A General Construction for 1-round δ-RMT and (0, δ)-SMT

Reihaneh Safavi-Naini, Mohammed Ashraful Alam Tuhin, and Pengwei Wang

Department of Computer Science, University of Calgary,
rei@ucalgary.ca, maatuhin@ucalgary.ca, pengwwan@ucalgary.ca

Abstract. In Secure Message Transmission (SMT) problem, a sender $\mathcal{S}$ is connected to a receiver $\mathcal{R}$ through N node disjoint bidirectional paths in the network, t of which are controlled by an adversary with *unlimited computational power*. $\mathcal{S}$ wants to send a message m to $\mathcal{R}$ in a *reliable* and *private* way. It is proved that SMT is possible if and only if $N \geq 2t+1$. In Reliable Message Transmission (RMT) problem, the network setting is the same and the goal is to provide reliability for communication, only. In this paper we focus on 1-round δ-RMT and $(0, \delta)$-SMT where the chance of protocol failure (receiver cannot decode the sent message) is at most δ, and in the case of SMT, privacy is perfect.

We propose a new approach to the construction of 1-round δ-RMT and (0, δ)-SMT for all connectivities $N \geq 2t+1$, using list decodable codes and message authentication codes. Our concrete constructions use folded Reed-Solomon codes and multireceiver message authentication codes. The protocols have optimal transmission rates and provide the highest reliability among all known comparable protocols. Important advantages of these constructions are, (i) they can be adapted to all connectivities, and (ii) have simple and direct security (privacy and reliability) proofs using properties of the underlying codes, and δ can be calculated from parameters of the underlying codes.

We discuss our results in relation to previous work in this area and propose directions for future research.

1 Introduction

In a Secure Message Transmission (SMT) system a sender is connected to a receiver through N wires, t of which are controlled by the adversary. Wires are abstractions of bidirectional node disjoint paths in a network. The adversary's control of a wire is by taking complete control of a node or a link on the path, allowing them to stop, inject or change arbitrarily, the messages that are sent on the path. The goal of the system is to provide *reliability* and *privacy* for the transmitted messages against an adversary with *unlimited computational* power without assuming any prior shared key between the sender and the receiver. In *Perfectly Secure Message Transmission* (PSMT) systems, the adversary will not learn anything about the message, and the receiver always correctly receives the sent message. SMT protocols can have one or more *rounds* and their communication efficiency for a given number of rounds is measured by the *transmission* *rate* which is the total number of communicated bits per one message bit. Protocols with the lowest rate for a given number of rounds, are called *optimal.* The initial motivation for this model was to simulate secure links between nodes in a distributed setting (e.g., multi-party computation [1, 2, 15]), where there are no direct secure links between nodes but there are multiple paths that connects the two nodes. In recent years, however, the protocols, and in particular 1-round SMT protocols, have found other applications including key agreement and key strengthening in wireless sensor networks (e.g., [3, 23, 24]).

SMT protocols are secure against unlimited adversaries. This is particularly important noting that the advent of quantum computers in future will make all secure protocols that rely on computational assumptions such as the hardness of integer factoring and discrete logarithm (including SSH and SSL) completely insecure. SMT model efficiently uses network path redundancy in networks as the main resource of the communicants for secure (private and reliable) communication without requiring any shared secret key.

It has been shown [4] that 1-round PSMT is possible if and only if $N = 3t+1$. That is, only when less than one third of the wires are corrupted. PSMT for $2t + 1 \leq N \leq 3t$ requires more that one round and the interaction significantly increases complexity of their implementations because of the need for maintaining state (in a secure way) and for $N \leq 2t$, it is impossible to have reliable transmission.

To increase the number of corrupted wires that can be tolerated by the protocol without increasing the number of rounds, one may sacrifice some reliability. A 1-round $(0, \delta)$-SMT protocol provides perfect privacy and bounds probability of error in receiving the message by δ. These protocols can be constructed for $N \geq 2t + 1$.

A related scenario, known as *Reliable Message Transmission (RMT)*, is when the only requirement is the reliability of communication assuming the same network and adversary model (N wires, t of which are controlled by the adversary) as the SMT problem. A trivial protocol for reliable transmission when $N \geq 2t+1$ is by sending the message on all the wires and using majority voting at the receiver to recover the correct message. This will correctly recover the message as only $t \leq \frac{N-1}{2}$ wires are corrupted. However the transmission rate of this protocol is N, which grows linearly with N (similar to repetition codes) and so the goal of δ-RMT protocols is to achieve optimal transmission rate (which is *constant* for 1-round when $N = 2t + K$, where $K \geq 1$ is a constant).

In this paper we consider 1-round δ-RMT and 1-round $(0, \delta)$-SMT protocols.

Towards a systematic construction of 1-round δ-RMT and $(0, \delta)$-SMT. All existing optimal 1-round (0, δ)-SMT protocols use complex combinations of secret sharing and authentication systems for message encoding, together with elaborate secret reconstruction and verification algorithms to construct a decoding algorithm for the SMT. A disadvantage of these complex and clever constructions is the difficulty of verifying their properties. It was shown [13] that the proofs of security of the 1-round (0, δ)-SMT protocol in [20] were not correct.

A limitation of these constructions in practice is that a protocol is designed for a specific type of connectivity (for example, $N = 2t+1$, or $N = 2t+K, K > 1$, or $N = (2+c)t, c > \frac{1}{t}$), and when used in a setting with different types of connectivity, the optimality of the protocol cannot be guaranteed. This means that for optimality guarantee, one may need to implement multiple protocols in cases that the connectivity is not known beforehand. This is a common situation when encoding and decoding algorithms are implemented during device manufacturing and without the deployment information. Ideally one would like one optimal construction that provides flexibility to be used with different connectivity that is faced during deployment. Similar observations can be made for optimal 1-round δ-RMT with $N \geq 2t+1$. The only construction with optimal rate is given in [18]. The construction uses a complex combination of secret sharing for encoding and an elaborate verification for the decoding.

In contrast to the above constructions, there is a simple and elegant construction of an optimal 1-round PSMT [8] for $N = 3t+1$, that uses Reed-Solomon (RS) code. The encoding and decoding in this construction are encoding and decoding of RS-codes. The minimum distance of the code is $d = 2t+1$ allowing t (adversarial) errors to be corrected. The dimension $t+1$ of the code ensures perfect privacy for SMT when t wires are corrupted. The construction also works for higher connectivity of the form, $N = 3t+K, K > 1$, and sends K messages instead of one. For this construction the receiver only has to implement the decoder of an RS-code which has many well-known implementations. For less connectivity, $2t+1 \leq N \leq 3t$, 1-round PSMT is not possible. However, it is an open question if it is possible to have simple and modular constructions for δ-RMT and (0, δ)-SMT using known primitives such as RS-codes.

Our contributions

A general construction of a 1-round δ-RMT. We give a general construction of a 1-round δ-RMT for $N \geq 2t+1$ from two components: a list decodable code and a Message Authentication Code (MAC). In a (ρ, L)-list decodable code (LD code) of length n, the number of codewords within distance ρn of any received word is at most L.

The basic idea of the construction is as follows. An information block $I_{\mathcal{S}}$ is first appended with a tag generated by a symmetric key authentication mechanism, to form a message $m_{\mathcal{S}}$, which is then encoded using the LD code, and each component is sent over a wire. LD code allows correction of up to t adversarial errors and so the decoder will obtain a list of L closest codewords to the received word. The key for the authentication mechanism will be generated by the sender and sent along the wires to the receiver and so parts that are sent over the corrupted wires, will be corrupted. The authentication information together with the key information will allow the receiver to use the corresponding verification mechanism to recognize the correct codeword in the decoded list. The authentication mechanism must ensure that despite partly corrupted keys, the receiver will output the correct codeword. We describe our approach and prove a general theorem that proves reliability of the construction, with a value of δ that can be calculated from the parameters of the underlying components.

We then give a concrete construction when $N = 2t + 1$ using a Folded RS-code and a new multireceiver MAC that we propose. The result is an optimal δ-RMT with *the smallest* δ (highest reliability) among all known optimal δ-RMT protocols.

The drawback of this construction is that the receiver algorithm in RMT is exponential. For higher connectivities of the form $N = (2+c)t, c > \frac{1}{t}$ however, we will have an optimal δ-RMT with efficient (polynomial) receiver algorithm. The main challenge in this construction is choosing the authentication mechanism and its parameters, as well as parameters of FRS-code to achieve the required performance. We give details of these selections for $N = 2t + 1$, and for higher connectivities we omit the details because of space. (We will provide details for SMT when $N = (2+c)t, c > \frac{1}{t}$, which gives a good idea of challenges of designing δ-RMT for these connectivities.)

Constructing 1-round $(0, \delta)$*-SMT from FRS-codes.* Although it is possible to give a general construction of $(0, \delta)$-SMT using an approach similar to δ-RMT, for clarity of results and because of space limitations we limit ourselves to concrete constructions. We first describe a construction for a 1-round $(0, \delta)$-SMT for $N = 2t + 1$ using FRS-codes and a multireceiver MAC, and then extend the result to the case where $N = (2 + c)t, c > \frac{1}{t}$. For $N = 2t + K, K > 1$, a similar approach can be used resulting in an optimal 1-round $(0, \delta)$-SMT.

The construction of 1-round PSMT for $N = 3t + 1$ [8] uses an RS-code to encode the message, and the receiver algorithm uses unique decoding algorithm of RS-codes. For connectivity $2t + 1 \leq N \leq 3t$, the minimum distance of the RS-code will be $t + 1 \leq d \leq 2t$, and so unique decoding for t adversarial errors is not possible.

We use *list decoding* to correct errors beyond unique error correcting radius of the code, and use an authentication mechanism to recognize the sent codeword. There are however two major challenges:
(i) In SMT the sender and the receiver *do not share a secret key* and so the key for the authentication mechanism (based on MACs) must be delivered to the receiver over the wires, some of which are corrupted.
(ii) For $N = 2t+1$, and code dimension $k = t+1$ which is dictated by the perfect privacy requirement, the code rate is $R = k/N = \frac{t+1}{2t+1}$ and so the percentage of errors that needs to be corrected is $\rho = \frac{t}{2t+1} = 1 - R$, which is the information theoretic *list decoding capacity* of the code. Codes that can achieve this capacity and have efficient decoders need special construction.

Our general approach that is used for all connectivities works as follows. An information block $I_{\mathcal{S}}$ is first appended with sufficient random pads (to guarantee privacy) and a tag generated by a multireceiver MAC algorithm, to form a message $m_{\mathcal{S}}$, which is then encoded using a Folded Reed-Solomon (FRS) code with well chosen parameters, *that depend on the available connectivity.* FRS-codes [11] are explicit LD codes that achieve list decoding capacity and have efficient decoding algorithm. On each wire, one component of the LD code together with part of the key information for the multireceiver MAC, are sent. The receiver uses the decoding of the FRS-code to recover the list of code vectors that are at distance at most t (in FRS-code) from the received vector, and then uses the key information that are sent over the wires, and are possibly changed by the adversary, to identify the correct message. The final SMT decoding algorithm either outputs the correct message, or outputs **Fail**. That is, *the decoder never outputs an incorrect message.* FRS-codes can be seen as RS-codes with two additional parameters: folding degree and decoding degree. These parameters affect ρ and the efficiency of the code. These parameters are chosen separately for a given connectivity, and in conjunction with the authentication mechanism, to achieve optimal performance for SMT.

To achieve optimal transmission rate for the SMT we have a number of innovations: (i) we use an authentication mechanism that is inspired by *multireceiver message authentication codes* (multireceiver MAC) introduced in [6], (ii) give new constructions for key-efficient multireceiver authentication for *message blocks*, and (iii) use special parsing and packaging of symbols in $m_{\mathcal{S}}$, and design parameters of the MAC and use it for only the information part of $m_{\mathcal{S}}$.

The two new multireceiver MACs effectively compress the authentication information that needs to be sent, and so reduce the communication cost of the protocols.

The value of δ, the success chance of the adversary in resulting the protocol to output **Fail**, is obtained by estimating the length of the decoded list and forgery probability of the MAC. The proof of perfect privacy of the construction is straightforwardly obtained from the choice of the dimension of the FRS-code.

For $N = 2t+1$, however, the SMT decoding requires exponential time because although the list decoding algorithm is efficient, the list has exponential size (in N). The list size and so the decoding cost becomes polynomial for higher connectivities of the form $N = (2+c)t, c > \frac{1}{t}$. The construction in this case uses the same building blocks (FRS-code and MAC based codeword identification) but uses different parameters for the code and a different multireceiver MAC. The transmission rate of the 1-round $(0, \delta)$-SMT, in all cases, is optimal.

An important property of the resulting $(0, \delta)$-SMT protocols is that *they have the lowest* δ and so the highest reliability among all known optimal 1-round (0, δ)-SMT protocols with comparable connectivity (for protocols that output correct messages, or **Fail**.) The SMT construction can be easily adapted to connectivities of the form $N = 2t + K$, where $K > 1$ is a constant. We have omitted the details because of space limitations. Our proposed constructions of multireceiver MACs provide optimal and near optimal (different by a factor of 2) forgery probabilities and are of independent interest.

Advantages of the approach

A general construction for 1-round δ-RMT. LD codes and multireceiver MACs are both well-established primitives with numerous efficient constructions. The advantage of a general construction using these two primitives as building blocks, is that one can choose appropriate constructions for a given setting. Moreover, advances in LD codes and MACs can result in better construction for δ-RMT systems. The instantiation of the LD code with FRS-code is directly adaptable (with revised message structure and code parameters) to $(0, \delta)$-SMT. This means that the main building block of the receiver will stay the same in both cases and development of more efficient decoding for FRS-codes will translate into more efficient receiver algorithms for SMT.

1-round $(0, \delta)$-SMT and RS-codes. FRS-codes are in fact RS-codes with blocks of symbols interpreted as elements of a larger field. The construction of $(0, \delta)$-SMT from FRS-codes provides an elegant and systematic construction for 1-round $(0, \delta)$-SMT *for all connectivities* $(N \geq 2t + 1)$ using RS-codes.

The decoding algorithm is the same for RMT and SMT and in all cases consists of a two step algorithm: list decoding of the FRS-code, and a message verification algorithm based on the MAC for every element of the list. The proof of privacy is based on the properties of the FRS-codes and are intuitive. The proof of reliability (calculation of δ) is also intuitive with concrete values depending on the parameters of the FRS-codes and the MAC.

A unified approach to 1-round δ-RMT and $(0, \delta)$-SMT. The above shows a unified approach for the construction of these two primitives, reliable communication without or with privacy, for all connectivities $N \geq 2t + 1$. This means that the sender and receiver can use a modular construction in which the module with the higher complexity, which is the list decoding module, is implemented once and its parameters are adjusted depending on the required properties (reliability only, or both reliability and privacy) and the choice of the MAC determined by the given N and t.

δ-RMTs are similar to error correcting codes with protection against adversarial errors with the difference that adversary's view is limited to the t corrupted wires. In error correcting codes with protection against adversarial channels, the adversary can see the whole codeword before choosing the error pattern. In δ-RMT the adversary only sees the positions that it corrupts.

An example. For the lowest possible connectivity $N = 2t + 1$, the computation of our 1-round (0, δ)-SMT (also δ-RMT) protocol is not polynomial (in N), but for higher connectivity it becomes polynomial (in N). For example, consider $N = 50$, $t = 20$, $c = 0.5$. For this network our approach will need $q \geq 16000$, q^4 computation, the list size will be q^3 and $\delta = \frac{42}{q^4} = 6(10^{-6})$, where q is the base field size. For $q = 16000$, the δ for the protocol in [22] is 0.0263. whereas, δ for the protocol in [21] is not defined (the protocol needs larger field size). To have the same δ as our protocol, [21] needs field size to be 1.2×10^{14}.

Related Work

Srinathan et al. designed an efficient and optimal 1-round δ-RMT protocol [18]. This protocol uses a complex combination of secret sharing by the sender and elaborate verification of the received information by the receiver to determine the correct message block. There are two optimal and efficient 1-round (0, δ)-SMT protocols for $N = 2t + 1$ [14, 22]. The protocol in [14] is based on the 1-round δ-RMT protocol of [18] and has similar complexity. Moreover, to achieve the optimal transmission rate for higher connectivity, this protocol can not be directly used as the encrypted message blocks are broadcasted. The protocol of [22] suffers from the same problem as that of [14].

On the other hand, for $N = (2+c)t, c > \frac{1}{t}$, there are two optimal and efficient 1-round (0, δ)-SMT protocols [22, 21]. The protocol of [21] uses two SMT protocols in two levels. The first SMT protocol for the lowest connectivity is used many times on different subsets of the wires. The second protocol which works for higher connectivity is applied to virtual wires obtained from the actual physical wires. The shortcoming of this protocol is that to achieve optimal rate, the sender needs to send a very large information block (of size at least N^3 field elements). The protocol in [22] also uses multiple secret sharings in a clever way which results in a lower δ than that of [14].

2 Background and Primitives

In SMT problem, there is an incomplete network, that connects a sender $\mathcal{S}$ to a receiver $\mathcal{R}$. The sender and the receiver are connected by N vertex-disjoint paths, also known as *wires* or channels. The network is undirected and communication on the wires is synchronous and bidirectional. Both $\mathcal{S}$ and $\mathcal{R}$ are honest. The goal is to enable $\mathcal{S}$ to send a message m, drawn from a message space $\mathcal{M}$ with a probability distribution $\Pr(m)$, to $\mathcal{R}$ such that $\mathcal{R}$ receives the message *correctly* and *privately.*

In a *message transmission protocol*, the sender $\mathcal{S}$ chooses m from a message space $\mathcal{M}$ with a probability distribution $\Pr(m)$, and uses a protocol with one or more rounds, to send the message to the receiver. In each protocol round, $\mathcal{S}$ or $\mathcal{R}$, constructs a protocol message that is sent over the wires to the other party. A protocol message is received by the recipient of the round, possibly in a corrupted form, before the next round starts. At the end of the protocol, the receiver outputs a message m', or outputs a **Fail**.

We consider only 1-round protocols. The adversary $\mathcal{A}$ has unlimited computational power and can corrupt and control a subset of wires: the adversary can eavesdrop, block or modify the communication over the corrupted wires. $\mathcal{A}$ can corrupt at most t out of the N wires and the corrupted wires are unknown to $\mathcal{S}$ and $\mathcal{R}$.

Denote by $V_{\mathcal{A}}(M_{\mathcal{S}},\ r_{\mathcal{A}})$ the random variable that denotes the view of the adversary $\mathcal{A}$ when attacking the protocol assuming the sender has chosen $M_{\mathcal{S}}$ and $r_{\mathcal{A}}$ is the random coins of the adversary. Let the statistical distance of two random variables X, Y defined over a set $\mathcal{U}$ be defined as, $\Delta(X, Y) = \frac{1}{2}\sum_{u\in\mathcal{U}} |\Pr[X = u] - \Pr[Y = u]|$.

Definition 1. *A message transmission protocol between $\mathcal{S}$ and $\mathcal{R}$ is an (ε, δ)-Secure Message Transmission ((ε, δ)-SMT) protocol if the following two conditions are satisfied:*

- *Privacy: For every two messages $m_0, m_1 \in \mathcal{M}$ and every $r \in \{0, 1\}^*$ used by the adversary,*
  $\Delta(V_{\mathcal{A}}(m_0, r), V_{\mathcal{A}}(m_1, r)) \leq \varepsilon$, where the probability is over the randomness of $\mathcal{S}$ and $\mathcal{R}$.
- *Reliability: $\mathcal{R}$ outputs the message m with probability $\geq 1 - \delta$, and **Fail** with probability $\leq \delta$. That is, the receiver never outputs an incorrect message and,* $\Pr[$*Receiver outputs* **Fail**$] \leq \delta$.

This is the definition of reliability used by Kurosawa et. al. [13]. The original definition of reliability in [9] however assumes that the receiver always outputs a message m' and δ-reliability is, $\Pr[m' \neq m] \leq \delta$. Kurosawa et al. require that the receiver be sure that the received message is correct. When $\varepsilon = 0$, the protocol is said to achieve *perfect privacy*, and when $\delta = 0$, the protocol is said to achieve *perfect reliability.* A *δ-Reliable Message Transmission (δ-RMT)* protocol is a protocol between $\mathcal{S}$ and $\mathcal{R}$ in the same network setting, and only requiring the reliability of transmission.

It was shown [9] that (0, δ)-SMT is possible if and only if $N \geq 2t + 1$ and 1-round PSMT is possible if and only if $N \geq 3t+1$ [4]. 1-round δ-RMT protocols exist if and only if $N \geq 2t + 1$ [9].

Communication efficiency of RMT and SMT protocols is in terms of the *number of rounds*, and *transmission rate.* The *number of rounds of an SMT protocol* is the number of interactions between $\mathcal{S}$ and $\mathcal{R}$. *Transmission rate of RMT and SMT protocols* is the ratio of the total communication to the length of the message: that is the communication cost of sending one bit.

Lower bounds on transmission rates of 1-round δ-RMT and 1-round $(0, \delta)$-SMT protocols are $\Omega(\frac{N}{N-t})$ [16] and $\Omega(\frac{N}{N-2t})$ [14], respectively. Protocols whose transmission rate asymptotically match the associated lower bounds are called *optimal.* For 1-round δ-RMT with $N = 2t + 1$, the lower bound on the transmission rate is $\Omega(1)$. For 1-round (0, δ)-SMT protocols with $N = 2t + 1$ and $N = (2 + c)t, c > \frac{1}{t}$, optimal protocols must have transmission rates $O(N)$ and $O(1)$, respectively.

Computation efficiency of RMT and SMT protocol is the amount of computation performed by $\mathcal{S}$ and $\mathcal{R}$ throughout the protocol. A protocol that needs exponential (in N) computation for $\mathcal{S}$ and $\mathcal{R}$, is called *inefficient.* Efficient protocols need polynomial (in N) computation.

2.1 Folded Reed-Solomon Codes

A (k, n) linear error correcting code over $\mathbf{F_q}$ is a subspace of dimension k of the n dimensional vector space over $\mathbf{F_q}$. The *information rate* of a linear error correcting code is, $R = \frac{k}{n}$. A decoder takes a corrupted word and determines the most likely codeword that was sent. In unique decoding, the closest (Hamming distance) codeword to the received one is found. In list decoding, a list of codewords within a radius from the received word is found. For a constant ρ, let ρn denotes the number of errors that can be corrected by the decoder.

Definition 2. *A code C with the encoding function $LD : {\mathbf{F_q}}^k \rightarrow {\mathbf{F_q}}^n$ is (ρ, L)-list decodable (LD) if the number of codewords within distance ρn of any received word is at most L. That is for every word $y \in {\mathbf{F_q}}^n$, there are at most L codewords at distance ρn (where ρ is the relative distance) or less from y.*

The *list decoding capacity* $\rho_{cap}(R)$ of a code with rate R is the information theoretic limit of list decodability and is given by $\rho_{cap}(R) = 1 - R = 2\rho_U(R)$, where $\rho_U(R) = (1 - R)/2$, is the unique decoding radius of the code. It is shown [7] that for sufficiently large alphabet size $\rho_{cap}(R)$ can reach $1 - R - \epsilon$. So for any code rate, *list decoding can potentially correct twice as many errors as unique decoding [11].*

To achieve this potential however, one needs special constructions that guarantee that the list size is bounded by L. Efficient list decoding algorithms are polynomial time. *Folded Reed-Solomon codes (FRS-codes)*, proposed by Guruswami et al. [11], is a special type of RS-codes that corrects up to a fraction $\rho = 1 - R - \epsilon$ of errors, for any rate R and arbitrary $\epsilon > 0$ using a polynomial time list decoding algorithm. The list size, however for some parameter choices of the code, becomes exponential. The codes are RS-codes over a field $\mathbf{F_q}$, but viewed as an RS-code over a larger field ${\mathbf{F_q}}^u$, by careful bundling of codeword symbols. To reach within ϵ of list decoding capacity, FRS-codes need an alphabet of size $n^{O(1/\epsilon)}$, where n is the block length. Authors argue that this alphabet size is "in the same ballpark as the best possible".

Description of Folded Reed-Solomon codes. FRS encoding and decoding are defined using an RS-code of length n and dimension k over a finite field $\mathbf{F_q}$, using two parameters, u, the *folding parameter* and s, which determines the $(s+1)$-variate interpolation used in the decoding. Let u be an integer, called the *folding parameter*, such that n in divisible by u. n is chosen as the largest integer that is less than $q = |\mathbf{F_q}|$ and is divisible by u.

Let γ be a generator of ${\mathbf{F_q}}^*$, the multiplicative group of the field $\mathbf{F_q}$. A codeword $(f(1), f(\gamma), f(\gamma^2), \cdots, f(\gamma^{n-1}))$ of $RS[n,k]$ is the evaluation of a polynomial $f(x)$, of degree at most $k-1$ over $\mathbf{F_q}$, at the (ordered) points $1, \gamma, \gamma^2, \cdots, \gamma^{n-1}$.

Definition 3. *The u-folded FRS-code is a code with block length $N = n/u$ over ${\mathbf{F_q}}^u$. The encoding of a message, represented by a polynomial $f(x)$ of degree at most $k-1$ over $\mathbf{F_q}$, is obtained as u-tuples, $(f(\gamma^{ju}), f(\gamma^{ju+1}), \cdots, f(\gamma^{ju+u-1}))$, for $0 \le j < N$. In other words, a codeword of the u-folded RS-code is in one-to-one correspondence with codewords of the RS-code C, and is obtained by grouping consecutive $u-$tuples of components of C.*

$$\begin{bmatrix} f(1) & f(\gamma^u) & \cdots & f(\gamma^{u(N-1)}) \\ f(\gamma) & f(\gamma^{u+1}) & \cdots & f(\gamma^{u(N-1)+1}) \\ \vdots & \vdots & \ddots & \vdots \\ f(\gamma^{u-1}) & f(\gamma^{2u-1}) & \cdots & f(\gamma^{uN-1}) \end{bmatrix}$$

Decoding u-folded RS-code uses $(s+1)$-variate interpolation followed by a list pruning step. In the Appendix A, we give an outline of the original decoding algorithm.

Linear-algebraic list decoding. In [10] a variant decoding for FRS-code is given in which the interpolation uses polynomial $Q(X, Y_1, \cdots, Y_s)$ where degree of Y_i is 1. This variant has a simpler exposition and allows a simpler way of choosing the code parameter, s and u. However it can correct less errors. By appropriate choice of s and u, the code can reach $1 - R - \epsilon$ radius and so asymptotically is optimal. Lemma 1 below, given in [10], gives the condition that needs to be satisfied by the two parameters and the number of errors to be corrected.

Lemma 1. *In linear-algebraic list decoding, for every integer u and s, the linear interpolation FRS decoding algorithm successfully list decodes to a radius $N-T$ as long as the agreement parameter T satisfies:*

$$T \geq N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}).$$

T is the number of correct positions. The algorithm outputs a list of size at most $|\mathbf{F_q}|^{s-1} = q^{s-1}$ codewords.

2.2 Multireceiver message authentication codes

A one-time MAC in information-theoretic setting, is a shared key cryptographic primitive, defined by two functions: a MAC function that takes a message $m \in \mathcal{M}$ and the shared key $k_{MAC} \in \mathcal{K}$ and outputs a tag $MAC(m, k_{MAC})$, which is appended to the message, and a verification function, $V((m', x'), k_{MAC})$, which outputs 1 if (m', x') is a valid pair for the key k_{MAC}, and 0, otherwise. The following definition is for security of one-time MAC.

Definition 4. *A one-time MAC, $MAC : q^{l_{msg}} \times q^{l_{key}} \rightarrow q^{l_{tag}}$ has forgery probability γ if the best success chance of a computationally unbounded adversary with access to a message and tag pair $(m, x), x = MAC(m, k_{MAC})$, to construct a different pair (m', x') where $m \neq m'$, and $V((m', x'), k_{MAC}) = 1$ is at most γ, where the probability is taken over all keys.*

Multireceiver authentication codes [6] allow a sender to efficiently send a message to a group of N receivers such that each receiver can individually verify the message, using his individual shared key k_i with the sender. The sender is honest but upto t receivers can be corrupted and attempt to forge a message to be acceptable by an uncorrupted receiver. In a $(t+1, N)$-multireceiver message authentication system, there are N receivers and at most t receivers can be corrupted.

Definition 5. *A one-time $(t+1, N)$-multireceiver authentication code (multireceiver MAC) with N receivers and $k_{MAC} = (k_S, k_1, ..., k_N)$, is γ-secure if the best success chance of any colluding set of receivers (size at most t) with access to a message and tag pair, $(m, x, x = MAC(m, k_S))$ in forging a different message, tag pair (m', x'), where $m \neq m'$, and $V_i((m', x'), k_i) = 1$, is at most γ, and the probability is over all unknown keys.*

2.3 New Constructions for Multireceiver MAC

The basic construction of $(t+1, N)$-multireceiver MACs is for authenticating a single message. To authenticate a block of messages one can use a one-time multireceiver MAC multiple times, or for more efficiency, use separately designed multireceiver MAC for message blocks. In the following, we give two new constructions for $(t+1, N)$-multireceiver MACs for message blocks that are used in the RMT and SMT constructions of this work. Construction 1 is a generalization of [6] for $d > 1$ messages. Construction 2 is built over a brand new MAC.

Construction 1.

Let $m = (m_1, \cdots, m_d)$, where $m_i \in \mathbf{F_q}^{s'}, i = 1, ..., d$, be the message block.

- *Key distribution:* A Trusted Initializer does the following: (i) randomly generates $d+1$ polynomials $P_1(z), P_2(z), \cdots, P_{d+1}(z)$, each of degree at most t, over $\mathbf{F_q}^{s'}$; chooses N random distinct elements $z_1, z_2, \cdots, z_N$, where $z_i \in \mathbf{F_q}^{s'}, i = 1, ..., N$; makes $z_1, z_2, \cdots, z_N$ public, assigns z_i to receiver i and privately sends $k_i = (P_1(z_i), P_2(z_i), \cdots, P_{d+1}(z_i))$ to receiver i, for $1 \leq i \leq N$ and to the sender.
- *Constructing authenticated messages:* The sender computes the authentication tag as:
$$A(z) = P_1(z)m_1 + P_2(z)m_2 + \cdots + P_d(z)m_d + P_{d+1}(z).$$
The authenticated messages consist of the message block and the tag polynomial, $((m_1, m_2, \cdots, m_d), A(z))$.
- *Verification:* Receiver i accepts $(m_1, m_2, \cdots, m_d, A(z))$ if and only if $A(z_i) = P_1(z_i)m_1 + P_2(z_i)m_2 + \cdots + P_d(z_i)m_d + P_{d+1}(z_i) \mod q^{s'}$.

The above construction is a $(t+1, N)$-multireceiver MAC for authentication of a block of size d. The size of the tag is $t+1$ elements of $\mathbf{F_q}^{s'}$ and so only depends on the collusion size (rather than the total number of receivers).

The following Theorem is proved in the Appendix B.

Theorem 1. *For construction 1, the forgery probability is bounded as $\gamma \leq q^{-s'}$.*

Construction 2.

This multireceiver MAC is built on a new one-time MAC which has message block size $\binom{t+2}{2} - 1$ (each block element from $\mathbf{F_q}^{s'}$) and has forgery probability bounded by $\frac{2}{q^{s'}}$.

- *Key distribution:* Same as Construction 1, with $d = t$.
- *Constructing authenticated messages:* For $m = (m_1, m_2, ..., m_{\binom{t+2}{2}-1})$, the sender computes,
$$\begin{aligned} A(z) = {} & m_1P_1(z) + \cdots + m_tP_t(z) + m_{t+1}P_1(z)^2 + \cdots + m_{2t}P_t(z)^2 \\ & + m_{2t+1}P_1(z)P_2(z) + \cdots + m_{\binom{t+2}{2}-1}P_{t-1}(z)P_t(z) + P_{t+1}(z). \end{aligned}$$
- *Verification:* Receiver i accepts $(m_1, m_2, \cdots, m_{\binom{t+2}{2}-1}, A(z))$ if and only if
$$\begin{aligned} A(z_i) = {} & m_1P_1(z_i) + \cdots + m_tP_t(z_i) + m_{t+1}P_1(z_i)^2 + \cdots \\ & + m_{\binom{t+2}{2}-1}P_{t-1}(z_i)P_t(z_i) + P_{t+1}(z_i). \end{aligned}$$

Here $m_i \in \mathbf{F_q}^{s'}$, and $P_1(z), P_2(z), \cdots, P_{t+1}(z)$ are polynomials of degree at most t over $\mathbf{F_q}^{s'}$. The MAC function is a linear sum (coefficients being the message block) of all products of at most two polynomials from the set, $\{P_1(z), P_2(z), \cdots, P_t(z)\}$. Finally $P_{t+1}(z)$ is used to mask the result. The final MAC value is a polynomial over $\mathbf{F_q}^{s'}$. The size of the message block (over $\mathbf{F_q}^{s'}$) that is authenticated by the MAC, is $\binom{t+2}{2} - 1$.

Theorem 2. *For construction 2, the forgery probability is bounded as $\gamma \leq 2q^{-s'}$.*

The proof outline is provided in the Appendix B.

3 Construction of 1-round δ-RMT for $N \geq 2t+1$

Construction 3: A general construction of 1-round δ-RMT. The protocol requires a (ρ, L) LD code of dimension k and length N over $\mathbf{F_q}$, with $\rho = \frac{t}{N}$ and list size L, and a $(t+1, N)$-multireceiver MAC with message space ${\mathbf{F_q}}^{k'}, k' = k - l_{tag} < k$, and forgery probability ϵ. Here l_{tag} is the length of tag in terms of ${\mathbf{F_q}}^{s'}$ elements.

- **Sender Algorithm**:
  1) Securely generates keys $(k_{\mathcal{S}}, k_1, ..., k_N)$ for a multireceiver MAC and assigns the key k_i to the i^{th} wire, W_i.
  2) Constructs the message block $m_{\mathcal{S}}, m_{\mathcal{S}} \in {\mathbf{F_q}}^k$ to be sent to the receiver as,
  $$m_{\mathcal{S}} = (I_{\mathcal{S}}, MAC(I_{\mathcal{S}}, k_{\mathcal{S}})). \qquad (1)$$
  The sender constructs the codeword $c_{\mathcal{S}}$ of the LD code by encoding $m_{\mathcal{S}}$ as $c_{\mathcal{S}} = LD(m_{\mathcal{S}})$. The sender sends the i^{th} component of the codeword $c_{\mathcal{S}}$, and k_i through wire W_i. $k_{\mathcal{S}}$ is kept by the sender.
- **Receiver Algorithm**:
  1) Parses the received (corrupted) N-vector; separates the key k_i from the i^{th} component (possibly corrupted) of the received word for $i = 1, \cdots, N$; constructs the corrupted codeword and uses LD decoding algorithm to obtain a list (of size L) of codewords that are at distances at most $\rho = \frac{t}{N}$ from the received word. *The list will always include the correct codeword.*
  2) To identify the sent codeword, the receiver (i) parses each codeword in the list into a message, tag pair $(\widehat{m_i}, \widehat{t_i}), i = 1, \cdots, L$; (ii) for each message $\widehat{m_i}$, uses all keys $k_j, j = 1, ..., N$, and checks the verification equations $V_j(\widehat{m_i}, k_j) = \widehat{t_i}$. The message is accepted if at least $t+1$ verification equations are passed; otherwise the codeword is rejected.
  The decoding algorithm of the SMT succeeds if there is a unique codeword that is accepted by the verification algorithm above. Otherwise, the receiver outputs a **Fail**.

Theorem 3. *The above construction is a 1-round δ-RMT protocol for $N = 2t+1$, with $\delta = 1 - (L-1)\epsilon$ and transmission rate $\frac{N(1+|k_i|)}{k - l_{tag}}$.*

Proof. (outline): The sender encodes the information block $I_{\mathcal{S}}$ to $m_{\mathcal{S}} \in {\mathbf{F_q}}^k$ and constructs the codeword $c_{\mathcal{S}}$ for $(m_{\mathcal{S}}, MAC(m_{\mathcal{S}}, k_{\mathcal{S}}))$. The receiver decodes a list of at most L codewords and candidate messages of the form $(\hat{m}, \hat{tag})$. For all possible t errors that the adversary adds, the success chance that another message of this format is accepted is upper bounded by the success probability of forging MAC for one uncorrupted wire, which is ϵ. The adversary succeeds if one element of the decoded list can be forged.

The code length is N and through wire W_i, the i^{th} component of $c_{\mathcal{S}}$ and the multireceiver key k_i is sent. So the number of field elements transmitted is $N(1 + |k_i|)$. The length of the information block is $k - l_{tag}$, so the transmission rate is $\frac{N(1+|k_i|)}{k-l_{tag}}$. Here $|k_i|$ is the length of the key for the i^{th} wire, and k is the dimension of the code. □

3.1 An Optimal δ-RMT

In the following we give an instantiation of the general construction above using (i) an FRS-code for the LD code, and (ii) Construction 2 for multireceiver MAC. The multireceiver MAC allows authentication of a message block of size $\sim t^2$ field elements requiring $\sim t$ field element for encoding of $m_{\mathcal{S}}$ as, to be sent on each wire, resulting in optimal transmission rate.

Selecting Parameters of the FRS-code. Let $N = 2t+1$ for the RMT. We consider a u-folded RS-code of length N with length $n = Nu$ for the underlying RS-code. Using the message format in (1) and using a block of size $(\binom{t+2}{2} - 1)$ of elements of $\mathbf{F_q}^{s'}$ for information block we will have the required dimension for the code as,

$$k = |m_{\mathcal{S}}| = (\binom{t+2}{2} - 1)s' + ts' + s' = \frac{s'(t^2+5t+2)}{2}. \tag{2}$$

For a code of length $N = 2t+1$ and dimension k as (2), we must choose folding parameter u, number of decoding variable s, and the finite field sizes s' and q, to ensure that decoding succeeds for linear-algebraic decoding (outlined in Section 2.1) for radius $\rho = t/N$ (t errors in the FRS-code). That is, the inequality (3) below, is satisfied.

$$t+1 \geq N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}). \tag{3}$$

Let $0 < \sigma < 1$ and set $s = \frac{N}{\sigma} - 1$. Furthermore, choose $s' = \left\lfloor \frac{2u(t+1-2\sigma)}{t^2+5t+2} \right\rfloor$, and $u > s^2 - 1$. The following shows that with these choices of parameters inequality (3) is satisfied. We have,

$$t+1 > t+1-\sigma = \sigma + (t+1-2\sigma) \overset{(a)}{=} \sigma + \frac{s'(t^2+5t+2)}{2u} \overset{(b)}{=} \sigma + \frac{k}{u}, \tag{4}$$

where (a) is because of the choice of s' and (b) is because of the value of k in (2). Note that because $s \geq 1$, we have $\frac{1}{u} > \frac{s}{s+1}\frac{1}{u-s+1}$ and so, (4) gives the following:

$$t+1 > \sigma + \frac{k}{u} > \sigma + \frac{s}{s+1}\frac{k}{u-s+1} \overset{(c)}{=} N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}),$$

where (c) is by using the value of s and because the rate of the code is $R = \frac{k}{n} = \frac{k}{Nu}$. Finally we can choose q to be the smallest prime that is bigger than the codeword length $n = Nu$.

The Protocol. The construction uses (i) an FRS-code with parameters u and s obtained above and (ii) Construction 2 of the multireceiver MAC in Section 2.3. The final protocol is as follows.

- **Sender Algorithm**:
  1. Uses the key generation algorithm of Construction 2 and obtains for wire $W_i, i = 1, ..., N$, the associated key,

$$k_i = (P_1(z_i), P_2(z_i), \cdots, P_{t+1}(z_i)).$$

The tag for the information part $I_{\mathcal{S}} = (m_0, m_1, \cdots, m_{(\binom{t+2}{2}-1)}), m_i \in \mathbf{F_q}^{s'}$:

$$A(z) = m_1 P_1(z) + \cdots + m_t P_t(z) + m_{t+1} P_1(z)^2 + \cdots + m_{2t} P_t(z)^2 \\ + m_{2t+1} P_1(z) P_2(z) + \cdots + m_{\binom{t+2}{2}-1} P_{t-1}(z) P_t(z) + P_{t+1}(z).$$

2. The message $m_{\mathcal{S}}$ is of the form (1), $m_{\mathcal{S}} = (m_0, m_1, \cdots, m_{(\binom{t+2}{2}-1)}, A(z))$. The dimension of the FRS-code is $(\binom{t+2}{2} - 1)s' + ts' + s'$ over $\mathbf{F_q}$, where $s' = \left\lfloor \frac{2u(t+1-2\sigma)}{t^2+5t+2} \right\rfloor$.

3. The sender encodes the message to a codeword $c_{\mathcal{S}}$ using the FRS encoding algorithm. Wire j, $1 \leq j \leq 2t+1$, transmits the j^{th} component of $c_{\mathcal{S}}$ and k_j.

– **Receiver Algorithm**: Uses the two step decoding of Construction 3 for FRS-code as the LD code, and Construction 2 as the multireceiver MAC. The algorithm outputs the correct message or **Fail**.

Theorem 4. *The above construction is a δ-RMT with $\delta = \frac{2(t+1)}{q^{s'-s+1}}$, which is equal to $\frac{N+1}{q^{s'-s}}$, when $N = 2t+1$. The transmission rate is constant.*

The proof is given in the Appendix B.

Comparison with Related Work

For $N = 2t+1$, this protocol has $\delta = \frac{N+1}{q}$. The value of δ for the only other known optimal 1-round δ-RMT protocol [14], is $\frac{N^2(N-1)}{q}$ ($q \geq \frac{N^2(N-1)}{\delta}$). The field size required in our construction is Nu.

Table 1 compares our protocol with the protocol in [14]. For simplicity of comparison we have used $s' = s$, resulting in $\delta = \frac{t+1}{q}$. The comparison shows that for all connectivities the proposed protocol has a much higher reliability. The field size although asymptotically is larger (N^4 and N^3, respectively) for concrete values (for example $N < 1000$, and $\delta < 10^{-3}$) could be smaller or comparable. Decoder efficiency for higher connectivities is the same. For $N = 2t+1$, however, our protocol has exponential cost.

Table 1. Comparison of 1-round δ-RMT protocols for different connectivities which never outputs an incorrect message; here Comp. refers to computation complexity, *Poly.* refers to polynomial (in N), *Exp.* refers to exponential (in N), and $\mathbf{F_q}$ is the field.

Author	**Comp.** $N = 2t+1$	**Comp.** $N = (2+c)t$	q	δ	**Optimality**
[14]	*Poly.*	*Poly.*	$\geq\approx \frac{N^3}{\delta}$	$\leq\approx \frac{N^3}{q}$	Yes
This Work	*Exp.*	*Poly.*	$\geq Nu$	$\leq \frac{2(t+1)}{q} \approx \frac{N+1}{q}$	Yes

4 1-round $(0, \delta)$-SMT

To use the approach of Construction 3 for $(0, \delta)$-SMT, one needs to ensure that the view of the adversary does not leak any information about the information block $I_{\mathcal{S}}$. Using FRS-code for LD code allows us to achieve this goal by choosing the dimension of the code to be at least $t+1$. This is because the knowledge of any t components of the FRS-codes, leaves the remaining $N-t \geq t+1$ components that carry the information block completely uncertain. Code parameters need to be chosen such that decoding up to $\rho = \frac{t}{N}$ is achievable. For $N = 2t+1$ given in Section 4.1, this requires the FRS-code to achieve the list decoding capacity. The construction for $N = (2+c)t, c > \frac{1}{t}$, given in Section 4.2, uses Construction 2 for multireceiver MAC to allow easier calculation of code parameters while maintaining optimal asymptotic performance. Same approach can also be used for connectivity $N = 2t + K$, where $K > 1$ is a constant. Details are omitted because of space.

In all cases, decoding is the two step Receiver Algorithm of Construction 3.

4.1 A construction for 1-round $(0, \delta)$-SMT for $N = 2t + 1$

The construction uses the approach of Construction 3, but with a different message structure to guarantee perfect privacy.

Message Structure. The message $m_{\mathcal{S}}$ consists of three parts: (i) information part $I_{\mathcal{S}} = (m_0, m_1, \cdots, m_{\sigma u-1}), m_i \in \mathbf{F_q}^{s'}$; (ii) ut random elements $(a_1, a_2, \cdots, a_{ut})$, $a_i \in \mathbf{F_q}$ that are used to ensure privacy that the ut captured components do not reveal anything about $I_{\mathcal{S}}$; (iii) $MAC(X, k_S)$ where $X = (m_0, m_1, \cdots, m_{\sigma u-1}, a_1, ..., a_{s'd-\sigma u})$.

That is,

$$m_{\mathcal{S}} = (m_0, m_1, \cdots, m_{\sigma u-1}, a_1, a_2, \cdots, a_{ut}, MAC(X, k_S)).$$

Here σ is a positive constant. To have optimal rate, the information block size must be a constant fraction of u. Using Construction 1, this size is $s'd$, where d is message length for the MAC. To find s', the information block (I_S) and the first $s'd - \sigma u$ elements a_i are used to form d blocks of size s', where $d = \lceil \frac{\sigma u}{s'} \rceil$. Each s' block is interpreted as an element of $\mathbf{F_q}^{s'}$ and MAC calculations are performed over $\mathbf{F_q}^{s'}$. The ut random elements appended to $I_{\mathcal{S}}$ will ensure perfect privacy. The MAC is only computed on X. The total length of $m_{\mathcal{S}}$ to be encoded by the FRS-code is $ut + \sigma u + s'(t+1)$ were $\sigma < 1$ is a constant. The key k_i for wire i consists of $d+1$ elements $P_1(z_i), P_2(z_i), \cdots, P_{d+1}(z_i)$ over $\mathbf{F_q}^{s'}$. The multireceiver MAC value for X is:

$$MAC(X, k_S) = A(z) = P_1(z)x_1 + P_2(z)x_2 + \cdots + P_d(z)x_d + P_{d+1}(z).$$

The codeword of the FRS-code that is constructed for $m_{\mathcal{S}}$, will have N components, each an element of $\mathbf{F_q}^u$. The adversary's view will contain only t elements of $\mathbf{F_q}^u$ and will be independent from $I_{\mathcal{S}}$.

Parameters of the FRS-code. The u-folded RS-code will have $N = 2t + 1, n = Nu$, and $k = ut + \sigma u + ts' + s'$. We must choose u, s, and the field size q, to ensure that decoding succeeds for linear-algebraic decoding (outlined in Section 2.1) up to radius $\rho = t/N$.

According to Lemma 1,

$$t + 1 \geq N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}).$$

We set the parameter $s = (N/\sigma) - 1$, $u > s^2 - 1$, and $s' = \left\lfloor \frac{u(1-3\sigma)}{t+1} \right\rfloor$, where $0 < \sigma < \frac{1}{3}$ is a positive constant. By using these values of σ, s', u, one can verify that the inequality is satisfied (See Appendix C.1 for details). Finally we can choose q to be the smallest prime that is bigger than the codeword length $n = Nu$.

Construction 4: (0, δ)-SMT protocol for $N = 2t + 1$.

- **Sender Algorithm**:
  1. Uses the key generation of Construction 1 and obtains for wire $W_i, i = 1, ..., N$, the associated key,
  $$k_i = (P_1(z_i), P_2(z_i), \cdots, P_{d+1}(z_i)).$$
  2. Constructs $m_{\mathcal{S}}$: Forms $I_{\mathcal{S}} = (m_0, m_1, \cdots, m_{\sigma u-1}), m_i \in \mathbf{F_q}^{s'}$ and calculates the tag,
  $$A(z) = x_1 P_1(z) + x_2 P_2(z) + \cdots + x_d P_d(z) + P_{d+1}(z),$$
  for d and s' chosen as above. Here $X = (m_0, ..., m_{\sigma u-1}, a_1, ..., a_{s'd-\sigma u})$. $m_{\mathcal{S}}$ is of the form (1) given by, $(m_0, m_1, \cdots, m_{\sigma u-1}, a_1, a_2, \cdots, a_{ut}, A(z))$.
  3. Constructs $c_{\mathcal{S}}$ and message transcript: The sender encodes the message to a codeword $c_{\mathcal{S}}$ using the FRS encoding algorithm. Wire j, $1 \leq j \leq 2t+1$, transmits the j^{th} component of $c_{\mathcal{S}}$ and k_j.
- **Receiver Algorithm**: Uses the decoding algorithm of Construction 3.

Theorem 5. *The SMT protocol described above is a $(0, \delta)$-SMT for $N = 2t+1$, with $\delta = \frac{t+1}{q^{s'-s+1}}$.*

The proof outline is given in Appendix B.
Transmission rate: The transmission rate is $\frac{uN+(s'd+s')N}{\sigma u} = \mathcal{O}(N)$ and it is optimal for 1-round (0, δ)-SMT for $N = 2t + 1$.
Computation complexity: The list size is at most q^{s-1}. Each element of the list must be verified and so the complexity of SMT decoding algorithm is $\mathcal{O}(q^N)$ (as $s = O(N)$).

Comparison with Related Work Table 2 compares the protocol with 1-round (0, δ)-SMT protocols that have the property that the output is either the correct message or **Fail**. For simplicity of comparison we have used $s' = s$, resulting in $\delta = \frac{t+1}{q} = \frac{N+1}{2q}$, when $N = 2t + 1$. The table shows that δ for this construction is the lowest. The minimum field size however is larger and decoding is computationally inefficient. In Section 4.2 we show that both these shortcomings can be removed for higher connectivities.

4.2 1-round $(0, \delta)$-SMT for $N = 2t + ct, c > 1/t$

Let $N = 2t + ct, c > \frac{1}{t}$. We use the same approach as Construction 4, using Construction 2 for multireceiver MAC and choose parameters of the FRS-code and multireceiver MAC such that the SMT construction has optimal rate and efficient computation. The message $m_{\mathcal{S}}$ has the format of (1) and can be written as $(m_0, m_1, \cdots, m_{(\binom{t+2}{2}-1)}, a_1, a_2, \cdots, a_{ut}, MAC(X, k_S))$, where $X = I_{\mathcal{S}} =$

Table 2. Comparison of 1-round $(0, \delta)$-SMT protocols for $N = 2t + 1$; here Comp. refers to computation complexity, Poly. refers to polynomial (in N), Exp. refers to exponential (in N) and $\mathbf{F_q}$ is the field. The protocols never output incorrect message. Here b is a constant and λ is the probability that the cheater wins in a secret sharing scheme with a cheater.

Author	**Comp.**	q	δ	**Optimality**
[13]	*Exp.*	$> N$	$\leq ((\binom{N}{t+1}) - 1)\lambda \approx N^{(N+1)/2}\lambda$	Yes
[19]	*Poly.*	$\geq \frac{2N^3}{\delta}$	$\leq \frac{N^3}{q}$	Yes
[5]	*Poly.*	$\geq bt(t+1) \approx N^2$	$\leq \frac{t(t+1)}{q} \approx \frac{N^2}{q}$	No
[22]	*Poly.*	$\geq bt(t+1) \approx N^2$	$\leq \frac{t(t+1)}{q} \approx \frac{N^2}{q}$	Yes
This Work	*Exp.*	$\geq Nu \approx N^4$	$\leq \frac{t+1}{q} \approx \frac{N+1}{2q}$	Yes

$(m_0, m_1, \cdots, m_{((\binom{t+2}{2})-1)})$ is the information block. The MAC function is over ${\mathbf{F_q}}^s$, where s is the parameter of FRS decoding (instead of ${\mathbf{F_q}}^{s'}$ for $N = 2t+1$). The tag value is:

$$\begin{aligned} A(z) &= m_1 P_1(z) + \cdots + m_t P_t(z) + m_{t+1} P_1(z)^2 + \cdots + m_{2t} P_t(z)^2 \\ &+ m_{2t+1} P_1(z) P_2(z) + \cdots + m_{\binom{t+2}{2}-1} P_{t-1}(z) P_t(z) + P_{t+1}(z), \end{aligned}$$

where $m_i \in {\mathbf{F_q}}^s$ and polynomials $P_1(z), P_2(z), \cdots, P_{t+1}(z)$ are over ${\mathbf{F_q}}^s$ and have degree t.

Parameters s, u, and q are chosen to allow the receiver to decode up to t errors. The number of correct wires $t + ct$ must satisfy,

$$t + ct \geq N\left(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}\right).$$

For a constant c_0, let $u = c_0 t$. We show that list decoding upto $\rho = \frac{t}{N}$ is possible for a constant value of $s = s_0$ when $t > 7c_0 + 2s_0 + 1$, and the value of c satisfies $c > \frac{s_0}{c_0} + \frac{1}{s_0}$. The details are in the Appendix C.2. Table 4.2 below gives example values for s_0 and c, and the size of the resulting list. The complete protocol is given below.

Table 3. Values of c and the list size for different values of s_0

s_0	list size	c
$s_0 = 1$	q^0	$c \approx \frac{1}{c_0} + 1$
$s_0 = 2$	q	$c \approx \frac{2}{c_0} + 1/2$
$s_0 = 3$	q^2	$c \approx \frac{3}{c_0} + 1/3$

SMT Protocol for $N = (2 + c)t, c > \frac{1}{t}$.

- **Sender Algorithm**:
  1. Uses the key generation of Construction 2 and obtains for wire $W_i, i = 1, ..., N$, the associated key,
  $$k_i = (P_1(z_i), P_2(z_i), \cdots, P_{t+1}(z_i)).$$
  2. The message $m_{\mathcal{S}}$ is,
  $$m_{\mathcal{S}} = (m_0, m_1, \cdots, m_{(\binom{t+2}{2}-1)}, a_1, a_2, ..., a_{ut}, A(z)),$$
  where the tag for the information block $I_{\mathcal{S}} = (m_0, m_1, \cdots, m_{(\binom{t+2}{2}-1)}), m_i \in \mathbf{F_q}^{s_0}$ is,
  $$\begin{aligned} A(z) = {} & m_1 P_1(z) + \cdots + m_t P_t(z) + m_{t+1} P_1(z)^2 + \cdots + m_{2t} P_t(z)^2 \\ & + m_{2t+1} P_1(z) P_2(z) + \cdots + m_{\binom{t+2}{2}-1} P_{t-1}(z) P_t(z) + P_{t+1}(z). \end{aligned}$$
  3. The FRS-code is over $\mathbf{F_q}^u$ and has dimension $k = ut + (\binom{t+2}{2} - 1)s_0 + ts_0 + s_0$. The sender encodes the message to a codeword $c_{\mathcal{S}}$ using the FRS encoding algorithm. Wire j, $1 \leq j \leq 2t + ct$, transmits the j^{th} component of $c_{\mathcal{S}}$ and k_j.
- **Receiver Algorithm**:
  Uses the SMT decoding algorithm of Construction 3.

Theorem 6. *The protocol above is a 1-round (0, δ)-SMT for $N = (2+c)t, c > \frac{1}{t}$ with optimal transmission rate, and has efficient (polynomial time) decoding. The value of δ is given by $\frac{2(t+1)}{q}$ and is the smallest among all known protocols with the same connectivity.*

The proof outline is given in the Appendix B.

Comparison with Related Work

There has been two other optimal (transmission rate) and efficient (computation) 1-round (0, δ)-SMT protocols for higher connectivity ($N = (2 + c)t, c > \frac{1}{t}$) [21, 22]. The protocol presented in Section 4.2 has the least δ. The comparison of these protocols is outlined in Table 4.

Table 4. Comparison of 1-round $(0, \delta)$-SMT protocols for $N = (2 + c)t, c > \frac{1}{t}$; here Comp. refers to computation complexity, Poly. refers to polynomial (in N), Exp. refers to exponential (in N) and $\mathbf{F_q}$ is the field. Here ν is a parameter used in wire-virtualization which refers to the number of *physical* wires in each *virtual* wire.

References	**Comp.**	δ	**Optimality**	**Outputs Incorrect Message**
[21]	*Poly.*	$\leq \frac{N^\nu t(t+1)}{q} \approx \frac{N^{\nu+2}}{q}$	Yes	Yes [1]
[22]	*Poly.*	$\leq \frac{t(t+1)}{q} \approx \frac{N^2}{q}$	Yes	No
This Work	*Poly.*	$\leq \frac{2(t+1)}{q} \approx \frac{N}{q}$	Yes	No

[1] The authors in [21] mention that their protocol can be modified to output only correct message block by using a different sub-protocol.

5 Concluding remarks

We showed a novel general approach to the construction of 1-round δ-RMT and (0, δ)-SMT protocols using LD codes and MACs. The approach has a number of advantages, (i) it is general, unifies construction of 1-round δ-RMT and (0, δ)-SMT protocols, and is applicable to all connectivities including $N = 2t+K, K \geq 1$, where K is a constant, (ii) relies on well-studied mathematical objects (list decodable codes and MACs) and so allow a wide range of instantiations; this also allows direct translation of advances in those areas into better constructions for 1-round δ-RMT and $(0, \delta)$-SMT, and finally (iii) resulting in proofs of security (privacy and reliability) to be intuitive and easily verifiable. Instantiation of this general approach, using FRS-codes and our proposed multireceiver MACs result in constructions that have optimal transmission rates and the smallest δ, when $N = 2t + 1$ and $N = (2 + c)t, c > \frac{1}{t}$. For $N = 2t + 1$ the protocol is not computationally efficient for our instantiations. It is an interesting open problem if this general construction can be instantiated to achieve efficient and optimal construction for $N = 2t + 1$. Another interesting open question is whether δ can be further lowered while maintaining optimality. Another open problem is establishing lower bound on δ and constructing protocols that can achieve the bound. We note that 1-round δ-RMT, can be seen as error correcting code in the traditional setting where channel corruption is adversarial and adversary has a limited view of the codeword (only t component).

References

1. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness Theorems for Non-cryptographic Fault-tolerant Distributed Computation (extended abstract). In Proc. of STOC, pages: 1–10, 1988.
2. D. Chaum, C. Crepeau, and I. Damgard. Multiparty Unconditionally Secure Protocols (Extended Abstract). In Proc. of FOCS, pages: 11–19, 1988.
3. H. Chan, A. Perrig, and D. Song. Random Key Predistribution for Sensor Networks. In Proc. of the IEEE Symposium on Security and Privacy, pp. 197-213, 2003.
4. D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly Secure Message Transmission. In Journal of the ACM, 40(1):17–47, 1993.
5. Y. Desmedt, S. Erotokritou, and R. Safavi-Naini. Simple and Communication Complexity Efficient Almost Secure and Perfectly Secure Message Transmission Schemes. In Proc. of AFRICACRYPT, pages: 166–183, 2010.
6. Y. Desmedt, Y. Frankle and M. Yung. Multi-receiver/Multi-sender network security: efficient authenticated multicast/feedback. In IEEE Infocom, pages: 2045–2054, 1992.
7. P. Elias. List Decoding for Noisy Channels. Technical Report 335, MIT Research Lab of Electronics, 1957.
8. M. Fitzi, M. Franklin, J. Garay, and S. H. Vardhan. Towards Optimal and Efficient Perfectly Secure Message Transmission. In Proc. of TCC, LNCS 4392, pages: 311–322, 2007.
9. M. K. Franklin and R. N. Wright. Secure Communication in Minimal Connectivity Models. In Journal of Cryptology, 13(1):9–30, 2000.

10. V. Guruswami. Linear-algebraic List Decoding of Folded Reed-Solomon Codes. In CoRR abs/1106.0436, 2011.
11. V. Guruswami and A. Rudra. Explicit Codes Achieving List Decoding Capability: Error-Correction With Optimal Redundancy. In IEEE Transactions on Information Theory, Vol. 54, No.1, 2008.
12. V. Guruswami and M. Sudan. Improved Decoding of Reed-Solomon and Algebraic-Geometric Codes. In FOCS, pages: 28–39, 1998.
13. K. Kurosawa and K. Suzuki. Almost Secure (1-round, n-channel) Message Transmission Scheme. In Proc. of ICITS, volume 4883 of LNCS, pages: 99–112, 2009.
14. A. Patra, A. Choudhary, K. Srinathan, and C. Rangan. Unconditionally Reliable and Secure Message Transmission in Undirected Synchronous Networks: Possibility, Feasibility and Optimality. In IJACT 2(2):159–197, 2010.
15. T. Rabin and M. Ben-Or. Verifiable Secret Sharing and Multiparty Protocols with Honest Majority (Extended Abstract). In Proc. of STOC, pages: 73–85, 1989.
16. K. Srinathan. Secure Distributed Communication. PhD Thesis, IIT Madras, 2006.
17. M. Sudan. Decoding of Reed Solomon Codes beyond the Error-Correction Bound. In J. Complexity 13(1):180–193, 1997.
18. K. Srinathan, A. Patra, A. Choudhary, and C. Rangan. Probabilistic Perfectly Reliable and Secure Message Transmission Possibility, Feasibility and Optimality. In INDOCRYPT, pages: 101–122, 2007.
19. K. Srinathann, A. Choudhary, A. Patra, and C. Rangann. Efficient Single Phase Unconditionally Secure Message Transmission with Optimum Communication Complexity. In PODC, page: 457, 2008.
20. K. Srinathan, A. Narayanan, and C. P. Rangan. Optimal Perfectly Secure Message Transmission. In Proc. of CRYPTO, volume 3152 of LNCS, Springer, 2004.
21. R. Safavi-Naini, M. A. A. Tuhin, and H. Shi. Optimal Message Transmission Protocols with Flexible Parameters. In Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security (ASIACCS '11), pages: 453–458, 2011.
22. M. A. A. Tuhin and R. Safavi-Naini. Optimal One Round Almost Perfectly Secure Message Transmission In Proceedings of Financial Cryptography and Data Security (FC '11), volume 7035 of LNCS, pages: 173–181, 2011.
23. Y. Wang. Robust Key Establishment in Sensor Networks. In SIGMOD Record 33(1): 14–19, 2004.
24. J. Wu and Douglas R. Stinson. Three Improved Algorithms for Multi-path Key Establishment in Sensor Networks Using Protocols for Secure Message Transmission. In IEEE Transactions on Dependable and Secure Computing, pp. 929–937, 2011.

Appendix

A Decoding of Folded Reed-Solomon Codes

A.1 A Decoding Algorithm using Multivariate Interpolation

The list decoding algorithm for RS-codes, given in [17, 12], uses bivariate interpolation and recovers all codewords that have agreement in T positions, with the received codeword. The decoding algorithm solves the following problem:

Input: A finite field $\mathbf{F_q}$*,* n *distinct pairs of elements* $\{x_i, y_i\}_{i=1}^{n}$ *from* $\mathbf{F_q} \times \mathbf{F_q}$*, and integers* k *and* T*.*

Output: A list of all functions $f : \mathbf{F_q}^k \to \mathbf{F_q}$ *satisfying:*

$f(x)$ *is a polynomial in* x *of degree at most* $k-1$ *with* $|\{i|f(x_i) = y_i\}| \geq T$.

Using bivariate interpolation followed by a root finding step, all codewords that have agreements in T positions with the received word, are found. The algorithm is improved [11] by using multivariate interpolation of polynomial. This improves the list decoding radius from $1 - \sqrt{R}$ to approach $1 - R - \epsilon$.

In the following we give an outline of the $(s+1)$-variate list decoding algorithm, where s is an integer satisfying $1 \leq s \leq u$. We begin with some basic definitions that will be used for multivariate polynomials.

Definition 6. *[11]: For a polynomial* $Q(X, Y_1, Y_2, \cdots, Y_s) \in \mathbf{F_q}[X, Y_1, Y_2, \cdots, Y_s]$, *the* $(1, k, k, \cdots, k)$*-weighted degree is defined as the maximum value of* $l + kj_1 + kj_2 + \cdots + kj_s$ *taken over all monomials* $X^l Y_1^{j_1} Y_2^{j_2} \cdots Y_s^{j_s}$ *that occur with a nonzero coefficient in* $Q(X, Y_1, Y_2, \cdots, Y_s)$.

Definition 7. *[11]:A polynomial* $Q(X, Y_1, Y_2, \cdots, Y_s)$ *over* $\mathbf{F_q}$ *is said to have a zero of multiplicity* $r \geq 1$ *at a point* $(\alpha, \beta_1, \beta_2, \cdots, \beta_s) \in \mathbf{F_q}^{s+1}$ *if* $Q(X + \alpha, Y_1 + \beta_1, Y_2 + \beta_2, \cdots, Y_s + \beta_s)$ *has no monomial of degree less than* r *with a nonzero coefficient. (The degree of the monomial* $X^i Y_1^{j_1} Y_2^{j_2} \cdots Y_s^{j_s}$ *equals* $i + j_1 + j_2 + \cdots + j_s$*).*

Let $Y \in (\mathbf{F_q}^u)^N$ be the received word of a FRS-code with at least T correct positions. Define the set of interpolation points I to be integers in $\{0, 1, \cdots, n-1\}$ except those in the set:

$$\bigcup_{j=0}^{\frac{n}{u}-1} \{ju + u - s + 1, ju + u - s + 2, \cdots, ju + u - 1\}.$$

If the $u-$tuple containing the y_i is correct and $i \in I$, then all the s values $y_i, y_{i+1}, \cdots, y_{i+s-1}$ are correct. We define $n_0 = |I|$ and note that $n_0 = \frac{(u-s+1)n}{u} = N(u - s + 1)$.

Lemma 2. *(Lemma 3.3, [11]): Let* $\{\alpha_i, y_{i_1}, y_{i_2}, \cdots, y_{i_s}\}$ *for* $1 \leq i \leq n_0$ *be vectors where* α_i *is the support of codeword in position* i *and vector* $(y_{i_1}, y_{i_2}, \cdots, y_{i_s})$ *is the received element in position* $i, i+N, \cdots, i+(s-1)N$. *Let* $Q(X, Y_1, Y_2, \cdots, Y_s) \in F[X, Y_1, Y_2, \cdots, Y_s]$ *be a nonzero polynomial of* $(1, k, k, \cdots, k)$*-weighted degree at most* D *that has a zero of multiplicity* r *at* $\{\alpha_i, y_{i_1}, y_{i_2}, \cdots, y_{i_s}\}$ *for every* i, $1 \leq i \leq n_0$. *Let* $f(X), g_1(X), \cdots, g_{s-1}(X)$ *be polynomials of degree at most* $k-1$ *such that for at least* $T > D/r$ *columns, we have* $f(\alpha_i) = y_{i_1}$ *and* $g_1(\alpha_i) = y_{i_2}, \cdots, g_{s-1}(\alpha_i) = y_{i_{s-1}}$. *Then,* $Q(X, f(X), g_1(X), \cdots, g_{s-1}(X)) \equiv 0$.

Here the polynomial $g_i(x)$ is given by $g_i(x) = f(\gamma^i x)$. The $(s+1)$-variate list decoding algorithm consists of two steps:

- Find a polynomial $Q(X, Y_1, \cdots, Y_s)$ with $(1, k, k, \cdots, k)$, with weighted degree at most D, that satisfies $Q(\alpha_i, y_{i_1}, y_{i_2}, \cdots, y_{i_s}) = 0$, for $1 \leq i \leq n_0$.
- Factor the polynomial into irreducible factors. Output all the polynomials f such that $y - f(x)$ is a factor of Q and $f(\alpha_i) = y_{i_1}, f(\gamma\alpha_i) = y_{i_2}, \cdots, g_{s-1}(\gamma^{s-1}\alpha_i) = y_{i_s}$ for at least T columns of the received codeword.

The following lemma gives the conditions that must be satisfied by the system parameters for the decoding to succeed.

Lemma 3. *1) Provided* $\frac{D^{s+1}}{(s+1)!k^s} > n_0\binom{r+s}{s+1}$*, a nonzero polynomial* $Q(X, Y_1, \cdots, Y_s)$ *with the above stated properties exists and moreover can be found in time polynomial in* n *and* r^s*.*

2) Let T *be an integer such that* $T > \frac{D}{(u-s+1)r}$*. Then every polynomial* $f(x) \in F[x]$ *of degree at most* $k-1$ *agrees with the received word* z *on at least* T *locations and satisfies* $Q(x, f(x), f(\gamma x), \cdots, f(\gamma^{s-1}x)) \equiv 0$*.*

Lemma 4. *: For every integer* u *and* s*, the* $(s+1)-$*variate FRS decoder successfully list decodes the* $u-$*folded Reed-Solomon code up to a radius* $N - T$ *as long as the agreement parameter* T *satisfies:*

$$T \geq \sqrt[s+1]{(N\frac{k}{u-s+1})^s \prod_{j=1}^{s}(1+\frac{j}{r})},$$

where T *stands for the number of correct positions. The algorithm runs in* $(q^{\mathcal{O}(s)})$ *times and outputs a list of size at most* $|\mathbf{F_q}|^{s-1} = q^{s-1}$*.*

A.2 Linear-Algebraic Decoding

Lemma 5. *A nonzero* $Q \in \mathbf{F_q}[X, Y_1, \cdots, Y_s]$ *of above form can be found by solving a homogeneous linear system over* $\mathbf{F_q}$ *if the number of monomials in* Q *larger than the number of interpolation conditions*

$$(D+1)s + D + k > N(u-s+1),$$

where $deg(A_i) \leq D$ *for* $i = 1, 2, \cdots, s$ *and* $deg(A_0) \leq D + k - 1$*.*

Lemma 6. *If* $f(X)$ *is a polynomial of degree at most* $k-1$ *whose FRS encoding agrees with the received word* y *in at least* T *columns for* $T\frac{D+k-1}{u-s+1}$*, then*

$$Q(X, f(X), f(\gamma X), \cdots, f(\gamma^{s-1}X)) = 0.$$

Lemma 7. *If the order of* γ *is at least* k*, the affine space of solutions to* $Q(X, f(X), f(\gamma X), \cdots, f(\gamma^{s-1}X)) = 0$ *has dimension* d *at most* $s-1$*. Further, one can compute using* $\mathcal{O}((Nm)^2)$ *field operations over* $\mathbf{F_q}$ *a matrix* $M \in \mathbf{F_q}^{k\times d}$ *and a vector* $z \in \mathbf{F_q}^k$ *such that the solutions are contained in the affine space* $Mx + z$ *for* $x \in \mathbf{F_q}^d$ *. Also, the matrix* M *can be assumed to have the* $d \times d$ *identity matrix as a submatrix.*

B Proofs

B.1 Proof of Theorem 1

Proof. (outline) Suppose t receivers want to cheat an honest receiver i. Colluders want to forge a message and tag pair, $m' = (m'_1, m'_2, \cdots, m'_d, A'(z))$ where $(m'_1, m'_2, \cdots, m'_d) \neq (m_1, m_2, \cdots, m_d)$ that passes the verification algorithm of the honest user. The colluders know their t keys, but do not know the secret key of user i, given by $P_1(z_i), P_2(z_i), \cdots, P_d(z_i), P_{d+1}(z_i)$. The known message and tag pair is given by,

$$A(z_i) = P_1(z_i)m_1 + P_2(z_i)m_2 + \cdots + P_d(z_i)m_d + P_{d+1}(z_i) \mod q^{s'},$$

and the forgery $m', A'(z)$, must pass the receiver i verification and so should satisfy the equation:

$$A'(z_i) = P_1(z_i)m'_1 + P_2(z_i)m'_2 + \cdots + P_d(z_i)m'_d + P_{d+1}(z_i) \mod q^{s'}.$$

It means the secret authentication key of receiver i satisfies,

$$\Delta A(z_i) = P_1(z_i)\Delta m_1 + P_2(z_i)\Delta m_2 + \cdots + P_d(z_i)\Delta m_d \mod q^{s'}.$$

There are $q^{s'(d-1)}$ choices for receiver i's secret key $P_1(z_i), P_2(z_i), \cdots, P_d(z_i)$. On the other hand, $P_1(z_i), P_2(z_i), \cdots, P_d(z_i)$ is indepndent from the adversary's view because she sees at most t points of the polynomials $P_1(z_j), P_2(z_j), \cdots, P_d(z_j)$ where j is a corrupted receiver, and the values $P_1(z_i), P_2(z_i), \cdots, P_d(z_i)$ are blinded by the random (indeonent from adversary's view) values $P_{d+1}(z_i)$. Therefore the probability that $P_1(z_i), P_2(z_i), \cdots, P_d(z_i)$ satisfy the above equation is $1/q^{s'}$. This means that the success probability of the colluders in constructing $m', A'(x)$ that passes the honest receiver verifiication test is not better than $1/q^{s'}$. □

B.2 Proof of Theorem 2

Proof. Using an argument similar to the proof of theorem 1, the probability that a forged message $m', A'(z)$ satisfy an honest receiver's verification algorithm is the probability of satisfying:

$$\begin{aligned} MAC(m', r) &= m'_1P_1(z_i) + \cdots + m'_tP_t(z_i) + m'_{t+1}P_1(z_i)^2 + \cdots + m'_{2t}P_t(z_i)^2 \\ &+ m'_{2t+1}P_1(z_i)P_2(z_i) + \cdots + m'_{\binom{t+2}{2}-1}P_{t-1}(z_i)P_t(z_i) + P_{t+1}(z_i) \\ &= A'(z_i), \end{aligned}$$

which means $P_1(z_i), P_2(z_i), \cdots, P_t(z_i)$ must satisfy,

$$\begin{aligned} &\Delta m_1P_1(z_i) + \cdots + \Delta m_tP_t(z_i) + \Delta m_{t+1}P_1(z_i)^2 + \cdots + \Delta m_{2t}P_t(z_i)^2 \\ &\quad + \Delta m_{2t+1}P_1(z_i)P_2(z_i) + \cdots + \Delta m_{\binom{t+2}{2}-1}P_{t-1}(z_i)P_t(z_i) = \Delta A(z_i). \end{aligned}$$

There are in total $2q^{s'(t-1)}$ of $(P_1(z_i), P_2(z_i), \cdots, P_t(z_i))$ vectors satisfying the above. Because the number of unknown values in $(P_1(z_i), P_2(z_i), \cdots, P_t(z_i))$ is $q^{ts'}$, the adversary's success probability is at most $2/q^{s'}$. □

B.3 Proof of Theorem 4

Proof. : First we need to show that the received codeword is decodable. According to the linear interpolation decoding algorithm, the decoding condition is satisfied if we choose parameter $s = (N/\sigma) - 1$, $u \gg s$ and $s' = \left\lfloor \frac{2u(t+1-2\sigma)}{t^2+5t+2} \right\rfloor$. This is because,

$$
\begin{aligned}
t+1 &\geq N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}) \\
t+1 &\geq N\frac{1}{s+1} + \frac{s}{s+1}\frac{k}{u-s+1} \\
t+1 &\geq \sigma + \frac{(t^2+5t+2)s'}{2u} \\
t+1 &> \sigma + t + 1 - 2\sigma. \qquad (5)
\end{aligned}
$$

To find δ for RMT, we note that using the same MAC function as in Section 2.1, the probability that another message, tag pair (m', x') with s $m' \neq m$, pass the verification test is less than $2/q^{s'}$. The size of the list of decoded messages is at most q^{s-1}. Therefore the probability that another messages, tag pair in the list pass at least one uncorrupted wire verifiication is at most $\frac{2}{q^{s'-s+1}}$. Because there are total $t+1$ uncorrupted wires, the reliability is at least $1 - \frac{t+1}{q^{s'-s+1}}$.

Finally the transmission rate is optimal as,

$$\frac{uN + (s't + s')N}{(\binom{t+2}{2} - 1)s'} = \mathcal{O}(1).$$

B.4 Proof of Theorem 5

Proof. (outline)

Perfect Privacy: The adversary knows t components of the FRS codeword, each consisting of u components of the underlying RS-code. The dimension of the FRS-code is $ut + \sigma u + ts' + s'$. This leaves $\sigma u + ts' + s'$ elements (coefficients of the polynomial that is associated with the underlying RS codeword), that are independent of the adversary's view. We note that only σu elements forms the information block of $m_{\mathcal{S}}$ and the remaining part is the verification information.

This gives in total ${\mathbf{F_q}}^{\sigma u}$ possible codewords for correct messages and so the adversary will be completely uncertain about the information block, $(m_0, m_1, \cdots, m_{\sigma u})$.

δ-Reliability:

The adversary controls t wires and so t positions (u components each) of FRS-code could be changed. Without loss of generality assume the first t wires are corrupted by the adversary and so the last $t+1$ wires are private.

To break the reliability of the protocol the adversary needs to be able to change the values sent over the corrupted wires, such that the list of codewords resulting from the list decoding step, contains a codeword that encodes a message $m'_{\mathcal{S}}$ for which at least $t+1$ verification equations are satisfied. This will result in more than one message passing the verification test of the protocol and so, the protocol outputs **Fail**.

Note that the adversary controls the verification keys of the t corrupted wires. We assume a powerful adversary (it is unclear how this adversary can be constructed) that can change the t wires such that the verification tests of those t wires successfully pass for a message $x' = m'_{\mathcal{S}}$. Since the adversary does not know the verification keys of wires $t+1, t+2, \cdots, N$, her best success chance in forging one of these values is,

$$\begin{aligned}&\Pr[(MAC(x', k_{t+1}) = A'(z_{t+1})) \vee \cdots \vee (MAC(x', k_{2t+1}) = A'(z_{2t+1}))] \leq \\ &\Pr[MAC(x', k_{t+1}) = A'(z_{t+1})] + \cdots + \Pr[MAC(x', k_{2t+1}) = A'(z_{2t+1})] \\ &= \frac{(t+1)}{q^{s'}}. \end{aligned} \tag{6}$$

The size of the decoded list is at most q^{s-1}. The probability that any other first $s'd$ element vector which is different from the correct one pass through the verification is $\frac{(t+1)}{q^{s'}} \times q^{s-1}$ and the probability is obtained by replacing the value of $s' = \left\lfloor \frac{u(1-3\sigma)}{t+1} \right\rfloor$. □

B.5 Proof of Theorem 6

Proof. Perfect Privacy:

The adversary controls t wires and so knows ut positions of the RS-codeword. The dimension of the FRS-code (and the RS-code) is $ut + (\binom{t+2}{2})s_0 + (t+1)s_0$ and so the first ut elements are indepndent of the $(\binom{t+2}{2})s_0 + (t+1)s_0$ and there are in total $\mathbf{F_q}^{\binom{t+2}{2}s_0}$ possible codewords that have the same ut positions. This means that the value $m_0, m_1, \cdots, m_{\binom{t+2}{2}s_0-1}$ cannot be guessed from the code components sent over the corrupted wires. These values are also independent from random key values that are sent on the t corrupted wires.

δ-Reliability:

Lemma 8. *The correct first $\binom{t+2}{2}s_0$ elements vector that passes through the authentication must pass through at least $t + ct$ of the authentication test. Any other first $\binom{t+2}{2}s_0$ elements of messages, in the decoded list, which is different from the correct one failed to be checked with probability at most $\frac{2(t+1)}{q^{s_0}}$.*

Theorem 7. *The reliability of 1-round SMT for $N = 2t + ct, c > 1/t$ using list decoding is at most $1 - \frac{2(t+1)}{q}$.*

According to the adversary's capability, he can change any t lines so that ut positions of FRS-code are changed. The MAC function is over $\mathbf{F_q}^{s_0}$.

Transmission rate:

The total number of elements that are transmitted is $uN + (2t+ct)(t+1)s_0$. The transmission rate is $(2t + ct)c_0t + (2t + ct)(t + 1)s_0/\binom{t+2}{2}s_0$ which is $\mathcal{O}(1)$. Therefore our 1-round $(0, \delta)$-SMT for $N = 2t + ct$ is optimal.

Computation Complexity:

The decoding needs $\mathcal{O}((Nu \log q)^2)$ computation. The authentication needs $\mathcal{O}(q^{s_0})$ computation. Therefore the total time is $\mathcal{O}(q^{s_0})$, which is efficient. □

C Detail Calculations and Comparisons

C.1 Details of 1-round (0, δ)-SMT for $N = 2t + 1$

The required dimension for the code is,

$$k = |m_{\mathcal{S}}| = ut + \sigma u + (t+1)s'. \tag{7}$$

For a code of length $N = 2t + 1$ and dimension k as (10), we must choose folding parameter u, number of decoding variable s, and the finite field sizes s' and q, to ensure that decoding succeeds for linear-algebraic decoding (outlined in Section 2.1) for radius $\rho = t/N$ (t errors in the FRS-code). That is, the inequality (11) below, is satisfied.

$$t + 1 \geq N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}). \tag{8}$$

Let $0 < \sigma < 1/3$ and set $s = \frac{N}{\sigma} - 1$. Furthermore, choose $s' = \left\lfloor \frac{u(1-3\sigma)}{t+1} \right\rfloor$, and $u > s^2 - 1$. The following shows that with these choices of parameters inequality (11) is satisfied. We have,

$$t + 1 > \sigma + (t + \sigma + 1 - 3\sigma) \overset{(d)}{=} \sigma + \frac{ut + \sigma u + s'(t+1)}{u} \overset{(e)}{=} \sigma + \frac{k}{u}, \tag{9}$$

where (d) is because of the choice of s' and (e) is because of the value of k in (10). Note that because $s \geq 1$, we have $\frac{1}{u} > \frac{s}{s+1}\frac{1}{u-s+1}$ and so, (12) gives the following:

$$t + 1 > \sigma + \frac{k}{u} > \sigma + \frac{s}{s+1}\frac{k}{u-s+1} \overset{(f)}{=} N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}),$$

where (f) is by using the value of s and because the rate of the code is $R = \frac{k}{n} = \frac{k}{Nu}$. Finally we can choose q to be the smallest prime that is bigger than the codeword length $n = Nu$.

C.2 Details of 1-round (0, δ)-SMT for $N = (2 + c)t$

According to Lemma 1,

$$\begin{aligned}
t + ct &\geq N(\frac{1}{s+1} + \frac{s}{s+1}\frac{uR}{u-s+1}) \\
t + ct &\geq (2t + ct)\frac{1}{s+1} + \frac{s}{s+1}\frac{k}{u-s+1} && (10) \\
(t + ct)(1 - \frac{1}{s+1}) &\geq \frac{t}{s+1} + \frac{s}{s+1}\frac{ut + \binom{t+2}{2}s + (t+1)s}{u-s+1} && (11) \\
\frac{s}{s+1}ct &> \frac{t}{s+1} - \frac{ts}{s+1} + \frac{s}{s+1}\frac{c_0t^2 + \binom{t+2}{2}s + (t+1)s}{c_0t - s + 1} && (12) \\
c &> \frac{1}{s} - 1 + \frac{c_0t^2 + \binom{t+2}{2}s + (t+1)s}{c_0t^2 - st + t}.
\end{aligned}$$

$$(13)$$

In the above (13) is by replacing R with k/N, (14) is by replacing k by $ut + \binom{t+2}{2}s + (t+1)s$, and (15) is by replacing u with c_0t. If we choose constant value $s = s_0$ and $t > 7c_0 + 2s_0 + 3$, the value c that promises the receiver to apply FRS-code to the list of decoded messages is,

$$c > \frac{s_0}{c_0} + \frac{1}{s_0}.$$