell$, R computes $s'_i = \sigma_i - z'_i$. **End.**

Proof of perfect security. Omitted. See the full version of this paper [1].

TC of the protocol. In this protocol:

$$TC(1) = (k+h)wt^{\mathcal{A}}n = O(h\ell)$$
$$TC(2) = O(n(wt^{\mathcal{A}}h + \ell + wt^{\mathcal{A}}n \cdot 2h)) = O(h^2n^2) = O(hn\ell)$$

We have that the total TC is $O(hn\ell)$ field elements.

5 PSMT in Directed Graphs

In this section we show our PSMT protocols in directed graphs. We let $W = \{w_1, \ldots, w_n\}$ be the critical set of forward paths and $Q = \{q_1, \ldots, q_u\}$ be the critical set of feedback paths.

In a directed graph without feedback ($Q = \emptyset$), S only needs to send a codeword $\mathbf{c}$, of which the information is the message s, to R via W with respect to ψ. Due to N&S-directed-1, S and R are $3\mathcal{A}$-connected, R can decode the information of $\mathbf{c}$ by correcting errors. Thus the protocol is perfectly secure and the TC is $O(h)$. We remark that Desmedt et al.'s protocol [8] is actually an alternative use of the worst case LSSS.

Next we consider a directed graph with feedback ($Q \neq \emptyset$). We give our 3-round protocols under the condition of N&S-directed-2 in Section 5.1. In Section 5.2, we show that N&S-directed-2 is not sufficient for 2-round PSMT protocols, and hence we give a new N&S condition and propose our protocols under this condition. The protocols given in this section are along the lines of the results in [18, 17].

5.1 3-Round Directed Protocols

Before we show our 3-round protocols, we notice that the adversary structure $\mathcal{A}$ is over all paths in $W \cup Q$. However, in our 3-round protocols, we do not need to assign shares (or entries) to the paths in Q. Thus we denote an adversary structure $\mathcal{A}'$ over the paths in W only, i.e., for any set $A \in \mathcal{A}$, there is a corresponding set $A' \in \mathcal{A}'$ such that $A' = A \cap W$. Thus S and R are $2\mathcal{A}'$-connected with the paths in W. Note that in this section, the linear codes in our protocols are constructed with respect to $\mathcal{A}'$.

3-round directed protocol for a single message s

Round 1 - S to R:

1. S chooses $wt^{\mathcal{A}}(u+1)+1$ random k-vectors $\mathbf{r}_1, \ldots, \mathbf{r}_{wt^{\mathcal{A}}(u+1)+1} \in \mathbb{F}^k$, and for each $1 \leq i \leq wt^{\mathcal{A}}(u+1)+1$, S encodes $\mathbf{r}_i$ to get codeword $\mathbf{c}_i = EC(\mathbf{r}_i) = (c_{i1}, \ldots, c_{ih})$.
2. For each $1 \leq i \leq wt^{\mathcal{A}}(u+1)+1$, S sends $\mathbf{c}_i$ via W with respect to ψ.

Round 2 - R to S:

1. R receives $wt^{\mathcal{A}}(u+1)+1$ h-vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{wt^{\mathcal{A}}(u+1)+1}$ from W. R uses the pseudo-basis construction scheme (see Section 2.3) to construct a pseudo-basis B from $\mathbf{x}_1, \ldots, \mathbf{x}_{wt^{\mathcal{A}}(u+1)+1}$, and then broadcasts B via all paths $q_1, \ldots, q_u \in Q$.

Round 3 - S to R:

1. For each $1 \leq v \leq u$, let B_v be the pseudo-basis that S receives on path q_v, and let b_v be the pseudo-dimension of B_v.
2. For each $1 \leq v \leq u$, if $b_v > wt^{\mathcal{A}}$, then S broadcasts "ignore v" via W; else then S finds the final error locator F_v from B_v. If $|F_v| > wt^{\mathcal{A}}$, then S broadcasts "ignore v" via W; else then S broadcasts B_v and F_v via W.

3. S sets $U := \emptyset$ and $T := \emptyset$. For each $1 \le v \le u$ such that $b_v \le wt^{\mathcal{A}}$, S adds all the actual codewords ($\mathbf{c}_i$'s) that correspond to the h-vectors in B_v to U. Thus at last, $|U| \le wt^{\mathcal{A}}u$. For each $\mathbf{r}_i$ such that $EC(\mathbf{r}_i) = \mathbf{c}_i \notin U$, if $i \notin T$ and $|T| < wt^{\mathcal{A}} + 1$, then S sets $T := T \cup \{i\}$. Thus at last, $|T| = wt^{\mathcal{A}} + 1$. For each $i \in T$, S decodes $r_i = DC(\mathbf{r}_i)$. S computes $\sigma = s + \sum_{i \in T} r_i$, and broadcasts σ and T via W.

Recovery Phase

Let $v := 1$, while $v \le u$:

1. if R receives "ignore v" from W, then R sets $v := v + 1$;
2. else if R receives B_v and F_v from W, then
   (a) if $B_v \neq B$, then R sets $v := v + 1$;
   (b) else, then with F_v, σ and T, R uses the decoding scheme from pseudo-basis (see Section 2.3) to get the information r_i of $\mathbf{c}_i$ for each $i \in T$. R then recovers $s = \sigma - \sum_{i \in T} r_i$, and terminates the protocol.

If $v > u$, then R knows that S did not receive the correct pseudo-basis B, so all paths $q_1, \ldots, q_u \in Q$ are corrupted. For each $i \in T$, R finds a set $A \in \mathcal{A}$ such that $Q \subseteq A$, and if A's entries in $\mathbf{x}_i$ are removed, all the remaining entries are a part of a codeword $\mathbf{c}'_i \in C$, then R decodes r'_i as the information of $\mathbf{c}'_i$. R recovers $s' = \sigma - \sum_{i \in T} r'_i$. **End.**

Proof of perfect security. Omitted. See the full version of this paper [1].

TC of the protocol. In this protocol:

$$\begin{aligned}
&TC(1) = h(wt^{\mathcal{A}}(u+1)+1) = O(h^2 n)\\
&TC(2) = O(u(wt^{\mathcal{A}}h)) = O(h^2 n)\\
&TC(3) = O(n(wt^{\mathcal{A}}hu + wt^{\mathcal{A}}u + 1 + wt^{\mathcal{A}} + 1)) = O(h^2 n^2)
\end{aligned}$$

We have that the total TC is $O(h^2 n^2)$ field elements.

Our 3-round protocol that transmits multiple messages is a generalization of the above protocol for a single message transmission. Thus we only show their differences as follows.

3-round directed protocol for $\ell = wt^{\mathcal{A}}u$ message $s_1, \ldots, s_\ell$

Round 1 - S to R: S does the same only for $wt^{\mathcal{A}}(u+1) + \ell$ random k-vectors.
Round 2 - R to S: R does the same.
Round 3 - S to R: S does the same until step 3.

3. S sets $U := \emptyset$. For each $1 \le v \le u$ such that $b_v \le wt^{\mathcal{A}}$, S adds all the actual codewords ($\mathbf{c}_i$'s) that correspond to the h-vectors in B_v to U. Thus at last, $|U| \le wt^{\mathcal{A}}u$.
4. S sets $T_1, \ldots, T_\ell := \emptyset$. For each $\mathbf{r}_i$ such that $EC(\mathbf{r}_i) = \mathbf{c}_i \notin U$, for each $1 \le j \le \ell$, if $i \notin T_j$ and $|T_j| < wt^{\mathcal{A}}$, then S sets $T_j := T_j \cup \{i\}$. Thus at last, all $T_1, \ldots, T_\ell$ are *the same* and $|T_j| = wt^{\mathcal{A}}$. There are at least ℓ vectors $\mathbf{r}_i$ such that $EC(\mathbf{r}_i) = \mathbf{c}_i \notin U$ and $i \notin T_j$ [4]. Let $\mathbf{r}_{i_1}, \ldots, \mathbf{r}_{i_\ell}$ be

[4] This is because $|U| \le wt^{\mathcal{A}}u$, $|T_j| = wt^{\mathcal{A}}$ and the total number of vectors $\mathbf{r}_i$ is $wt^{\mathcal{A}}(u+1) + \ell$.

ℓ such vectors, then for each $1 \le j \le \ell$, S sets $T_j := T_j \cup \{i_j\}$. Thus $|T_j| = wt^{\mathcal{A}} + 1$, and all $T_1, \ldots, T_\ell$ are *different.* For each $1 \le j \le \ell$ and $i \in T_j$, S decodes $r_i = DC(\mathbf{r}_i)$, computes $\sigma_j = s_j + \sum_{i \in T_j} r_i$, and broadcasts σ_j and T_j via W.

Recovery Phase For each $1 \le j \le \ell$, R does the same to recover s_j. **End.**

Proof of perfect security. Omitted. See the full version of this paper [1].

TC of the protocol. In this protocol:

$$TC(1) = h(wt^{\mathcal{A}}(u+1) + wt^{\mathcal{A}}u) = O(h\ell)$$
$$TC(2) = O(u(wt^{\mathcal{A}}h)) = O(h\ell)$$
$$TC(3) = O(n(wt^{\mathcal{A}}hu + wt^{\mathcal{A}}u + wt^{\mathcal{A}}u(1 + wt^{\mathcal{A}} + 1))) = O(hn\ell)$$

We have that the total TC is $O(hn\ell)$ field elements.

5.2 2-Round Directed Protocols

In [18], Patra et al. showed that in the threshold model, the minimal connectivity for PSMT in directed graph is not sufficient for a 2-round protocol. Here we do the similar. That is, we prove that in the general adversary model, N&S-directed-2 is not sufficient for a 2-round protocol. Note that the general assumption is that the feedback channels are not reliable (i.e., not $2\mathcal{A}$-connected).

Theorem 3 *Given a directed graph $G(V, E)$ and an adversary structure $\mathcal{A}$, 2-round PSMT is possible if and only if S and R are $2\mathcal{A}$-connected with the forward paths and $3\mathcal{A}$-connected in G.*

Proof. First we prove the necessity of the condition. $2\mathcal{A}$-connectivity with the forward paths is obviously necessary. Now assume that S and R are $3\mathcal{A}$-separated in G and there is a 2-round PSMT protocol Π. Let $view^S$ and $view^R$ be the views of S and R respectively. In Round 1 of Π, $view^S$ and $view^R$ can be different if the adversary corrupts some feedback paths. Since the feedback paths are not reliable, S cannot detect the differences. Thus after Round 2, because Π is perfectly private, with respect to $\mathcal{A}$, we regard $view^S$ as a codeword whose information is the message. Thus $view^R$ is $view^S$ plus an error vector caused by a set $A \in \mathcal{A}$. Since S and R are $3\mathcal{A}$-separated, R cannot correct the errors and decode the message. Thus Π is not perfectly reliable. We have a contradiction.

Next we show a 2-round PSMT protocol under this condition. We let $\mathcal{A}' = \mathcal{A} \cup \{Q\}$ (if $Q \in \mathcal{A}$, then $\mathcal{A}' = \mathcal{A}$). Since S and R are $2\mathcal{A}$-connected with the forward paths, they are $3\mathcal{A}'$-connected in G. The linear code in this protocol is constructed with respect to $\mathcal{A}'$.

2-round directed protocol for a single message s

Round 1 - R to S: R chooses a random k-vector $\mathbf{r}$, and encodes it to get the codeword $\mathbf{c} = EC(\mathbf{r}) = (c_1, \ldots, c_h)$. Suppose that $c_1, \ldots, c_t$ are the entries in $\mathbf{c}$ such that $\psi(c_1, \ldots, c_t) = Q$, the linear code allows all these entries to be independent[5]. R then sends the entries $c_1, \ldots, c_t$ via Q with respect to ψ.

[5] This is possible. See the full version of this paper [1] for more details.

Round 2 - S to R: Upon the entries $c'_1, \ldots, c'_t$ that S receives on Q, S constructs a k-vector $\mathbf{r}'$ such that $c'_1, \ldots, c'_t$ are in the codeword $\mathbf{c}' = EC(\mathbf{r}') = (c'_1, \ldots, c'_h)$. S decodes $r' = DC(\mathbf{r}')$. S then sends $c'_{t+1}, \ldots, c'_h$ via W with respect to ψ and broadcasts $\sigma = s + r'$.

Recovery Phase R receives $c''_{t+1}, \ldots, c''_h$ and σ on W. R constructs an h-vector $\mathbf{x} = (c_1, \ldots, c_t, c''_{t+1}, \ldots, c''_h)$. Thus $\mathbf{x} = \mathbf{c}' + \mathbf{e}$ where $\mathbf{e}$ is an error vector caused by a set $A \in \mathcal{A}'$. Due to the $3\mathcal{A}'$-connectivity, R can decode the information r' of $\mathbf{c}'$ from $\mathbf{x}$ and recover $s = \sigma - r'$. **End.**

Proof of perfect security is omitted. See the full version of this paper [1].

Clearly the TC of this protocol is $O(h)$, and the protocol can transmit ℓ messages with a TC of $O(h\ell)$. □

6 Conclusion and Open Problems

In this paper, we regarded general access structures as a special linear code and exploited its properties to design PSMT protocols in the general adversary model. The construction of our protocols is based on the idea of defining adversary structure over critical paths. We are the first to study interactive PSMT with a constant round complexity. Moreover, the transmission complexity of our protocols is similar to the best protocols that use non-constant rounds, which is quite unexpected. Also our study on PSMT over multiple messages is new in this context.

Evidently, there are still many unknown properties of the linear codes we proposed. The most obvious one is the tight upper bound on h, which is open for decades. Another interesting problem is whether in the presence of non zero invalid error-vectors, it is possible to have a pseudo-dimension that is smaller than $O(h)$.

The TC of our 2-round undirected and 3-round directed protocols for multiple message transmission is $O(hn\ell)$. In [22, 2, 15], the authors used a technique called *generalized broadcast* to reduce the TC by $O(n)$. We wonder if generalized broadcast can further reduce the TC of our protocols to $O(h\ell)$.

Acknowledgment. We would like to thank the anonymous referees for their helpful comments on the earlier version of the paper.

References

1. The full version of this paper will be available on the authors' web pages.
2. S. Agarwal, R. Cramer, and R. de Hann. Asymptotically optimal two-round perfectly secure message transmission. In *Proc. CRYPTO '06*, volume 4117 of *LNCS*, pages 394–408, 2006.
3. J. Benaloh and J. Leichter. Generalized secret sharing and monotone functions. In *Proc. CRYPTO '88*, volume 403 of *LNCS*, pages 27–35, 1988.
4. C. Blundo, A. De Santis, R. De Simone, and U. Vaccaro. Tight bounds on the information rate of secret sharing schemes. *Des. Codes Cryptography*, 11(2):107–122, 1997.

5. R. Cramer, I. Damgård, and U. M. Maurer. General secure multi-party computation from any linear secret-sharing scheme. In *Proc. Eurocrypt '00*, volume 1807 of *LNCS*, pages 316–334, 2000.
6. L. Csirmaz. The size of a share must be large. *J. Cryptography*, 10(4):223–231, 1997. A Preliminary version published in 1995.
7. Y. Desmedt and Y. Wang. Perfectly secure message transmission revisited. In *Proc. Eurocrypt '02*, volume 2332 of *LNCS*, pages 502–517, 2002.
8. Y. Desmedt, Y. Wang, and M. Burmester. A complete characterization of tolerable adversary structures for secure point-to-point transmissions without feedback. In *Proc. ISAAC '05*, volume 3827 of *LNCS*, pages 277–287, 2005.
9. D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly secure message transmission. *J. ACM*, 40(1):17–47, 1993.
10. Y. Frankel and Y. Desmedt. Classification of ideal homomorphic threshold schemes over finite Abelian groups. In *Proc. Eurocrypt '92*, volume 658 of *LNCS*, pages 25–34, 1992.
11. M. Hirt and U. M. Maurer. Player simulation and general adversary structures in perfect multiparty computation. *J. Cryptology*, 13(1):31–60, 2000.
12. M. Ito, A. Saito, and T. Nishizeki. Secret sharing schemes realizing general access structure. In *Proc. IEEE Globecom '87*, pages 99–102, 1987.
13. M. Karchmer and A. Wigderson. On span programs. In *Proc. IEEE Structure in Complexity Theory*, pages 102–111, 1993.
14. M. Kumar, P. Goundan, K. Srinathan, and C. P. Rangan. On perfectly secure communication over arbitrary networks. In *Proc. ACM PODC '02*, pages 293–202, 2002.
15. K. Kurosawa and K. Suzuki. Truly efficient 2-round perfectly secure message transmission scheme. In *Proc. Eurocrypt '08*, volume 4965 of *LNCS*, pages 324–340, 2008. Also available in *IEEE Transaction on Information Theory*, 55(11):5223–5232, 2009.
16. F. J. MacWilliams and N. J. A. Sloane. *The theory of error-correcting codes.* North-Holland Publishing Company, 1978.
17. A. Patra, A. Choudhary, and C. P. Rangan. On communication complexity of secure message transmission in directed networks. In *Proc. ICDCN '10*, volume 5935 of *LNCS*, pages 42–53, 2010.
18. A. Patra, A. Cloudhary, and C. P. Rangan. Brief announcement: perfectly secure message transmission in directed networks re-visited. In *Proc. ACM PODC '09*, pages 278–279, 2009.
19. A. Patra, B. Shankar, A. Choudhary, K. Srinathan, and C. P. Rangan. Perfectly secure message transmission in directed networks tolerating threshold and non threshold adversary. In *Proc. CANS '07*, volume 4856 of *LNCS*, pages 80–101, 2007.
20. A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.
21. G. J. Simmons, W. Jackson, and K. Martin. The geometry of shared secret schemes. *Bulletin of the Institute of Combinatorics and its Applications*, 1(1):71–88, 1991.
22. K. Srinathan, A. Narayanan, and C. P. Rangan. Optimal perfectly secure message transmission. In *Proc. CRYPRO '04*, volume 3152 of *LNCS*, pages 545–561, 2004.
23. M. van Dijk. On the information rate of perfect secret sharing schemes. *Des. Codes Cryptography*, 6(2):143–169, 1995.
24. Q. Yang and Y. Desmedt. Cryptanalysis of secure message transmission protocols with feedback. In *Proc. ICITS '09*, volume 5973 of *LNCS*, 2009.