On Round Complexity of Unconditionally Secure VSS

Arpita Patra **Ashish Choudhary*** **Ashwinkumar B.V** **C. Pandu Rangan**
Department of Computer Science and Engineering
Indian Institute of Technology Madras
Chennai India 600036
Email:{ arpita,ashishc,ashwin }@cse.iitm.ernet.in, rangan@iitm.ernet.in

Abstract

In this work, we initiate the study of round complexity of *unconditionally secure weak secret sharing* (UWSS) and *unconditionally secure verifiable secret sharing* (UVSS) [1] in the presence of an *all powerful* t-active adversary. Specifically, we show the following for UVSS: (a) 1-round UVSS is possible iff $t = 1$ and $n > 3$ (b) 2-round UVSS is possible if $n > 3t$ (c) 5-round UVSS is possible if $n > 2t$. For UWSS we show the following: (a) 1-round UWSS is possible iff $n > 3t$ and (b) 3-round UWSS is possible if $n > 2t$. Comparing our results with existing results for trade-off between fault tolerance and round complexity of perfect (zero error) VSS and WSS [8, 7, 9], we find that probabilistically relaxing the conditions of VSS/WSS helps to increase fault tolerance significantly.

Keywords: Verifiable Secret Sharing, Error Probability, Information Theoretic Security.

1 Introduction

In this paper, we initiate the study of round complexity of two important secure distributed computation tasks: *unconditionally secure verifiable secret sharing* (UVSS) and *unconditionally secure weak secret sharing scheme* (UWSS). Roughly speaking, in *secret sharing* [11], a dealer **D** wants to share a secret s among a set of n players, such that no set of t players can reconstruct s while any set of $t + 1$ or more players will be able to reconstruct s by pooling their shares. Verifiable secret sharing (VSS) [4] extends ordinary secret sharing to work against active corruption. It is a stronger notion than standard secret sharing and provides robustness against t malicious players, possibly including **D**. In UVSS [10], each property of VSS holds, but with a negligible error probability. The round complexity of interactive protocols is one of their most important complexity measure. Consequently, substantial research work has been done to study the round complexity of various distributed computation tasks, such as Byzantine agreement, secure message transmission, zero knowledge proofs, multiparty computation (MPC), VSS, etc. VSS and UVSS are important building blocks in the design of perfectly (zero error) secure MPC [2, 3] and unconditionally (negligible error) secure MPC [5, 10, 1] respectively. In addition, they also find application in Byzantine agreement, generating global coin toss, etc. So it is important to study the round complexity of VSS and UVSS. In [8], the authors have studied the interplay between the round complexity and fault tolerance of VSS protocols. However, nothing is known in the literature regarding the trade-off for UVSS protocols. In this paper, we initialize the study of the trade-off between the round complexity and fault tolerance of UVSS protocols.

*Work supported by Project No.SE/05-06/076/DITX/CPAN on Protocols for Secure Communication and Computation.
[1] In the literature, these problems are also called as statistical WSS and statistical VSS [8] respectively.

1.1 Network Model

We consider the standard *secure channel* settings, where there are n players $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$, who are pairwise connected by perfectly secure channels and a common broadcast channel is available to all the players. The broadcast channel allows a player to send some information identically to all the players. We assume **D** to be any one of the players from $\mathcal{P}$. Our protocols will *also* work for an external dealer where **D** is an entity outside the set $\mathcal{P}$. We assume the system to be synchronous. The protocol operates in a sequence of rounds, where in each round, a player performs some local computation, sends new messages to his neighbors through private channel and broadcasts some information over the broadcast channel, receives message sent by his neighbor in the previous round and receives message sent over broadcast channel in previous round, in that order. An adversary $\mathcal{A}_t$ with *unbounded computing power* can actively control at most t of the n players (possibly including **D**) during the protocol. Thus there exists $n - t$ honest players in the system. Active corruption means that $\mathcal{A}_t$ takes full control of the player and makes it (mis)behave in an arbitrary manner. The adversary is *centralized* and *adaptive* [5] and is allowed to *dynamically* corrupt players during protocol execution (and his choice may depend on the data seen so far). Moreover, the adversary is a *rushing adversary*, who in a particular round, first collects all the messages addressed to the corrupted players and exploits this information to decide on what the corrupted players send during the same round. The error probability is expressed in terms of an error parameter k. The protocols operate on a finite field $\mathbb{F} = GF(q)$, where $q = 2^k$ and n is polynomial in k. Thus, each element of $\mathbb{F}$ can be represented by k bits. S denotes the secret which **D** wants to share, where S is a sequence of $\ell \geq 1$ field elements, represented by $S = [s^1\ s^2\ \ldots\ s^\ell] \in \mathbb{F}^\ell$. Moreover, we assume ℓ to be polynomial in k.

1.2 Definitions and Terminologies

Definition 1 (UWSS with Agreement[5, 10]) *A (n, t)-UWSS scheme with agreement for sharing $S \in \mathbb{F}^\ell$ is a pair of protocols (***Sh, Rec***) that satisfy the following with error probability 2^{-k}:*

1. Termination: *If* **D** *is honest then all honest players will complete* **Sh** *and if the honest players invoke* **Rec***, then each honest player will complete* **Rec**.

2. Secrecy: *If* **D** *is honest and no honest player has yet started* **Rec***, then $\mathcal{A}_t$ has no information about S in information theoretic sense.*

3. *Once all currently uncorrupted players complete protocol* **Sh***, there exists a value $s^* \in \mathbb{F}^\ell \cup \{NULL\}$ such that the following requirements hold*

- Correctness: *If the dealer is uncorrupted throughout protocols* **Sh** *and* **Rec** *then $s^* = S$ and each honest player will output S at the end of* **Rec**.
- Weak Commitment: *If the dealer is corrupted, then all honest players output either s^* or 'NULL' upon completion of protocol* **Rec**.

Definition 2 (UVSS with Agreement [5, 10]) *A (n, t)-UVSS scheme with agreement for sharing $S \in \mathbb{F}^\ell$ is a pair of protocols (***Sh, Rec***) that satisfy the* termination, secrecy *and* correctness *property of UWSS and the following stronger commitment property: Once all currently uncorrupted players complete* **Sh***, there exists an $s^* \in \mathbb{F}^\ell$, such that following holds, with error probability 2^{-k}:*

- Strong Commitment: *Each honest player outputs s^* at the end of* **Rec**.

Definition 3 (UVSS without Agreement [6]) *UVSS without agreement is slightly weaker than UVSS with agreement in the sense that the honest players may not agree in their output at the end of protocol*

Rec; *i.e., some honest player(s) may output s^* while some other honest player(s) may output "NULL", where the later may happen with probability at most 2^{-k}.*

Similarly we can define UWSS without agreement. VSS [8]/WSS [7] satisfies all the properties of UVSS/UWSS (with agreement), except that there is no error probability allowed. We now define information checking protocol (IC), which is an important building block for designing UWSS and UVSS protocols.

Information Checking (IC) and IC Signatures [5, 10]: IC is an information theoretically secure method for authenticating data and is used to generate IC signatures. When a player $INT \in \mathcal{P}$ receives an IC signature from $\mathbf{D} \in \mathcal{P}$, then INT can later produce the signature and have the players in $\mathcal{P}$ verify that it is a valid signature. An IC scheme consists of a sequence of three protocols:

1. **Distr**$(\mathbf{D}, INT, \mathcal{P}, S)$ is initiated by the dealer $\mathbf{D}$, who hands secret $S \in \mathbb{F}^\ell$ to *intermediary* INT. In addition, $\mathbf{D}$ hands some **authentication information** to INT and **verification information** to individual players in $\mathcal{P}$, also called as *receivers*.

2. **AuthVal** $(\mathbf{D}, INT, \mathcal{P}, S)$ is initiated by INT to ensure that in protocol **RevealVal**, secret S held by INT will be accepted/will be considered as valid.

3. **RevealVal** $(\mathbf{D}, INT, \mathcal{P}, S)$ where INT produces S, along with **authentication information** and the individual receivers in $\mathcal{P}$ produce **verification information**. Depending upon the values produced by INT and the receivers, either S is accepted or rejected.

The **authentication information**, along with S, which is held by INT at the end of **AuthVal** is called $\mathbf{D}$'s IC signature on S, obtained by INT. The IC signature must satisfy the following properties:

1. If $\mathbf{D}$ and INT are uncorrupted, then S will be accepted in **RevealVal**.

2. If INT is uncorrupted, then at the end of **AuthVal**, INT knows an S, which will be accepted in **RevealVal**, except with probability 2^{-k}.

3. If $\mathbf{D}$ is uncorrupted, then during **RevealVal**, with probability at least $1-2^{-k}$, every $S' \neq S$ produced by a corrupted INT will be rejected.

4. If $\mathbf{D}$ and INT are uncorrupted, then at the end of **AuthVal**, S is information theoretically secure.

Remark 1 *Following [8], our main aim in this paper is to study the round complexity of UWSS and UVSS problems as a stand alone application. There is an alternative definition of UVSS, which is suitable to use in the context of general unconditionally secure MPC (see [1] for details). However, such a strong definition is not required when we want to study the round complexity of UVSS as a stand alone application. We stress that our 5-round UVSS protocol with $n = 2t + 1$ satisfies this alternative definition given in [1] and can be applied for unconditionally secure MPC.*

Definition 4 (Unconditionally Secure Hashing) *Let $R = [r_1 \ r_2 \ \ldots \ r_\ell] \in \mathbb{F}^\ell$ be a vector and $\alpha \in \mathbb{F} - \{0\}$. Then we define $v = USHash(\alpha; R)$ as the* **hash value** *of R with respect to* **hash key** *α, where $v = r_1 + r_2\alpha + r_3\alpha^2 + \ldots + r_\ell\alpha^{\ell-1}$. It is easy to see that two different ℓ length vector may have same hash value at a common hash key α with probability at most $\frac{\ell-1}{|\mathbb{F}|}$, which is at most $2^{-O(k)}$ in our context. Moreover, if R is uniformly selected at random from $\mathbb{F}$, such that $\mathcal{A}_t$ does not know R but knows at most t hash values corresponding to t different hash keys, then $(\ell - t)$ elements of R will be information theoretically secure (provided $(\ell - t) > 0$).*

Round Complexity of VSS and WSS [8, 7, 9]: Any VSS (WSS) protocol consists of two phases: sharing phase and reconstruction phase. The sharing phase may consist of several rounds. In the

reconstruction phase, player P_i produces his entire view v_i generated during sharing phase and a reconstruction function $\mathbf{Rec}(v_1, v_2, \ldots, v_n)$ is applied to obtain protocol's output. In [8], round complexity of any VSS/WSS protocol is defined as the number of rounds in its sharing phase. The reconstruction phase of any VSS/WSS protocol can be done in a single round [8, 7, 9].

Round Complexity of UVSS, UWSS and IC: As in the case of VSS and WSS, we define the round complexity of any UVSS/UWSS protocol as the number of rounds in its sharing phase. We define the round complexity of IC protocol as the number of rounds in **Distr** and **AuthVal** protocol. Unlike VSS/WSS, the reconstruction phase of UVSS/UWSS protocol cannot be done in a single round, if $\mathcal{A}_t$ is rushing. In [6], it is shown that in any UVSS (also UWSS) protocol (irrespective of the number of rounds in sharing phase) designed with $n = 2t + 1$, reconstruction cannot be done in a single round if $\mathcal{A}_t$ is rushing. However, it is also shown that reconstruction can be done in a single round, but in that case, the resultant UVSS will be an UVSS without agreement (see Definition 3). Moreover, it is also shown any UVSS/UWSS scheme without agreement can be converted into an UVSS/UWSS scheme with agreement by adding one extra round in the reconstruction phase. Now by using similar argument as in [6], we can show that if $\mathcal{A}_t$ is rushing, then it is impossible to design the following, with only one round in the reconstruction phase: (a) single round $(4t, t)$ UWSS with agreement; (b) single round $(4, 1)$ UVSS with agreement; (c) two round $(4t, t)$ UVSS with agreement. We do not provide formal proof due to space constraint. All the UVSS (UWSS) protocols in this paper are designed assuming $\mathcal{A}_t$ is rushing and satisfies Definition 2 (1). So our protocols have two rounds in their reconstruction phase and we compare our UVSS/UWSS protocols with the existing VSS/WSS protocols based on the round complexity of *only* sharing phase. Note that if $\mathcal{A}_t$ is non-rushing, then the reconstruction phase of all our UVSS/UWSS protocols can be achieved in a single round. Similarly, if $\mathcal{A}_t$ is rushing, then the **RevealVal** of **IC** protocol will take two rounds, otherwise it can be done in a single round.

Following the approach of [8], in our protocols, we assume that if **D** is discarded during sharing phase, then a pre-determined $\hat{S} \in \mathbb{F}^\ell$ will be taken as **D**'s secret. Also, if P_i expects to receive some value from P_j and either no value or some syntactically incorrect value arrives from P_j, then P_i replaces the received value by some fixed default syntactically correct value.

1.3 Existing Literature on Round Complexity of VSS and WSS

The existing tradeoff between round complexity and fault tolerance of VSS and WSS is given in Table 1. Recently, Katz et.al [9] have studied the design of VSS protocols, with the aim of reducing the number

# Rounds	WSS	VSS
1	$n > 4t$ [7]	$t = 1,\ n > 4$; impossible if $t > 1$ [8]
2	$n > 4t$ [7]	$n > 4t$ [8]
3	$n > 3t$ [7]	$n > 3t$ [8]; sufficiency shown by inefficient protocol; efficient protocol given in [7]
4	$n > 3t$ [7]	$n > 3t$ [8]

Table 1: Existing tradeoff between round complexity and fault tolerance of VSS and WSS

of rounds in which broadcast is used. Specifically, they designed a three round VSS with $n = 3t + 1$ (which is the optimum value of n for any VSS [2]), where the broadcast channel is used only in one round during sharing phase. Using similar techniques, our protocols can be easily modified so that broadcast channel is used in only one round of sharing phase.

1.4 Existing Literature on UVSS and UWSS

The notion of UVSS, UWSS and IC were first introduced by Rabin et.al in [10]. In [10], it is shown that UVSS (UWSS) is possible iff $n \geq 2t + 1$. The UVSS, UWSS and IC protocol of [10] were significantly improved by Cramer et.al [5]. In [6], Cramer et.al have proved the lower bound on the communication complexity of reconstruction phase of any UVSS protocol (without agreement) designed with $n = 2t+1$, as stated in the following theorem:

Theorem 1 ([6]) *Let $n = 2t+1$ and $\mathbf{D}$ be honest. Under these settings, in any UVSS protocol without agreement, with single round of reconstruction, the total number of bits broadcasted during reconstruction phase is $\Omega(nH(S) + kn^2)$, where S is the secret and $H(S)$ is its entropy. Moreover the bound is tight.*

1.5 Our Contribution and Its Significance

Our contributions in this paper can be summarized as follows:

1. We initiate the study of the trade-off between round complexity and fault tolerance of UVSS and UWSS protocols. Specifically, for UVSS, we show the following: (a) 1-round UVSS is possible iff $t = 1$ and $n \geq 4$. Moreover, if $t > 1$, then no 1-round UVSS is possible irrespective of the value of n; (b) 2-round UVSS is possible if $n > 3t$; (c) 5-round UVSS is possible if $n > 2t$. For UWSS, we show the following: (a) 1-round UWSS is possible iff $n > 3t$; (b) 3-round UWSS is possible if $n > 2t$. Comparing these results with the results of Table 1, we find that probabilistically relaxing the conditions of VSS/WSS significantly helps in improving the fault tolerance.

2. In [6], the authors have given a single round honest dealer UVSS scheme without agreement for $n = 2t + 1$, with single round of reconstruction, which shares k bits (i.e., $\ell = 1$ element from $\mathbb{F}$) by broadcasting $O(kn^2)$ bits ($O(n^2)$ elements from $\mathbb{F}$) during reconstruction phase. Though the scheme satisfies the lower bound of Theorem 1 for $\ell = 1$, it will not satisfy the lower bound for general ℓ. However, our single round $(3t + 1, t)$ UWSS protocol can be converted into a single round honest dealer $(2t + 1, t)$ UVSS without agreement, for any $\ell \geq 1$, with single round of reconstruction, whose communication complexity satisfies Theorem 1.

3. In [5], the authors have given a four round IC protocol, which signs an $\ell = 1$ length secret with a communication overhead (both private and broadcast) of $O(n)$ field elements. So to generate an IC signature on $\ell > 1$ length secret, the protocol needs to be parallely executed ℓ times, resulting in a communication overhead of $O(n\ell k)$ bits. However, in this paper, we design a 3-round IC protocol, which signs on $\ell \geq 1$ length secret by communicating and broadcasting $O((\ell + n)k)$ bits. If ℓ is not constant, then our protocol significantly improves the communication and round complexity of the IC protocol given in [5].

4. In [5], the authors have given a $(2t + 1, t)$ UVSS protocol, whose sharing phase takes at most eleven rounds. Moreover, the protocol shares a single length secret; i.e., $\ell = 1$ by communicating and broadcasting $O(n^3k)$ bits. The protocol needs to be executed ℓ times to share ℓ length secret, incurring a communication overhead (both private and broadcast) of $O(n^3\ell k)$ bits. However, by using our efficient IC protocol, we design a five round UVSS protocol which shares an $\ell \geq 1$ length secret by communicating and broadcasting $O(n^2(\ell + n)k)$ bits, thus significantly improving the communication and round complexity of the UVSS protocol of [5].

5. By using similar techniques from [6], we can modify our five round $(2t+1, t)$ UVSS protocol, such that its reconstruction phase satisfies Theorem 1. Then by using our modified UVSS protocol, we can reduce the communication complexity of pre-processing step of the MPC protocol of [6] from

$O(n^5k)$ bits to $O(n^3k)$ bits per multiplication. We do not give the details in this paper and hope to address these issues in our future work.

6. The existing UVSS/UWSS/IC protocol(s) are designed to handle single length secret; i.e., $\ell = 1$. So in applications, where we need to handle $\ell > 1$ length secret, such as multiparty set intersection problem (and many other specific MPC problems), these protocols need to be parallely executed ℓ times, which results in significant communication overhead. However, all our protocols are designed for a general $\ell \geq 1$ length secret and can efficiently handle $\ell > 1$ length secret. Our protocols are much more efficient than ℓ invocations of the protocol on single field element.

2 Secret Distribution Protocol

We now design a single round protocol **Secret Distribution** in Table 2 with $n \geq 2t + 1$, which allows **D** to share $S \in \mathbb{F}^\ell$. The protocol is used as a black-box in our UWSS and UVSS protocols.

• **D** selects a random non-zero polynomial $M(x)$ over $\mathbb{F}$ of degree $\ell + t - 1$, such that the lower order ℓ coefficients of $M(x)$ are elements of S. **D** then computes $M(1), M(2), \ldots, M(\ell + t)$.
• **D** selects $\ell + t$ random polynomials $f_1(x), f_2(x), \ldots, f_{\ell+t}(x)$, each of degree t, such that for $1 \leq i \leq \ell + t$, $f_i(0) = M(i)$. **D** evaluates each $f_i(x)$ at $x = 1, 2, \ldots, n$. Let F_i denotes the vector $[f_1(i)\ f_2(i)\ \ldots\ f_{\ell+t}(i)]$ and T denotes the $n \times (\ell + t)$ matrix, whose i^{th} row is F_i. **D** also selects n random non-zero hash keys from $\mathbb{F}$, denoted by $\alpha_1, \alpha_2, \ldots, \alpha_n$.
• To P_i, **D** gives the vector F_i, the hash key α_i and the n tuple $[v_{1i}\ v_{2i}\ \ldots\ v_{ni}]$ where for $1 \leq j \leq n$, $v_{ji} = USHash(\alpha_i; F_j)$.

Table 2: **Protocol Secret Distribution**: Single Round Secret Distribution Protocol

Before proving the properties of protocol **Secret Distribution**, we first pictorially represent the values computed by **D**.

$M(x)$, lower order ℓ coefficients are elements of S					
$M(1)$	$M(2)$	$\ldots$	$M(j)$	$\ldots$	$M(\ell + t)$
$f_1(x)$	$f_2(x)$	$\ldots$	$f_j(x)$	$\ldots$	$f_{\ell+t}(x)$
$f_1(0) = M(1)$	$f_2(0) = M(2)$	$\ldots$	$f_j(0) = M(j)$	$\ldots$	$f_{\ell+t}(0) = M(\ell + t)$
$f_1(1)$	$f_2(1)$	$\ldots$	$f_j(1)$	$\ldots$	$f_{\ell+t}(1)$
$f_1(2)$	$f_2(2)$	$\ldots$	$f_j(2)$	$\ldots$	$f_{\ell+t}(2)$
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
$f_1(i)$	$f_2(i)$	$\ldots$	$f_j(i)$	$\ldots$	$f_{\ell+t}(i)$
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
$f_1(n)$	$f_2(n)$	$\ldots$	$f_j(n)$	$\ldots$	$f_{\ell+t}(n)$

$$
\begin{array}{rcl}
F_1 & = & [f_1(1)\ f_2(1)\ f_3(1)\ \ldots\ f_j(1)\ \ldots\ f_{\ell+t}(1)] \\
F_2 & = & [f_1(2)\ f_2(2)\ f_3(2)\ \ldots\ f_j(2)\ \ldots\ f_{\ell+t}(2)] \\
\ldots & \ldots & \ldots \\
F_i & = & [f_1(i)\ f_2(i)\ f_3(i)\ \ldots\ f_j(i)\ \ldots\ f_{\ell+t}(i)] \\
F_n & = & [f_1(n)\ f_2(n)\ f_3(n)\ \ldots\ f_j(n)\ \ldots\ f_{\ell+t}(n)]
\end{array}
$$

Now we prove the properties of protocol **Secret Distribution**.

Lemma 1 *In protocol* **Secret Distribution**, *any $t + 1$ players can jointly reconstruct S.*

PROOF: The proof follows from the fact that any $t+1$ players will know $t+1$ points on each $f_i(x)$, from the F vectors given to them. Since each $f_i(x)$ is of degree t, the knowledge of $t+1$ points is sufficient to construct each $f_i(x)$ and hence each $M(i)$. Now using the $M(i)$'s, $M(x)$ and hence S can be reconstructed. □

Lemma 2 *If* $\mathbf{D}$ *is honest then* S *is information theoretic secure against* $\mathcal{A}_t$.

PROOF: If $\mathcal{A}_t$ comes to know about $t+1$ F vectors, then from Lemma 1, $\mathcal{A}_t$ will know $M(x)$ and hence S. Without loss of generality, let $\mathcal{A}_t$ controls the first t players. So $\mathcal{A}_t$ knows the vectors $F_1, F_2, \ldots, F_t$, t hash keys $\alpha_1, \alpha_2, \ldots, \alpha_t$ and t hash values with respect to each $F_i, 1 \leq i \leq n$. The hash values corresponding to $F_1, F_2, \ldots, F_t$ do not reveal any new information to $\mathcal{A}_t$ and hence can be removed from his view. However, the hash values corresponding to F_{t+1} give t new independent equations on elements of F_{t+1} to $\mathcal{A}_t$. So from properties of $USHash$, $\mathcal{A}_t$ falls short of $(\ell+t)-t = \ell$ points to completely know F_{t+1}. Now, the vectors $F_j, t+2 \leq j \leq n$ are linearly dependent on $F_j, 1 \leq j \leq t+1$. So the hash values corresponding to $F_j, t+2 \leq j \leq n$ are linear combination of the hash values corresponding to $F_j, 1 \leq j \leq t+1$ and hence can be removed from $\mathcal{A}_t$'s view. We prove this formally. Let us use the following notations:

- $g_i(x)$ is the $t-1$ degree polynomial defined by the first t values of $f_i(x)$. Since $\mathcal{A}_t$ controls the first t players, he knows each $g_i(x)$ from the vectors $F_1, F_2, \ldots, F_t$.
- $J(x) = (x-1) \times (x-2) \ldots (x-t)$

Let $F_j(x) = \sum_{i=1}^{\ell+t} f_i(j) \times x^{i-1}$ and $k_i(x) = f_i(x) - g_i(x)$. Then

$$\forall x \in \{1, 2, \ldots, t\} : \qquad k_i(x) = 0 \tag{1}$$

As $k_i(x)$ is a t degree polynomial and $\{1, 2, \ldots, t\}$ are its roots, so we get

$$k_i(x) = c_i * (x-1) * (x-2) * \ldots * (x-t) \Longrightarrow k_i(x) = c_i \times J(x) \Longrightarrow c_i = \frac{k_i(x)}{J_i(x)} \tag{2}$$

Now

$$f_i(x) = g_i(x) + k_i(x) \Longrightarrow f_i(x) = g_i(x) + c_i \times J(x) \tag{3}$$

Now $\mathcal{A}_t$ knows t hash values corresponding to F_{t+1}, which can be expressed as:

$$\begin{aligned} \forall j \in \{\alpha_1 \ldots \alpha_t\} : \qquad F_{t+1}(j) &= \sum_{i=1}^{\ell+t} f_i(t+1) * j^{i-1} \\ &= \sum_{i=1}^{\ell+t} (g_i(t+1) + c_i * J(t+1)) * j^{i-1} \text{ From Equation (3)} \end{aligned}$$

We now show that t hash values corresponding to $F_k, t+2 \leq k \leq n$ does not give any new information to $\mathcal{A}_t$. Consider any $k \in \{t+2, t+3, \ldots, n\}$ and any $j \in \{\alpha_1, \alpha_2, \ldots, \alpha_t\}$. Now similar to the last equation, we have

$$F_k(j) = \sum_{i=1}^{\ell+t} (g_i(k) + c_i * J(k)) * j^{i-1} \tag{4}$$

Now $F_k(j)$ can be expressed as

$$F_k(j) = \sum_{i=1}^{\ell+t} g_i(k) * j^{i-1} - \frac{J(k)}{J(t+1)} * \sum_{i=1}^{\ell+t} g_i(t+1) * j^{i-1}$$
$$+ \frac{J(k)}{J(t+1)} * \sum_{i=1}^{\ell+t} (g_i(t+1) + c_i * J(t+1)) * j^{i-1} \quad (5)$$
$$= \sum_{i=1}^{\ell+t} g_i(k) * j^{i-1} - \frac{J(k)}{J(t+1)} * \sum_{i=1}^{\ell+t} g_i(t+1) * j^{i-1}$$
$$+ \frac{J(k)}{J(t+1)} * F_{t+1}(j) \quad (6)$$

We see that in (6) all the terms are known to $\mathcal{A}_t$ and hence the t hash values corresponding to $F_{t+2}, F_{t+3} \ldots, F_n$ can be computed from the t hash values corresponding to $F_1, F_2, \ldots, F_{t+1}$. So, in sum, the view of $\mathcal{A}_t$ contains t points on each $f_i(x)$ (from the first t F_i's) and t hash values corresponding to F_{t+1}. Since each $f_i(x)$ is of degree t, $\mathcal{A}_t$ falls short of one point (on each $f_i(x)$) to completely interpolate each $f_i(x)$. Since ℓ elements of F_{t+1} are information theoretically secure, so is S. □

3 Single Round UWSS with $n = 3t + 1$

We now design a single round $(3t + 1, t)$ UWSS protocol **1-Round-UWSS** to share $S \in \mathbb{F}^\ell$.

Sharing Phase: **D** executes protocol **Secret Distribution** with $n = 3t + 1$. So P_i obtains the following from **D**: vector F_i of length $\ell + t$, the random hash key α_i and the n tuple $[v_{1i}\ v_{2i}\ \ldots\ v_{ni}]$ where for $1 \leq j \leq n$, $v_{ji} = USHash(\alpha_i; F_j)$.

Reconstruction Phase: (a) **Round 1**: For $1 \leq i \leq n$, P_i broadcasts the vector F'_i; (b) **Round 2**: For $1 \leq i \leq n$, P_i broadcasts the hash key α'_i and the n tuple $[v'_{1i}\ v'_{2i}\ \ldots\ v'_{ni}]$.

Local Computation (by each player):

1. Construct a directed graph G called *approval graph* over the set of n players, such that there exists an arc (P_k, P_j) (k can be equal to j) in G iff $v'_{jk} = USHash(\alpha'_k; F'_j)$, which indicates that P_k approves the vector F'_j broadcasted by P_j. Since all information are broadcasted, every (honest) player constructs the same graph G.

2. Each player whose in-degree (in G) is at least $n - t$ is included in a set $CORE$. Next, players in $CORE$ whose vectors are not approved by at least $n - t$ players in $CORE$ are removed from $CORE$. This process continues until no more players can be removed from $CORE$. Let $\overline{CORE} = \mathcal{P} \setminus CORE$.

3. Player $P_j \in CORE$, who has an arc (P_j, P_k) to player $P_k \in \overline{CORE}$ in G is removed from $CORE$, but not included in $\overline{CORE}$. If the removal of P_j from $CORE$ reduces the in-degree of some other player $P_l \in CORE$ to less than $n - t$ then remove P_l from $CORE$. This process continues, until no more player can be removed from $CORE$.

4. If $|CORE| < n - t$, then output NULL. Else try to reconstruct the original $n \times (\ell + t)$ matrix T as follows:
   (a) If $P_j \in CORE$, then insert F'_j as the j^{th} row of T. Since $|CORE| \geq n - t$, T will have at least $2t + 1$ rows.
   (b) Check if each column of T is t-consistent (**see Remark 2**). If not then output NULL. Else recover $M'(1), M'(2), \ldots, M'(\ell + t)$ by interpolating the values of each column and recover $M'(x)$ and compute S'.

Table 3: **Protocol 1-Round-UWSS: A Single Round UWSS Protocol with** $n = 3t + 1$

Remark 2 *Let $i_1, i_2, \ldots, i_k$ denote the indices of the rows which are filled in matrix T during step 4(b) of local computation of protocol* **1-Round-UWSS**. *Let $f'_j(i_1), f'_j(i_2), \ldots, f'_j(i_k)$ denote the values in the*

j^{th} column of the matrix T. Then j^{th} column is said to be t-consistent if there exists a polynomial $w_j(x)$ of degree t such that $w_j(i_1) = f'_j(i_1), w_j(i_2) = f'_j(i_2), \ldots, w_j(i_k) = f'_j(i_k)$.

Claim 1 *Let* **D** *be honest and P_i be an honest player. If some corrupted player P_j broadcasts $F'_j \neq F_j$ in reconstruction phase, then any arc like (P_i, P_j) may be present in G with probability at most $2^{-O(k)}$.*

PROOF: Let π_{ij} be the probability that arc (P_i, P_j) is present in G. Since **D** and P_i are honest, $\alpha'_i = \alpha_i$ and is unknown to $\mathcal{A}_t$ till **Round 2** of **Reconstruction Phase**. Thus P_j broadcasts $F'_j \neq F_j$ without knowing α_i. So from the properties of $USAuth$, $\pi_{ij} \leq \frac{\ell+t-1}{|\mathbb{F}|}$. Thus total probability that $\mathcal{A}_t$ can find an honest P_i and a corrupted P_j, such that arc (P_i, P_j) is present in G is at most $\sum_{i,j} \pi_{ij} \leq \frac{n^2(\ell+t-1)}{|\mathbb{F}|}$, which is at most $2^{-O(k)}$ in our context. □

Claim 2 *If* **D** *is honest, then except with probability at most $2^{-O(k)}$, an honest P_i is present in $CORE$*

PROOF: If **D** is honest, then P_i will have an incoming arc in G from all the honest players. The only reason that P_i is removed from $CORE$ is that there exists an arc (P_i, P_j) in G, where $P_j(\text{corrupted}) \in \overline{CORE}$. However, from Claim 1, this can happen with a probability at most $2^{-O(k)}$. □

Lemma 3 **1-Round-UWSS** *satisfy correctness property with error probability at most $2^{-O(k)}$.*

PROOF: If **D** is honest, then from Claim 1, with probability at least $1-2^{-O(k)}$, $CORE$ contains P_j only if $F'_j = F_j$. So each column of T will be t-consistent and hence $S' = S$ will be reconstructed. However, if a corrupted P_j who broadcasted $F'_j \neq F_j$, is included in $CORE$, then all the columns of partially filled matrix T will not be t-consistent and NULL will be output. This happens with probability at most $2^{-O(k)}$. □

Claim 3 *If* **D** *is corrupted and $|CORE| \geq n - t$, then at the end of the* **sharing phase** *there was a unique secret $S' \in \mathbb{F}^\ell \cup \{NULL\}$ defined by the honest players in $CORE$.*

PROOF: If **D** is corrupted and $|CORE| \geq n - t$ then it contains at least $(n - t) - t \geq t + 1$ honest players. Now the vector F'_i possessed by an honest player $P_i \in CORE$ is used to fill up the i^{th} row of T. Now consider the matrix T with only F'_i's corresponding to the honest players inserted in it. There are two possible cases: **(a) The values along each of the $\ell + t$ columns are t-consistent**: this implies that the F'_i's corresponding to the honest players in $CORE$ define an unique $\ell + t - 1$ degree polynomial $M'(x)$. Then the unique secret S' defined by the honest players in $CORE$ is the lower order ℓ coefficients of $M'(x)$. **(b) The values along at least one of the $\ell + t$ columns is not t-consistent**: In this case, the defined secret S' is NULL. □

Lemma 4 *Protocol* **1-Round-UWSS** *satisfies weak commitment property.*

PROOF: We have to consider the case when **D** is corrupted. If $|CORE| < n - t$ then NULL will be output. So let $|\mathbf{CORE}| \geq \mathbf{n - t}$. From Claim 3, $CORE$ contains a set $\mathcal{H}$ of at least $t+1$ honest players, who define a unique secret $S' \in \mathbb{F}^\ell \cup \{NULL\}$ at the end of **sharing phase**. Also, any player in $CORE$ cannot have an outgoing arc to any other player in $\overline{CORE}$. But the corrupted players (at most t) in $CORE$, along with the players outside $CORE$ (which could be at most t) may define some other secret S'' during reconstruction phase. However, in that case, $|CORE| \leq 2t < (n - t)$. Hence, the corrupted players cannot change the commitment from S' to S'' during reconstruction phase. But they could behave such that $NULL$ gets reconstructed. So weak commitment on S' holds. □

Theorem 2 **1-Round-UWSS** *is an efficient $(3t+1, t)$ UWSS protocol with agreement which privately communicates and broadcasts $O((n\ell + n^2)k)$ bits.*

PROOF: Secrecy follows from Lemma 2. Rest follows from the working of the protocol.

3.1 Improving the Results of [6]

In [6], the authors have given a single round honest dealer UVSS scheme without agreement for $n = 2t+1$, with single round of reconstruction, which shares k bits (i.e., $\ell = 1$ element from $\mathbb{F}$) by broadcasting $O(kn^2)$ bits ($O(n^2)$ elements from $\mathbb{F}$) during reconstruction phase. Though the scheme satisfies the lower bound of Theorem 1 for $\ell = 1$, it will not satisfy the lower bound for general ℓ. However, we now show that **1-Round-UWSS** can be converted into a single round honest dealer $(2t+1, t)$ UVSS without agreement, for any $\ell \geq 1$, with single round of reconstruction, whose communication complexity satisfies Theorem 1. Then the protocol is as follows: sharing phase will be same as in **1-Round-UWSS**. During reconstruction phase, player P_j broadcasts *only* F'_j. After this, there is no further communication. Now each player P_i locally verifies the F'_j's by using the *private* hash key and hash values possessed *only* by him. If number of F'_j's which passes verification is at least $n - t$, then P_i inserts all such F'_j in his local copy of T and then performs the same computation as in 4(b) on it. Else, it outputs NULL. Notice that the rows inserted (locally) in T by two different honest players may be different with very negligible probability because a corrupted F'_j can be approved (locally) by an honest player with very negligible probability. So, the resultant protocol will be an honest dealer UVSS without agreement.

Theorem 3 *There exists a 1-round honest dealer $(2t + 1, t)$ UVSS scheme without agreement, which shares ℓk bits by broadcasting $O((n\ell + n^2)k)$ bits during reconstruction phase.*

Substituting $H(S) = \ell k$ in Theorem 1, we find that the communication complexity of the reconstruction phase of our resultant UVSS scheme matches the existing lower bound.

4 Single Round UVSS with $n = 4, t = 1$

In [8] it is shown that there exists a single round $(5, 1)$ perfect VSS. We now design a single round $(4, 1)$ UVSS protocol called **1-Round-UVSS**, thus showing that probabilistically relaxing the conditions of VSS helps to increase the fault tolerance. The protocol is designed using the protocol **Secret Distribution** given in Section 2 as a black-box and is similar to our single round UWSS.

Let the players be denoted by P_1, P_2, P_3, P_4, where P_1 is dealer and S is the secret.

Sharing Phase: Same as the sharing phase of protocol **1-Round-UWSS**, with $n = 4$ and $t = 1$.

Reconstruction Phase: Same as in **1-Round-UWSS**, except that **D** (P_1) is not allowed to participate.

Local Computation by players $P_i, 2 \leq i \leq 4$: Construct the *approval graph* G (as in protocol **1-Round-UWSS**) over P_2, P_3 and P_4, using the information broadcasted by P_2, P_3 and P_4 during reconstruction phase. All the players who have in-degree at least two in G are included in $CORE$. Remove all the players from $CORE$ who do not have an in-coming arc (in G) from at least two players in $CORE$. Then do the following:

1. If $|CORE| = 0$, then construct $M'(x)$ using F'_2 and F'_3, reconstruct S' and terminate. /* **D** is corrupted. */

2. If $|CORE| = 2$, then construct $M'(x)$ using the F'_i's of the players in $CORE$, reconstruct S' and terminate.

3. If $|CORE| = 3$ and each player in $CORE$ has an incoming arc from all the players in $CORE$, then do the same computation as in the previous case.

4. If $|CORE| = 3$, but at least one player in $CORE$ does not have an incoming arc from all the players in $CORE$, then construct $M'(x)$ using F'_2 and F'_3 and reconstruct S' and terminate.

Table 4: **Protocol 1-Round-UVSS: A Single Round UVSS Protocol with $n = 4$ and $t = 1$**

The secrecy of **1-Round-UVSS** follows from the secrecy of **1-Round-UWSS**. We now show that the protocol satisfies correctness and strong commitment property.

Claim 4 *Protocol* **1-Round-UVSS** *satisfies correctness property with very high probability.*

PROOF: If **D** is honest, then among the remaining three players at most one can be corrupted. Let P_4 be the corrupted player among P_2, P_3 and P_4. Then P_2 and P_3 will be present in $CORE$ (since P_2 (P_3) will have incoming arcs from P_2 and P_3 in G). From Claim 1, if P_4 is also present in $CORE$, then with very high probability, it has broadcasted $F'_4 = F_4$. The proof now follows using similar argument as in Lemma 3. □

Claim 5 *Protocol* **1-Round-UVSS** *satisfies strong commitment property.*

PROOF: We have to only consider the case when **D** (P_1) is corrupted. In this case, P_2, P_3 and P_4 are honest and behave correctly in reconstruction phase (recall that **D** is not allowed to participate in reconstruction phase). Note that the F' vectors corresponding to any two honest players define a unique secret S' because here $t = 1$. Now we divide our argument depending upon the size of $CORE$. If $|CORE| = 0$, then it implies that **D** has not given consistent values to anybody during sharing phase. So secret S' is reconstructed from F'_2 and F'_3, implying that S' is the unique secret defined by **D** in the sharing phase and is reconstructed (in reconstruction phase) irrespective of the behavior of the corrupted player (**D**). The similar argument holds for the case when $|CORE| = 3$ and at least one player in $CORE$ do not have an incoming arc from all the players in $CORE$. For the case when $|CORE| = 2$ or $|CORE| = 3$, with each player in $CORE$ having an incoming arc from all the players in $CORE$, the committed secret is the one defined by the polynomials of the players in $CORE$. □

5 Two Round UVSS with $n = 3t + 1$

We now design a two round $(3t + 1, t)$ UVSS protocol **2-Round-UVSS**, given in Table 6, to share $S = s$; i.e., $\ell = 1$. For that, we first design a two round $(3t + 1, t)$ UWSS protocol **2-Round-UWSS**, given in Table 5, which is used as a black-box in **2-Round-UVSS**. We do not proof the properties of **2-Round-UWSS**.

The principle behind our *two* round UVSS protocol is similar to the *three* round *perfect* (where error probability is 0) VSS protocol proposed in [7]. The secret s is hidden by **D** in a bivariate polynomial $F(x, y)$ and each player P_i gets the univariate polynomials $F(x, i)$ and $F(i, y)$. Then every pair of players compare their common shares by "binding" them with a random pad and broadcasting them. In the reconstruction phase the random pads are revealed, allowing the players to compute the shares and finally reconstruct the secret. To ensure that P_i discloses the same random pads in reconstruction phase, P_i shares a random field element using **2-Round-UWSS** and chooses his random pads as *points on the respective polynomials* which are given to the individual players as part of protocol **2-Round-UWSS**. During reconstruction phase, players whose instance of protocol **2-Round-UWSS** fails, get disqualified from the main protocol. On the other hand, players whose instance of single round UWSS succeeds, disclose their original pads. Note that if **D** is corrupted, then he can distribute inconsistent values to the honest players during first round of sharing phase. So when the honest players compare their common shares during second round of sharing phase, they may find them to be inconsistent. In the three round perfect VSS protocol of [7], such inconsistencies are resolved by **D** during third round of sharing phase, which cannot be done here because sharing phase has now only two rounds. However, in spite of this, our protocol satisfies the requirement of UVSS. Note that we could not use **1-Round-UWSS** as a black-box to design a 2-round UVSS. Before discussing the proofs of protocol **2-Round-UVSS**, we first give the following definition.

Sharing Phase (Two Rounds): **Round 1**:

• **D** chooses a random bivariate polynomial $F(x, y)$ over $\mathbb{F}$ of degree t in each variable such that $F(0, 0) = s$. **D** privately sends to player P_i the polynomials $f_i(x) = F(x, i)$ and $g_i(y) = F(i, y)$.

• Player $P_i, 1 \leq i \leq n$, acting as a dealer, starts single round UWSS protocol **1-Round-UWSS**^{P_i} in order to share a random value $s_i \in \mathbb{F}$ (thus UWSS protocol is executed for $\ell = 1$ secret). Let the vectors distributed (privately) by P_i to the n players in **1-Round-UWSS**^{P_i} be denoted by $F_1^{iW}, F_2^{iW}, \ldots, F_n^{iW}$, where $P_j, 1 \leq j \leq n$ receives F_j^{iW}. Since $\ell = 1$, F_j^{iW} consists $t + 1$ elements. Accordingly, let $F_j^{iW}(x)$ denotes the t degree polynomial formed by using the elements of F_j^{iW} as coefficients (starting from the lower order terms).

Round 2: Player $P_i, 1 \leq i \leq n$ broadcasts the following: $a_{ij} = f_i(j) + F_j^{iW}(0)$ and $b_{ij} = g_i(j) + F_i^{jW}(0)$.
/* $F_j^{iW}(0)$ denotes constant term of $F_j^{iW}(x)$ received by P_j from P_i in **1-Round-UWSS**^{P_i}. */

Local Computation (by each player):

• Player P_i and P_j are said to be consistent if $a_{ij} = b_{ji}$ and $b_{ij} = a_{ji}$. Form a consistency graph G over the set of n players, where there exists an edge between P_i and P_j if they are consistent with each other. Since a_{ij}'s and b_{ij}'s are public information, same G will be constructed by all (honest) players.

• Construct a set $CORE^{Sh}$ ($= \emptyset$ initially) and add P_i in $CORE^{Sh}$ if degree of P_i (in G) is at least $n - t$. Remove P_j from $CORE^{Sh}$ if P_j is not consistent with at least $n - t$ players in $CORE^{Sh}$. Continue this process till no more players can be removed. If $|CORE^{Sh}| < n - t$ then discard **D** and terminate the protocol.

Reconstruction Phase (Two Rounds): Only the players in $CORE^{Sh}$ participate.

• For each $P_i \in CORE^{Sh}$, concurrently run the reconstruction phase of **1-Round-UWSS**^{P_i}. This will take **two** rounds. If reconstruction phase fails (i.e., output is NULL), then remove P_i from $CORE^{Sh}$.

• If the reconstruction phase of **1-Round-UWSS**^{P_i} does not fail, then the vectors and hence the polynomials $F_j^{iW}(x), 1 \leq j \leq n$, distributed by P_i in **1-Round-UWSS**^{P_i} to the n players are recovered. Now compute $f_i(j) = a_{ij} - F_j^{iW}(0)$, $1 \leq j \leq n$ using the values a_{ij}, which P_i broadcasted during second round of sharing phase. If there exists a polynomial $f_i(x)$ of degree at most t passing through the $f_i(j)$'s, then include P_i in a set $CORE^{Rec}$ ($= \emptyset$ initially)

• Consider all players from $CORE^{Rec}$ and use their reconstructed $f_i(x)$'s to construct a bivariate polynomial $F'(x, y)$. If $F'(x, y)$ is of degree t in both x and y, then reconstruct $s' = F'(0, 0)$. Else output NULL.

Table 5: **Protocol 2-Round-UWSS: A Two Round UWSS with** $n = 3t + 1$

Definition 5 *In protocol* **2-Round-UVSS***, we say that a player P_i is consistent with bivariate polynomial $F(x, y)$ if the polynomials given to P_i during sharing phase, namely $f_i(x)$ and $g_i(y)$ lie on $F(x, y)$; i.e., $f_i(x) = F(x, i)$ and $g_i(y) = F(i, y)$.*

Lemma 5 *If* **D** *is honest then except with probability 2^{-k}, $CORE^{Rec}$ contains each honest player, consistent with the bivariate polynomial $F(x, y)$ defined by* **D***. Moreover even if a dishonest player is present in $CORE^{Rec}$, it is consistent with $F(x, y)$.*

Proof: If **D** is honest, then the information received by each honest player during sharing phase will be consistent with bivariate polynomial $F(x, y)$ and hence they will be pairwise consistent and will be included in $CORE^{Sh}$. From the properties of **2-Round-UWSS**, for each honest player P_i, **2-Round-UWSS**^{P_i} will succeed with probability at least $(1-2^{-k})$ and their corresponding recovered polynomials $f_i(x)$ will be t-consistent. So all the honest players (at least $2t + 1$) will be in $CORE^{Rec}$ and will define $F(x, y)$.

Now consider a dishonest player $P_j \in CORE^{Rec}$. This implies that P_j is consistent with at least $(n - t) - t \geq t + 1$ honest players in $CORE^{Rec}$, who define the bivariate polynomial $F(x, y)$. Also $P_j \in CORE^{Rec}$ implies that **2-Round-UWSS**^{P_j} is successful and the recovered $f_j(x)$ is t-consistent. Since P_j is consistent with at least $t + 1$ honest players in $CORE^{Rec}$, who define $F(x, y)$ and since recovered $f_j(x)$ is t-consistent, it follows that recovered $f_j(x)$ is consistent with $F(x, y)$. □

Sharing Phase (Two Rounds): **Round 1**:

• **D** chooses a random bivariate polynomial $F(x, y)$ over $\mathbb{F}$ of degree t in each variable such that $F(0, 0) = s$. **D** privately sends to player P_i the polynomials $f_i(x) = F(x, i)$ and $g_i(y) = F(i, y)$.

• Player $P_i, 1 \leq i \leq n$, acting as a dealer, starts two round UWSS protocol **2-Round-UWSS**^{P_i} in order to share a random value $s_i \in \mathbb{F}$ by using a bivariate polynomial $F^{iW}(x, y)$ of degree t in both variable, such that $F^{iW}(0, 0) = s_i$. Thus P_j gets the polynomials $F^{iW}(x, j)$ and $F^{iW}(j, y)$ from P_i as a part of UWSS.

Round 2: Player $P_i, 1 \leq i \leq n$ broadcasts the following: $a_{ij} = f_i(j) + F^{iW}(0, j)$ and $b_{ij} = g_i(j) + F^{jW}(0, i)$. Concurrently, **Round 2** of **Sharing Phase** of **2-Round-UWSS**^{P_i} is executed.

Local Computation (by each player):

• Player P_i and P_j are said to be consistent if $a_{ij} = b_{ji}$ and $b_{ij} = a_{ji}$. Form a consistency graph G over the set of n players, where there exists an edge between P_i and P_j if they are consistent with each other. Since a_{ij}'s and b_{ij}'s are public information, same G will be constructed by all (honest) players.

• Construct a set $CORE^{Sh}$ ($= \emptyset$ initially) and add P_i in $CORE^{Sh}$ if degree of P_i (in G) is at least $n - t$. Remove P_j from $CORE^{Sh}$ if P_j is not consistent with at least $n - t$ players in $CORE^{Sh}$. Continue this process till no more players can be removed. Remove P_i from $CORE^{Sh}$ if P_i gets disqualified as the dealer in protocol instance **2-Round-UWSS**^{P_i}. If $|CORE^{Sh}| < n - t$ then discard **D** and terminate the protocol.

• If $P_i \in CORE^{Sh}$, then let $CORE^{Sh}_{iW}$ denote the $CORE^{Sh}$ corresponding to the instance **2-Round-UWSS**^{P_i}. For each $P_i \in CORE^{Sh}$, check if $|CORE^{Sh} \cap CORE^{Sh}_{iW}| \geq n - t$. If not, then discard P_i from $CORE^{Sh}$. If $|CORE^{Sh}| < n - t$ then discard **D** and terminate the protocol.

Reconstruction Phase (Two Rounds): Only the players in $CORE^{Sh}$ participate.

• Initialize $CORE^{Rec} = CORE^{Sh}$. For each $P_i \in CORE^{Rec}$, run the reconstruction phase of **2-Round-UWSS**^{P_i}. This takes **two** rounds. If reconstruction phase fails (i.e., output is NULL), then remove P_i from $CORE^{Rec}$.

• For each $P_i \in CORE^{Rec}$, use the values a_{ij} broadcasted by him during **Round 2** of **sharing phase** to compute $f_i(j) = a_{ij} - F^W_i(0, j)$, $1 \leq j \leq n$. If there exists a polynomial $f_i(x)$ of degree at most t passing through the $f_i(j)$'s, then keep P_i in $CORE^{Rec}$, otherwise discard it from $CORE^{Rec}$.

• Consider all players from $CORE^{Rec}$ and use their reconstructed $f_i(x)$'s to construct a bivariate polynomial $F'(x, y)$. If $F'(x, y)$ is of degree t in both x and y, then reconstruct $s' = F'(0, 0)$. Else output a predefined $\hat{s} \in \mathbb{F}$.

Table 6: **Protocol 2-Round-UVSS: A Two Round UVSS with** $n = 3t + 1$

Claim 6 *If* **D** *is dishonest and does not get disqualified during sharing phase, then* $CORE^{Sh}$ *contains at least* $t + 1$ *honest players. Moreover, except with probability* $2^{-O(k)}$*, each honest player in* $CORE^{Sh}$ *will be present in* $CORE^{Rec}$*.*

Proof: If **D** is dishonest and does not get disqualified during sharing phase then it implies that $CORE^{Sh}$ contains at least $n - t$ players, of which $(n - t) - t \geq t + 1$ are honest. Now a player P_i gets removed from $CORE^{Rec}$ in only two cases: (a) the reconstruction phase of **2-Round-UWSS**^{P_i} fails or (2) the reconstruction phase of **2-Round-UWSS**^{P_i} is successful but the resulting polynomial $f_i(x)$ is of degree larger than t. However, from the properties of single round UWSS, for an honest P_i, the first event can occur with probability at most 2^{-k}, where as the second event cannot occur at all. Hence, each honest player present in $CORE^{Sh}$ will also be present in $CORE^{Rec}$ with very high probability. □

Lemma 6 *If* **D** *is dishonest and does not get disqualified during sharing phase, then except with probability* 2^{-k}*, the protocol satisfies strong commitment property.*

Proof: From Claim 6, if **D** is dishonest and does not get disqualified during sharing phase, then except with probability 2^{-k}, each honest player (at least $t + 1$) of $CORE^{Sh}$ will also be present in $CORE^{Rec}$. Now there are three possible cases:

1. $CORE^{Sh}$ contains exactly $t+1$ honest players: In this case $|CORE^{Sh}| = 2t+1$ and it contains t corrupted players. It also implies that the honest players in $CORE^{Sh}$ are consistent with each other and define a bi-variate polynomial $F'(x,y)$ of degree at most t in both x and y. Moreover, the corrupted players in $CORE^{Sh}$ are also consistent with these $t+1$ honest players. From Claim 6, these $t+1$ honest players will be present in $CORE^{Rec}$. Now if the remaining t corrupted players in $CORE^{Sh}$ are also present in $CORE^{Rec}$, it implies that these corrupted players are also consistent with $F'(x,y)$ (following the argument provided for the second part of Lemma 5). So in reconstruction phase, $s' = F'(0,0)$ will be reconstructed.

2. $CORE^{Sh}$ contains more than $t+1$ honest players, who are all consistent with each other: Similar to previous case, here also all honest players in $CORE^{Sh}$ define a unique bi-variate polynomial $F'(x,y)$. Also if a corrupted player is present in $CORE^{Sh}$, then it implies that it is consistent with at least $(n-t)-t \geq t+1$ honest players in $CORE^{Sh}$ and hence with $F'(x,y)$. Now following the same argument given in the previous case $s' = F'(0,0)$ will be reconstructed.

3. $CORE^{Sh}$ contains more than $t+1$ honest players, but are not consistent with each other: Hence the $f_i(x)$ polynomials of all honest players in $CORE^{Sh}$ does not define a bivariate polynomial of degree at most t in both x and y. In this case, **D** has committed a secret which is a predefined (standard) value $\hat{s}$ from $\mathbb{F}$. From Claim 6, each honest player from $CORE^{Sh}$ will be present in $CORE^{Rec}$, except with probability at most 2^{-k}. Now irrespective of whether the corrupted players in $CORE^{Sh}$ are present in $CORE^{Rec}$ or not, the $f_i(x)$ polynomials corresponding to the honest players in $CORE^{Rec}$ will not reconstruct a bivariate polynomial of degree at most t in both x and y. Hence $\hat{s}$ will be reconstructed and so the strong commitment on $\hat{s}$ is satisfied. □

Remark 3 *Note that the third case in the proof of Lemma 6 is different from the* WEAK COMMITMENT *property of UWSS. In the* WEAK COMMITMENT *property, there exists a $s^* \in \mathbb{F}$ which is defined after the sharing phase, such that depending upon the behavior of corrupted players during reconstruction phase, either s^* or NULL is reconstructed. On the other hand, in the third case of Lemma 6, the shares given by* **D** *to the players in $CORE^{Sh}$ does not define a unique secret. So it can be viewed as* **D** *committing a fixed $\hat{s} \in \mathbb{F}$.* **Now irrespective of the behavior of the corrupted players during reconstruction phase, D's commitment on $\hat{s}$ is not violated**.

Lemma 7 *Protocol* **2-Round-UVSS** *satisfies perfect secrecy.*

PROOF (SKETCH): Follows using entropy based argument, used to prove the secrecy of 3 round perfect VSS protocol of [7]. Informally, the proof follows from the secrecy of **2-Round-UWSS** and properties of bivariate polynomial of degree t. □

Theorem 4 **2-Round-UVSS** *is an efficient two round $(3t+1, t)$ UVSS protocol which privately communicates $O(n^4k)$ bits and broadcasts $O(n^4k)$ bits.*

PROOF: Follows from Lemma 7, Lemma 5 and Lemma 6 and working of the protocol. □

6 Three Round UWSS with $n = 2t+1$

We design a three round $(2t+1, t)$ UWSS protocol **3-Round-UWSS**, given in Table 7 to share $S \in \mathbb{F}^\ell$.

Theorem 5 *In protocol* **3-Round-UWSS**, *the following must hold:*

1. **D** *is honest and*
   (a) If P_i is honest, then $P_i \in NB$ and V_i^{Sh} and $V_{FR_i}^{Rec}$ are same at least at $(t+1)$ locations.
   (b) If $P_i \in NB$ is dishonest and broadcast at least one of the polynomial $F_i(x)$ or $R_i(x)$ incorrectly in reconstruction phase, then V_i^{Sh} and $V_{FR_i}^{Rec}$ mismatches at least at $(t+1)$ locations with probability more than $(1-2^{-O(k)})$.

2. **D** *is dishonest and if P_i is honest and $P_i \in NB$, then V_i^{Sh} and $V_{FR_i}^{Rec}$ matches at least at $(t+1)$ locations (irrespective of the values at these locations) with probability more than $(1-2^{-O(k)})$.*

Sharing Phase, Round 1: **D** executes **Secret Distribution** to share S, ensuring that each random *hash key* $\alpha_i \in \mathbb{F} - \{0, 1, \ldots, n-1\}$. Hence P_i gets the vector F_i consisting of $\ell + t$ elements. Let $F_i(x)$ be the $\ell + t - 1$ degree polynomial formed by using the elements of F_i as coefficients (in increasing order of power). In addition, P_i also gets the *hash key* α_i and the n tuple $[v_{1i}, v_{2i}, \ldots, v_{ni}]$ from **D**, where for $1 \leq j \leq n, v_{ji} = USHash(\alpha_i; F_j)$. Moreover, **D** also gives another random $\ell + t - 1$ degree polynomial $R_i(x)$ and n tuple $[r_{1i}, r_{2i}, \ldots, r_{ni}]$ to P_i, where for $1 \leq j \leq n, r_{ji} = USHash(\alpha_i; R_j)$. Here R_i is the vector consisting of the coefficients of $R_i(x)$.

Round 2: Player P_i chooses a random $d_i \in \mathbb{F} \setminus \{0\}$ and broadcasts $B_i(x) = d_i F_i(x) + R_i(x)$, along with d_i.

Round 3: For $1 \leq j \leq n$, **D** checks $d_i v_{ij} + r_{ij} \stackrel{?}{=} B_i(\alpha_j)$. If **D** finds any inconsistency, he broadcasts $F_i(x)$ and hence the vector F_i. Parallely, player P_j, $1 \leq j \leq n$ broadcasts "Accept" or "Reject", depending upon $d_i v_{ij} + r_{ij} \stackrel{?}{=} B_i(\alpha_j)$.

Local Computation (by each player):

1. Divide the set $\mathcal{P}$ in two sets. Each player P_i whose $F_i(x)$ is broadcasted by **D** during third round are included in a set B. The remaining players are included in another set NB. If $|B| > t$ then discard **D** and terminate.

2. For each $P_i \in NB$, construct an n length bit vector V_i^{Sh}, where for $1 \leq j \leq n$, the j^{th} bit is 1, if P_j has broadcasted "Accept" during **Round 3**, otherwise 0. V_i^{Sh} is public as it is constructed using broadcasted information. If $\exists P_i \in NB$ such that V_i^{Sh} contains at least $t+1$ 0's, then discard **D** and terminate the protocol.

Reconstruction Phase (a) **Round 1**: Each $P_i \in NB$ broadcasts $F_i'(x)$ (hence F_i') and $R_i'(x)$; (b) **Round 2**: Each $P_i \in NB$ broadcasts $[v_{1i}', v_{2i}', \ldots, v_{ni}']$, $[r_{1i}', r_{2i}', \ldots, r_{ni}']$ and α_i'.

Local Computation (by each player):

1. For the polynomial $F_i'(x)$ broadcasted by $P_i \in NB$, construct an n length response vector $V_{F_i(x)}^{Rec}$ where the j^{th} bit of $V_{F_i(x)}^{Rec}$ contains 1 if $v_{ij}' = USHash(\alpha_j'; F_i')$, otherwise 0. Similarly, construct the response vector $V_{R_i(x)}^{Rec}$ corresponding to $R_i(x)$. Finally compute $V_{FR_i}^{Rec} = V_{F_i(x)}^{Rec} \otimes V_{R_i(x)}^{Rec}$, where $\otimes$ denotes bit wise AND.

2. $P_i \in NB$ is included in $CORE^{Rec}$ ($= \emptyset$ initially) if V_i^{Sh} matches with $V_{FR_i}^{Rec}$ at least at $t+1$ locations.

3. Set $CORE = B \cup CORE^{Rec}$. If $|CORE| < n - t$, then output NULL and terminate. Else try to reconstruct the original $n \times (\ell + t)$ matrix T constructed by **D** in protocol **Secret Distribution** during first round, as follows:

   - If $P_j \in NB$ and included in $CORE$ then insert F_j' as the j^{th} row of T. If $P_i \in B$, then insert F_i broadcasted by **D** in third round as i^{th} row of T. Since $|CORE| > n - t$, at least $t+1$ rows will be inserted in T. Now performs the same computation as done in step 4(b) of local computation during reconstruction phase of **Protocol 1-Round-UWSS**.

Table 7: **Protocol 3-Round-UWSS: A Three Round UWSS Protocol with** $n = 2t + 1$

PROOF: Property 1(a) is easy. For 1(b), let **D** be honest, P_i be dishonest and $P_i \in NB$. This implies that **D** and all the honest players are satisfied by the values broadcasted by P_i during second round of **sharing phase**. So V_i^{Sh} will contain 1 at $(t+1)$ locations corresponding to honest players. Now if P_i broadcasts incorrect $F_i'(x) \neq F_i(x)$ during reconstruction phase, then $V_{F_i(x)}^{Rec}$ may contain 1 at the j^{th} position, corresponding to an honest player P_j, provided $v_{ij} = USHash(\alpha_j; F_i')$. However, since **D** is honest, the **hash key** α_j and the *hash value* v_{ij} is unknown to P_i at the time of broadcasting $F_i'(x)$. So from the properties of $USHash$, $v_{ij} = USHash(\alpha_j; F_i')$ with probability $\pi_{ij} \leq \frac{\ell+t-1}{|\mathbb{F}|-n}$. Thus total probability that adversary can find P_i, P_j such that a corrupted player P_i will be approved by

an honest player P_j is at most $\sum_{i,j} \pi_{ij} \leq \frac{n^2(\ell+t-1)}{|\mathbb{F}|} \approx 2^{-O(k)}$. Similar argument holds if P_i broadcasts $R_i'(x) \neq R_i(x)$. Thus if P_i broadcasts incorrect $F_i'(x)$ or $R_i'(x)$, then except with probability at most $2^{-O(k)}$, V_i^{Sh} and $V_{FR_i}^{Rec}$ mismatches at $(t+1)$ locations corresponding to honest players.

For property 2, we say that $v \in \mathbb{F}$ is α_j consistent with a polynomial $F(x)$ over $\mathbb{F}$ if $F(\alpha_j) = v$. Here $\alpha_j \in \mathbb{F}$. Now consider the case when $\mathbf{D}$ is dishonest and $P_i \in NB$ is an honest player. We show that V_i^{Sh} and $V_{F_i(x)}^{Rec} \otimes V_{R_i(x)}^{Rec}$ will match at $(t+1)$ locations (irrespective of the values at these locations) corresponding to $(t+1)$ honest players. For this, consider an honest P_j. Now there are two possible cases: **(a) V_i^{Sh} contains 0 at j^{th} position**: Thus $d_i v_{ij} + r_{ij} \neq B_i(\alpha_j)$, implying that either v_{ij} is not α_j-consistent with $F_i(x)$ or r_{ij} is not α_j-consistent with $R_i(x)$ or both. So during reconstruction phase the j^{th} location in $V_{F_i(x)}^{Rec} \otimes V_{R_i(x)}^{Rec}$ will also contain 0. **(b) V_i^{Sh} contains 1 at j^{th} position**: Thus $d_i v_{ij} + r_{ij} = B_i(\alpha_j)$, implying that either both v_{ij} and r_{ij} are α_j-consistent with $F_i(x)$ and $R_i(x)$ or both v_{ij} and r_{ij} are not α_j-consistent with $F_i(x)$ and $R_i(x)$. The problem comes in later case, when both v_{ij} and r_{ij} are not α_j-consistent with $F_i(x)$ and $R_i(x)$ but still j^{th} location in V_i^{Sh} is 1. We claim that this can happen for an unique $d_i \in \mathbb{F} - \{0\}$ for the pair $F_i(x)$ and $R_i(x)$, which the dishonest $\mathbf{D}$ must guess with probability $\frac{1}{|\mathbb{F}|-1} \approx 2^{-O(k)}$ during first round. If there exist another $e_i \in \mathbb{F} - \{0\}$, such that $e_i v_{ij} + r_{ij}$ is also α_j consistent with $e_i F_i(x) + R_i(x)$. This implies $(d_i - e_i)v_{ij}$ is α_j consistent with $(d_i - e_i)F_i(x)$ or v_{ij} is α_j consistent with $F_i(x)$ which is a contradiction. Hence if V_i^{Sh} contains 1 at j^{th} position, then so does $V_{F_i(x)}^{Rec} \otimes V_{R_i(x)}^{Rec}$ with probability $\geq 1 - 2^{-O(k)}$. □

Lemma 8 *If $\mathbf{D}$ is not discarded in sharing phase then the probability that an honest player P_i will be included in CORE is at least $1 - 2^{-O(k)}$.*

PROOF: Since $CORE = B \cup CORE^{Rec}$, all honest players in B will be included in $CORE$. We now show that all honest players in NB are included in $CORE^{Rec}$ and hence in $CORE$, with very high probability. Now $P_i \in NB$ is included in $CORE^{Rec}$ if V_i^{Sh} matches with $V_{F_i}^{Rec} \otimes V_{R_i}^{Rec}$ atleast at $t+1$ locations. From Theorem 5, for an honest P_i, this condition will be satisfied with probability 1 for an honest $\mathbf{D}$ and with probability at least $(1 - 2^{-k})$ for a dishonest $\mathbf{D}$. Hence except with probability 2^{-k}, an honest $P_i \in NB$ will be added in $CORE^{Rec}$ and hence in $CORE$. □

Lemma 9 *If $\mathbf{D}$ is honest then B will contain all corrupted players and $CORE^{Rec}$ will contain all players who disclose correct $F_i(x)$ and $R_i(x)$ (as given by $\mathbf{D}$) in reconstruction phase. Moreover, players in NB who disclose incorrect $F_i(x)$ or $R_i(x)$ or both during reconstruction phase, will not be included in $CORE^{Rec}$ with probability at least $(1 - 2^{-k})$.*

PROOF: It is easy to see that when $\mathbf{D}$ is honest, B contains only corrupted players. Now a player $P_i \in NB$ is included in $CORE^{Rec}$ if V_i^{Sh} matches with $V_{F_i(x)}^{Rec} \otimes V_{R_i(x)}^{Rec}$ at least at $t+1$ locations. Now according to Theorem 5, when $\mathbf{D}$ is honest, this property is always true if P_i is honest, where as it may hold with probability at most 2^{-k} if P_i is corrupted and broadcasted incorrect $F_i(x)$ or $R_i(x)$ during reconstruction phase. Hence the lemma. □

Lemma 10 *If $\mathbf{D}$ is dishonest, then $CORE^{Rec}$ can contain dishonest players who discloses incorrect secrets and their authentication information.*

PROOF: Follows from Theorem 5. □

Theorem 6 3-Round-UWSS *is an efficient three round $(2t+1, t)$-UWSS protocol with agreement which privately communicates and broadcasts $O((n\ell + n^2)k)$ bits.*

PROOF: Communication complexity and efficiency is easy to follow. We now prove the properties of UWSS.

1. SECRECY: We only need to consider the case when **D** is honest. Without loss of generality let $\mathcal{A}_t$ controls the first t players. The proof will be similar to the proof of Lemma 2, where the secrecy is shown by proving that lower order ℓ elements of F_{t+1} (and hence ℓ coefficients of $F_{t+1}(x)$) are information theoretically secure. We now prove that same holds here also. From the properties of $USHash$, ℓ coefficients of $R_{t+1}(x)$ are information theoretically secure. Since $F_{t+1}(x)$ and $R_{t+1}(x)$ are independent of each other and d_{t+1} is randomly selected, it implies that $B_{t+1}(x) = d_{t+1}F_{t+1}(x) + R_{t+1}(x)$ has a completely independent distribution from $F_{t+1}(x)$ and $R_{t+1}(x)$. So even the knowledge of $B_{t+1}(x)$ keeps lower order ℓ coefficients of $F_{t+1}(x)$ information theoretically secure.

2. CORRECTNESS: Follows from the fact that if **D** is honest then with probability at least $1-2^{-O(k)}$, all the players in $CORE^{Rec}$ produces correct information during **reconstruction phase**.

3. WEAK COMMITMENT: If a dishonest **D** is not discarded during sharing phase, then from Lemma 8, except with probability $2^{-O(k)}$, each honest $P_i \in NB$ will be present in $CORE$, along with its corresponding $F_i(x)$. If $F_i(x)$'s corresponding to the honest players in $CORE$ does not define a unique secret s', then irrespective of the polynomials broadcasted by corrupted players in $CORE$ during reconstruction phase, NULL will be output. On the other hand, if the $F_i(x)$'s corresponding to the honest players in $CORE$ define a unique secret s', then depending upon the polynomials broadcasted by the corrupted players in $CORE$, either s' or NULL will be output. □

Comparison with UWSS Protocol of [5]: The first two steps of UVSS protocol of [5], along with some additional checking constitutes a five round $(2t+1, t)$ UWSS with agreement, which shares an $\ell = 1$ length secret (i.e., single field element) with a communication overhead of $O(n^3k)$ bits (both private and broadcast). So to share ℓ length secret, the UWSS protocol of [5] will have a communication overhead of $O(n^3\ell k)$ bits. Comparing this with Theorem 6, we find that our UWSS protocol significantly improves the round and communication complexity of UWSS protocol of [5].

7 Information Checking Protocol with $n = 2t + 1$

We now present a three round IC protocol **IC**, which allows **D** to sign on secret $S \in \mathbb{F}^\ell$.

Lemma 11 *If INT is honest and* **D** *has not broadcasted $F(x)$ during* **Round 3**, *then V^{Sh} and $V^{Rec}_{FR} = V^{Rec}_{F(x)} \otimes V^{Rec}_{R(x)}$ will match at atleast $t+1$ locations (irrespective of the values at these location) corresponding to the honest receivers in $\mathcal{P}$, except with probability at most $2^{-O(k)}$.*

PROOF: The proof follows using similar argument used to prove property 2 of Theorem 5. □

Lemma 12 *If* **D** *is honest, then a corrupted INT will be unable to forge* **D***'s signature on $S' \neq S$, except with an error probability of at most $2^{-O(k)}$.*

PROOF: Similar to the proof of property 1(b) of Theorem 5. □

Lemma 13 *If* **D** *and INT are uncorrupted, then S is information theoretic secure up to* **RevealVal**.

PROOF: Proof follows using similar argument used to prove privacy of **3-Round-UWSS**. □

Theorem 7 *Protocol* **IC** *is an efficient three round IC scheme, which privately communicates and broadcasts $O((\ell + n)k)$ bits.*

IC($\mathbf{D}, INT, \mathcal{P}, S$)

Distr($\mathbf{D}, INT, \mathcal{P}, S$): Round 1: $\mathbf{D}$ selects a random $\ell + t - 1$ degree polynomial $F(x)$ over $\mathbb{F}$, whose lower order ℓ coefficients are elements of S. In addition, $\mathbf{D}$ selects another random $\ell + t - 1$ degree polynomial $R(x)$, over $\mathbb{F}$, which is independent of $F(x)$. $\mathbf{D}$ selects n distinct random elements $\alpha_1, \alpha_2, \ldots, \alpha_n$ from $\mathbb{F} - \{0, 1, \ldots, n-1\}$. Let F and R denote the $\ell + t$ length vector consisting of the coefficients of $F(x)$ and $R(x)$ respectively (in the order of increasing power). $\mathbf{D}$ privately gives $F(x)$ and $R(x)$ to INT. To receiver $P_i \in \mathcal{P}$, $\mathbf{D}$ privately gives α_i, v_i and r_i, where $v_i = USHash(\alpha_i; F)$ and $r_i = USHash(\alpha_i; R)$. The polynomial $R(x)$ is called **authentication information**, while for $1 \leq i \leq n$, the values α_i, v_i and r_i are called **verification information**.

AuthVal($\mathbf{D}, INT, \mathcal{P}, S$): Round 2: INT chooses a random $d \in \mathbb{F} \setminus \{0\}$ and broadcasts $d, B(x) = dF(x) + R(x)$.

Round 3: For $1 \leq j \leq n$, $\mathbf{D}$ checks $dv_j + r_j \stackrel{?}{=} B(\alpha_j)$. If $\mathbf{D}$ finds any inconsistency, he broadcasts $F(x)$. Parallely, receiver P_i broadcasts "Accept" or "Reject", depending upon whether $dv_i + r_i = B(\alpha_i)$ or not.

Local Computation (by each player): IF $F(x)$ is broadcasted in **Round 3** then accept the lower order ℓ coefficients of $F(x)$ as $\mathbf{D}$'s secret and terminate. ELSE construct an n length bit vector V^{Sh}, where the $j^{th}, 1 \leq j \leq n$ bit is 1(0), if $P_j \in \mathcal{P}$ has broadcasted "Accept" ("Reject") during **Round 3**. The vector V^{Sh} is public, as it is constructed using broadcasted information. If V^{Sh} does not contain $n - t$ 1's, then discard $\mathbf{D}$ and terminate.

If $F(x)$ is not broadcasted during **Round 3**, then $(F(x), R(x))$ is called $\mathbf{D}$'s IC signature on S given to INT.

RevealVal($\mathbf{D}, INT, \mathcal{P}, S$): (a) **Round 1**: INT broadcasts $F(x), R(x)$; (b) **Round 2**: P_i broadcasts α_i, v_i and r_i.

Local Computation (by each player): For the polynomial $F(x)$ broadcasted by INT, construct an n length vector $V^{Rec}_{F(x)}$ whose j^{th} bit contains 1 if $v_j = USHash(\alpha_j; F)$, else 0. Similarly, construct the vector $V^{Rec}_{R(x)}$ corresponding to $R(x)$. Finally compute $V^{Rec}_{FR} = V^{Rec}_{F(x)} \otimes V^{Rec}_{R(x)}$, where $\otimes$ denotes bit wise AND. Since broadcasted information is public, each player (honest) will compute the same vectors $V^{Rec}_{F(x)}$ and $V^{Rec}_{R(x)}$ and hence V^{Rec}_{FR}. If V^{Rec}_{FR} and V^{Sh} matches at least at $t + 1$ locations (irrespective of bit value at these locations), then accept the lower order ℓ coefficients of $F(x)$ as S. In this case, we say that $\mathbf{D}$'s signature on S is correct. Else reject $F(x)$ broadcasted by INT and we say that INT has failed to produce $\mathbf{D}$'s signature.

PROOF: Communication complexity is easy. The properties of **IC** now follows from Lemma 11, Lemma 12 and Lemma 13 and working of the protocol. □

Comparison with the IC Protocol of [5]: In [5], the authors have given a four round IC protocol, which signs an $\ell = 1$ length secret with a communication overhead (both private and broadcast) of $O(n)$ field elements. So to generate an IC signature on $\ell > 1$ length secret, the protocol needs to be parallely executed ℓ times, resulting in a communication overhead of $O(n\ell k)$ bits. If ℓ is not constant, then clearly our three round IC protocol performs better than the four round IC protocol of [5].

8 Five Round UVSS with $n = 2t + 1$

We now design a five round $(2t+1, t)$ UVSS protocol to share an $\ell = 1$ length secret $s \in \mathbb{F}$. The protocol is somewhat inspired by the UVSS protocol of [5], which sequentially executes two set of IC protocols. This is followed by other consistency checks, which take three additional rounds. Since the IC protocol proposed in [5] takes four rounds, the UVSS protocol of [5] takes at most eleven rounds. The **Sharing Phase** of our five round UVSS protocol is presented in Table 8 and Table 9. In the protocol, there are two set of IC protocols, which are parallely executed. For the sake of clear presentation, the parallel steps of these two set of executions are separated into two separate columns.

In our protocol, we use the following definition:

Definition 6 *Let $P_i, P_j \in \mathcal{P}$ denote two players, where P_i is given the polynomials $f_i(x)$ and $g_i(y)$ and P_j is given the polynomials $f_j(x)$ and $g_j(y)$. Then P_i and P_j are said to be consistent with each other if $f_i(j) = g_j(i)$ and $f_j(i) = g_i(j)$. A vector $(e_1, e_2, \ldots, e_n) \in \mathbb{F}^n$ is t-consistent if there exists a polynomial $w(x)$ of degree at most t such that for $1 \leq i \leq n$, $w(i) = e_i$.*

Sharing Phase: Round I	
1. $\mathbf{D}$ chooses a random bivariate polynomial $F(x,y)$ of degree t in both variable, where $F(0,0) = s$. For $1 \leq i \leq n$, $\mathbf{D}$ computes $f_i(x) = F(x,i)$ and $g_i(y) = F(i,y)$. For $1 \leq j \leq n$, considering P_i as INT, $\mathbf{D}$ executes **Round 1** of $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, f_i(j))$ and $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, g_i(j))$, to give his IC signature on n shares of $f_i(x)$ and $g_i(y)$ to P_i.	2. For each pair (P_i, P_j), player P_i acting as a dealer, selects a random value $r_{ij} \in \mathbb{F}$. Treating P_j as INT, P_i executes **Round 1** of $\mathbf{IC}(P_i, P_j, \mathcal{P}, r_{ij})$. Here r_{ij} will be used by P_i and P_j to check the equality of $f_i(j)$ and $g_j(i)$, which should be common to both of them.
Sharing Phase: Round II	
1. In response to **Round 1** of $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, f_i(j))$ and $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, g_i(j))$, P_i acting as INT, executes **Round 2** of $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, f_i(j))$ and $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, g_i(j))$, for $1 \leq j \leq n$. Thus P_i tries to check the validity of $\mathbf{D}$'s signature on $f_i(j)$'s and $g_i(j)$'s as received during **Round I**.	2. In response to **Round 1** of $\mathbf{IC}(P_i, P_j, \mathcal{P}, r_{ij})$, player P_j, acting as INT, executes **Round 2** of $\mathbf{IC}(P_i, P_j, \mathcal{P}, r_{ij})$ to check the validity of P_i's signature on r_{ij}, received during **Round I**. 3. For $1 \leq i \leq n$, player P_i broadcasts: (a) $a_{ij} = f_i(j) + r_{ij}$, (b) $b_{ij} = g_i(j) + r_{ji}$.
Sharing Phase: Round III	
1. If $\mathbf{D}$ is not satisfied by the broadcast of P_i (as INT) in previous round (during the execution of **Round 2** of $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, f_i(j))$ and $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, g_i(j))$ for $1 \leq j \leq n$), then $\mathbf{D}$ broadcasts $f_i(x) = F(x,i)$ and $g_i(y) = F(i,y)$. This completes $\mathbf{D}$'s actions of **Sharing Phase**. If $\mathbf{D}$ has not broadcasted $f_i(x)$ and $g_i(y)$, then P_i has obtained a valid IC signature of $\mathbf{D}$ on the n shares of $f_i(x)$ and $g_i(y)$.	2. If P_i is dissatisfied by the broadcast of P_j in previous round (during execution of **Round 2** of $\mathbf{IC}(P_i, P_j, \mathcal{P}, r_{ij})$, then P_i broadcasts the signal $Unhappy_i^j$. We call (P_i, P_j) as an **accusing pair**. 3. If P_i finds that $f_i(j)$'s or $g_i(j)$'s, $1 \leq j \leq n$, which are given to it by $\mathbf{D}$ during **Round I** are not t-consistent, then P_i broadcasts $Complaint_i^{\mathbf{D}}$ signal. We call P_i as a **complaining** player.
Sharing Phase: Local computation by each player at the end of Round III:	
1. Form two sets $\mathbf{D}^B$ and $\mathbf{D}^{NB}$. Include P_i in $\mathbf{D}^B$ if $\mathbf{D}$ has broadcasted $f_i(x)$ and $g_i(y)$ during **Round III**, otherwise P_i is included in $\mathbf{D}^{NB}$. If $\lvert\mathbf{D}^B\rvert > t$, then discard $\mathbf{D}$ and terminate. 2. For every $(P_i, P_j) \in \mathbf{D}^B$, check if they are consistent (see Definition 6) with respect to the polynomials corresponding to them, which $\mathbf{D}$ has broadcasted during **Round III**. If not, then discard $\mathbf{D}$ and terminate. 3. If $P_i \in \mathbf{D}^B$, then $f_i(j)$'s and $g_i(j)$'s are known publicly. So terminate and ignore the execution of $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, f_i(j))$, $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, g_i(j))$, $\mathbf{IC}(P_i, P_j, \mathcal{P}, r_{ij})$ and $\mathbf{IC}(P_j, P_i, \mathcal{P}, r_{ji})$, for $1 \leq j \leq n$.	
Sharing Phase: Round IV	
1. If $P_i, P_j \in \mathbf{D}^{NB}$ and (P_i, P_j) is an **accusing pair**, then P_i and P_j defends themselves by broadcasting $f_i(j), g_i(j)$ and $f_j(i), g_j(i)$ respectively, along with $\mathbf{D}$'s signature on them. 2. If $P_i, P_j \in \mathbf{D}^{NB}$ and if $a_{ij} \neq b_{ji}$ or $a_{ji} \neq b_{ij}$ (during **Round II**), then P_i, P_j do the same actions as in the above step. In this case, we call pair (P_i, P_j) as **conflicting pair**. 3. If $P_i \in \mathbf{D}^{NB}$ is a **complaining** player, then P_i defends himself by broadcasting the values $f_i(j)$'s and $g_i(j)$'s, $1 \leq j \leq n$, which it has received from $\mathbf{D}$ during **Round I**, along with $\mathbf{D}$'s signature on these values. 4. Corresponding to each $P_i \in \mathbf{D}^B$, player $P_j \in \mathbf{D}^{NB}$ broadcasts $g_j(i)$ and $f_j(i)$ (which P_j has received during **Round I**), along with $\mathbf{D}$'s signature on these values. We call this broadcast as $P_i - P_j - B - NB - Consistency - Checking - Broadcast$. This broadcast is done to ensure whether the players in $\mathbf{D}^{NB}$ are consistent with the players in $\mathbf{D}^B$. However this does not hamper the secrecy of the protocol.	
Sharing Phase: Round V	
The receivers in $\mathcal{P}$ broadcasts **verification information** corresponding to the signatures which are produced during step 1, 2, 3 and 4 of previous round.	

Table 8: Sharing phase of five round UVSS with $n = 2t + 1$ to share a secret value $s \in \mathbb{F}$.

Claim 7 *If $\lvert\mathbf{D}^B\rvert > t$, then $\mathbf{D}$ is corrupted.*

PROOF: P_i is included in $\mathbf{D}^B$, if $\mathbf{D}$ is not satisfied with the values broadcasted by P_i during the execution

Local Computation by Each Player at the End of Round V (If D is not discarded)

1. For each **accusing** or **conflicting pair** (P_i, P_j), such that $P_i, P_j \in \mathbf{D}^{NB}$, do the following:

   (a) Check the validity of $\mathbf{D}$'s signature on $f_i(j), g_i(j), f_j(i)$ and $g_j(i)$, which P_i and P_j has produced during **Round IV**. If $\mathbf{D}$'s signature on $f_i(j)$ or $g_i(j)$ is found to be invalid, then discard P_i from $\mathbf{D}^{NB}$. Similarly, if $\mathbf{D}$'s signature on $f_j(i)$ or $g_j(i)$ is found to be invalid, then discard P_j from $\mathbf{D}^{NB}$.

   (b) If both P_i and P_j are not discarded during previous step but either $f_i(j) \neq g_j(i)$ or $f_j(i) \neq g_i(j)$, then discard $\mathbf{D}$ and terminate (**see claim 9**)). Else, publicly accept $g_i(j)$ and $g_j(i)$ as the j^{th} and i^{th} share of $g_i(y)$ and $g_j(y)$ respectively **(see Theorem 8)**.

2. If P_i is not discarded from $\mathbf{D}^{NB}$ and is a **complaining player**, then check if the values $f_i(j)$'s and $g_i(j)$'s, $1 \leq j \leq n$, produced by P_i during **Round IV** have got $\mathbf{D}$'s valid signature on them. If not, then discard P_i from $\mathbf{D}^{NB}$. Else, check if the values $f_i(j)$'s and $g_i(j)$'s, $1 \leq j \leq n$ are t-consistent. If either $f_i(j)$'s or $g_i(j)$'s are not t-consistent, then discard $\mathbf{D}$ and terminate (**see Claim 10**). Otherwise discard P_i from $\mathbf{D}^{NB}$.

3. If P_j is not discarded from $\mathbf{D}^{NB}$, then corresponding to $P_i - P_j - B - NB - Consistency - Checking - Broadcast$, do the following: (a) Check if the values broadcasted by P_j (namely $f_j(i)$ and $g_j(i)$) have got $\mathbf{D}$'s valid signature on them. If not, then discard P_j from $\mathbf{D}^{NB}$; (b) If signatures are valid, then check whether P_j is consistent with P_i (w.r.t $f_i(x)$ and $g_i(y)$ broadcasted by $\mathbf{D}$, during **Round III**). In case of any inconsistency, discard $\mathbf{D}$ and terminate (**see Claim 11**). Otherwise, publicly accept $g_j(i)$ as the i^{th} share of $g_j(y)$.

4. If the size of final $\mathbf{D}^{NB}$ is less than $t+1$, then discard $\mathbf{D}$ and terminate the protocol.

Table 9: Sharing Phase of five round UVSS with $n = 2t+1$ to share a secret value $s \in \mathbb{F}$ Contd ...

of **Round 2** of any of the protocols $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, f_i(j))$ or $\mathbf{IC}(\mathbf{D}, P_i, \mathcal{P}, g_i(j))$, for $1 \leq j \leq n$. If both $\mathbf{D}$ and P_i are honest, then P_i will be never included in $\mathbf{D}^B$. So, if $P_i \in \mathbf{D}^B$, then either $\mathbf{D}$ or P_i is corrupted. So for an honest $\mathbf{D}$, $\mathbf{D}^B$ is always less than $t+1$. □

Claim 8 *If there exists $P_i, P_j \in \mathbf{D}^B$ who are not consistent with each other with respect to their corresponding $f(x)$ and $g(x)$ polynomials which are broadcasted by $\mathbf{D}$, then $\mathbf{D}$ is corrupted.*

PROOF: Follows from the fact that if $\mathbf{D}$ is honest then only corrupted players are included in $\mathbf{D}^B$ and $\mathbf{D}$ would have broadcasted $f_i(x) = F(x,i), g_i(y) = F(i,y), f_j(x) = F(x,j)$ and $g_j(y) = F(j,y)$, corresponding to $P_i, P_j \in \mathbf{D}^B$, where $F(x,y)$ is the original bivariate polynomial. □

Claim 9 *Suppose at the end of* **Round IV** *of* **Sharing Phase**, *there exists a* **conflicting** *or* **accusing pair** (P_i, P_j), *such that $P_i, P_j \in \mathbf{D}^{NB}$. Moreover the values $f_i(j), g_i(j)$ and $f_j(i), g_j(i)$ produced by P_i and P_j respectively have got $\mathbf{D}$'s valid signature on them. Furthermore, either $f_i(j) \neq g_j(i)$ or $f_j(i) \neq g_i(j)$. Then except with probability at most $2^{-O(k)}$, $\mathbf{D}$ is corrupted.*

PROOF: If $\mathbf{D}$ is honest and (P_i, P_j) is an **accusing** or **conflicting pair**, then at least one of the P_i or P_j is corrupted. Let P_i be corrupted. Then at least one of the values $f_i(j)$ or $g_i(j)$, produced by P_i during **Round IV** will be different from the actual $f_i(j)$ or $g_i(j)$, which $\mathbf{D}$ had given to P_i during **Round I**. Let $f_i(j)$ be incorrect. However, P_i has to produce $\mathbf{D}$'s IC signature on the incorrect $f_i(j)$. But from the property of **IC** protocol (see Lemma 12), corrupted P_i cannot forge honest $\mathbf{D}$'s signature on incorrect $f_i(j)$, except with an error probability of at most $2^{-O(k)}$. Thus, if either $f_i(j) \neq g_j(i)$ or $f_j(i) \neq g_i(j)$ and if $\mathbf{D}$'s signature on these values are valid, then except with error probability of at most $2^{-O(k)}$, $\mathbf{D}$ is corrupted. □

Claim 10 *If there exists a* **complaining player** $P_i \in \mathbf{D}^{NB}$, *such that the values $f_i(j)$'s or $g_i(j)$'s, $1 \leq j \leq n$ produced by P_i are not t-consistent and have got valid signature of $\mathbf{D}$ on them, then except with an error probability of at most $2^{-O(k)}$, $\mathbf{D}$ is corrupted.*

PROOF: If $\mathbf{D}$ is honest then an honest P_i can never be a **complaining player**. However, a corrupted $P_i \in \mathbf{D}^{NB}$ can become a **complaining player** and may produce t-inconsistent $f_i(j)$'s or $g_i(j)$'s (which

are not given to him by $\mathbf{D}$) and can forge $\mathbf{D}$'s signature on these values. But from Lemma 12, this can happen with probability at most $2^{-O(k)}$. □

Claim 11 *Let $P_i \in \mathbf{D}^B$ and $P_j \in \mathbf{D}^{NB}$, where $\mathbf{D}$ has broadcasted $F(x,i)$ and $F(i,y)$, corresponding to P_i during* **Round III**. *Suppose P_j has broadcasted $f_j(i)$ and $g_j(i)$ during* **Round IV** *in $P_i - P_j - B - NB - Consistency - Checking - Broadcast$, such that both $f_j(i)$ and $g_j(i)$ has got $\mathbf{D}$'s valid signature on it. Moreover, P_i and P_j are inconsistent with each other, i.e., either $F(j,i) \neq g_j(i)$ or $F(i,j) \neq f_j(i)$. Then except with error probability of at most $2^{-O(k)}$, $\mathbf{D}$ is corrupted.*

Proof: If $\mathbf{D}$ is honest then only corrupted players are included in $\mathbf{D}^B$ and all the honest players are present in $\mathbf{D}^{NB}$. So the values broadcasted by an honest $P_j \in \mathbf{D}^{NB}$ during $P_i - P_j - B - NB - Consistency - Checking - Broadcast$, corresponding to $P_i \in \mathbf{D}^B$ will always be consistent with P_i. However a corrupted $P_j \in \mathbf{D}^{NB}$ may broadcast either incorrect $f_j(i)$ or $g_j(i)$ during $P_i - P_j - NB - Consistency - Checking - Broadcast$, along with valid $\mathbf{D}$'s signature on them, such that either $F(j,i) \neq g_j(i)$ or $F(i,j) \neq f_j(i)$. But, according to the property of **IC** protocol, a corrupted P_j cannot do so, except with an error probability of at most $2^{-O(k)}$. □

Claim 12 *At the end of* **Sharing Phase**, *if the size of final $\mathbf{D}^{NB} < t+1$, then $\mathbf{D}$ is corrupted.*

Proof: The proof follows from the fact that if $\mathbf{D}$ is honest then all the honest players (at least $t+1$) will be present in $\mathbf{D}^{NB}$ and no honest player in $\mathbf{D}^{NB}$ is removed at the end of **Round IV**. □

We now enumerate all possible events under which an honest $\mathbf{D}$ can be discarded and show that none can occur except with an error probability of at most $2^{-O(k)}$.

Lemma 14 *An honest $\mathbf{D}$ can be discarded during* **Sharing Phase** *only with an error probability of at most $2^{-O(k)}$.*

Proof: It is easy to see that if $\mathbf{D}$ is honest then $|\mathbf{D}^B| \leq t$. More each $P_i, P_j \in \mathbf{D}^B$ will be consistent with each other. Moreover, the size of final $\mathbf{D}^{NB}$ will be at least $t+1$. Now an honest $\mathbf{D}$ can be discarded during **Sharing Phase**, only if one of the following events occur:

1. *At the end of* **Round IV**, *there exists a* **conflicting pair** *or an* **accusing pair** *(P_i, P_j), where $P_i, P_j \in \mathbf{D}^{NB}$. Moreover, the values $f_i(j), g_i(j)$ and $f_j(i), g_j(i)$, as produced by P_i and P_j respectively, have got $\mathbf{D}$'s valid signature and either $(f_i(j) \neq g_j(i))$ or $(f_j(i) \neq g_i(j))$*: From Claim 9, this can happen for an honest $\mathbf{D}$ with an error probability of at most $2^{-O(k)}$. Since there can be $O(n^2)$ such pairs, the total error probability is $O(n^2)2^{-O(k)} \approx 2^{-O(k)}$.

2. *At the end of* **Round IV**, *there exists a* **complaining player** *$P_i \in \mathbf{D}^{NB}$, such that the values $f_i(j)$'s or $g_i(j)$'s, $1 \leq j \leq n$ produced by P_i are not t-consistent and have got valid signature of $\mathbf{D}$ on these values*: From Claim 10, this can happen for an honest $\mathbf{D}$ with an error probability of at most $2^{-O(k)}$. Since here can be $O(n)$ such corrupted players, the total error probability is $O(n)2^{-O(k)} \approx 2^{-O(k)}$.

3. *At the end of* **Round IV**, *there exists a pair (P_i, P_j), where $P_i \in \mathbf{D}^B$ and $P_j \in \mathbf{D}^{NB}$, such that the values broadcasted by P_j during $P_i - P_j - B - NB - Consistency - Checking - Broadcast$ are inconsistent with P_i and have got valid signature of $\mathbf{D}$ on them*: From Claim 11, this can happen for an honest $\mathbf{D}$ with an error probability of at most $2^{-O(k)}$. Since there can be $O(n^2)$ such pairs, the total error probability is $O(n^2)2^{-O(k)} \approx 2^{-O(k)}$. □

Next we enumerate all possible events under which an honest player can be discarded during **Sharing Phase** and show that none can occur except with an error probability of at most $2^{-O(k)}$.

Lemma 15 *An honest player P_j can be discarded during* **Sharing Phase** *only with an error probability of at most $2^{-O(k)}$.*

PROOF: If $P_j \in \mathbf{D}^B$, then P_j cannot be discarded. So, we have to consider the case where $P_j \in \mathbf{D}^{NB}$. From the protocol, player $P_j \in \mathbf{D}^{NB}$ will be discarded only if one of the following events occur:

1. *There exists a* **conflicting** *or* **accusing pair** (P_j, P_i)*, where $P_i \in \mathbf{D}^{NB}$, such that* $\mathbf{D}$*'s signature on $f_j(i)$ or $g_j(i)$, as produced by P_j is invalid*: If $\mathbf{D}$ is honest, then this will never happen because from the property of **IC** protocol, an honest P_j will always be able to produce valid signature of an honest $\mathbf{D}$ on $f_j(i)$ and $g_j(i)$. However, if $\mathbf{D}$ is corrupted, then the honest P_j will be able to produce valid signature of a dishonest $\mathbf{D}$ on $f_j(i)$ and $g_j(i)$ with probability at least $1 - 2^{-O(k)}$ (see Lemma 11). But, in the later case, P_j can be discarded, but this happens with an error probability of at most $2^{-O(k)}$.

2. P_j *is a* **complaining player**, *such that* $\mathbf{D}$*'s signature on at least one of $f_j(i)$'s or $g_j(i)$'s, as produced by P_j during* **Round IV** *fails*: Since P_j is an honest as well as a **complaining player**, indeed he has received either t-inconsistent $f_j(i)$'s or t-inconsistent $g_j(i)$'s during **Round I** from $\mathbf{D}$. So as a proof, P_j broadcasts these inconsistent values, along with $\mathbf{D}$'s signature on them. By the properties of **IC** protocol, except with an error probability of at most $2^{-O(k)}$, $\mathbf{D}$'s signature on these values are valid and will be accepted. However, with an error probability of at most $2^{-O(k)}$, the signature may fail and P_j may be discarded.

3. *The values broadcasted by P_j (namely $f_j(i)$ and $g_j(i)$) during $P_i - P_j - B - NB - Consistency - Checking - Broadcast$ corresponding to some $P_i \in \mathbf{D}^B$ has got* $\mathbf{D}$*'s invalid signature on them*: If $\mathbf{D}$ is honest, then this never happens for an honest P_j. However, if $\mathbf{D}$ is corrupted, then from the properties of **IC** protocol, this can happen with error probability of at most $2^{-O(k)}$. □

Let $\mathbf{D'}^{NB}$ denotes the set of players in $\mathbf{D}^{NB}$, who are not discarded at the end of **Sharing Phase**. If $\mathbf{D}$ is not discarded, then the properties given in Theorem 8 are true.

Theorem 8 *If* $\mathbf{D}$ *is not discarded during* **Sharing Phase**, *then the following holds:*

Property 1. Except with an error probability of at most 2^{-k}, no honest player is discarded.

Property 2. All the players in $\mathbf{D}^B$ are be consistent with each other.

Property 3. There exists at least one honest player in $\mathbf{D'}^{NB}$.

Property 4. Each honest $P_i \in \mathbf{D'}^{NB}$ have t-consistent $f_i(j)$'s and $g_i(j)$'s, $1 \leq j \leq n$.

Property 5. All honest players in $\mathbf{D'}^{NB}$ are consistent with each other. Moreover, each honest player in $\mathbf{D'}^{NB}$ is consistent with all the players in $\mathbf{D}^B$.

Property 6. Corresponding to each **conflicting** *or* **accusing pair** (P_i, P_j)*, where $P_i, P_j \in \mathbf{D'}^{NB}$, the shares $g_i(j)$ and $g_j(i)$ are known publicly.*

Property 7. Every corrupted player $P_i \in \mathbf{D'}^{NB}$ commits $g_i(j)$ to honest player $P_j \in \mathbf{D'}^{NB}$ (by agreeing with $f_j(i)$).

Property 8. Every corrupted player $P_i \in \mathbf{D'}^{NB}$ commits $g_i(k)$ publicly by agreeing with $f_k(x)$, where $P_k \in \mathbf{D}^B$ and $f_k(x)$ is broadcasted by $\mathbf{D}$ *during* **Round III**.

PROOF: The proof follows from the proof of the all the previous claims and working of the protocol. □

If **D** is not discarded during **Sharing Phase**, the protocol proceeds to **Reconstruction Phase** as shown in Table 10. Before explaining the **Reconstruction Phase**, we first list out the values which are known *publicly* at the end of **Sharing Phase** and are going to be directly used during **Reconstruction Phase**.

1. The polynomials $f_i(x) = F(x, i)$ and $g_i(y) = F(i, y)$, corresponding to each $P_i \in \mathbf{D}^B$.
2. For $P_j \in \mathbf{D'}^{NB}$ and $P_i \in \mathbf{D}^B$, the share $g_j(i)$ (**see step 3(b) during local computation at the end of Round IV**).
3. If $P_i, P_j \in \mathbf{D'}^{NB}$, but the pair (P_i, P_j) was either an **accusing** or **conflicting pair** during **Round III**, then the shares $g_i(j)$ and $g_j(i)$ (**see step 1(b) during local computation at the end of Round IV**).

Reconstruction Phase (Two Rounds): Only the players from the set $\mathbf{D}^B$ and $\mathbf{D'}^{NB}$ participate, where $\mathbf{D'}^{NB}$ denotes the set of players in $\mathbf{D}^{NB}$ who are not discarded during **Sharing Phase**. Set $CORE = \mathbf{D'}^{NB}$.

1. Each $P_i \in CORE$ broadcasts P_j's signature on r_{ji} received from P_j during **Round I**, provided $P_j \in CORE$ and the share $g_i(j)$ is not known publicly. In the **next round**, the receivers in $\mathcal{P}$ broadcasts **verification information** corresponding to r_{ji}. Each player locally verifies the signature. If the signature produced by P_i fails for even one such r_{ji}, then discard P_i from $CORE$. Else each player locally tries to recover the n shares of $g_i(y)$, denoted by $g_{ij}, 1 \leq j \leq n$ as follows:

$$\begin{aligned} g_{ij} &= f_j(i) \quad \text{if } P_j \in \mathbf{D}^B \\ &= g_i(j) \quad \text{if } P_j \in \mathbf{D'}^{NB} \text{ and } P_i, P_j \text{ were involved in either an } \textbf{accusing} \text{ or } \textbf{conflicting pair} \\ &= b_{ij} - r_{ji} \quad \text{where } b_{ij} \text{ was broadcasted by } P_i \text{ during } \textbf{Round II} \end{aligned}$$

Remove P_i from $CORE$, if g_{ij}'s are not t-consistent. Otherwise reconstruct $g_i(y)$ by interpolating g_{ij}'s.

2. Take the recovered $g_i(y)$'s (from the players of $CORE$), along with the $g_i(y)$'s corresponding to the players in $\mathbf{D}^B$. Using them, interpolate $F^H(x, y)$, reconstruct $s' = F^H(0, 0)$ and terminate.

Table 10: Reconstruction Phase of Five Round UVSS with $n = 2t + 1$.

We now prove the properties of the five round UVSS protocol.

Lemma 16 *The five round UVSS protocol satisfies perfect secrecy.*

PROOF: We have to only consider the case when **D** is honest. If **D** is honest then $\mathbf{D}^B$ will contain only corrupted players. So the polynomials corresponding to them which are broadcasted by **D** gives no new information to the adversary. The r_{ij}'s exchanged between honest P_i, P_j are completely random and unknown to the adversary. Correspondingly, the blinded common shares broadcasted by P_i and P_j will give no information about their common shares to the adversary. The proof now follows from the properties of a bivariate polynomial of degree t and secrecy of **IC** protocol (see Lemma 13). □

Lemma 17 *The UVSS protocol satisfies correctness property except with error probability of* $2^{-O(k)}$.

PROOF: We have to only consider the case when **D** is honest. From Lemma 14, the probability that an honest **D** might get discarded during sharing phase is at most $2^{-O(k)}$. When **D** is honest, all the honest players (at least $t+1$) will be present in $\mathbf{D'}^{NB}$ (and hence in $CORE$) and will be consistent with each other and with the original bivariate polynomial $F(x, y)$. Moreover, only corrupted players will be present in $\mathbf{D}^B$ and the $f(x), g(x)$ polynomials corresponding to these players (which are broadcasted by **D** during **Round III**) will be consistent with $F(x, y)$. Now consider a corrupted player $P_i \in CORE$.

During reconstruction phase, P_i has to produce the signature of each $P_j \in CORE$ on the random r_{ji}, which P_i has received from P_j during **Round II**. Now except with an error probability of at most $2^{-O(k)}$, P_i cannot forge an honest P_j's signature on incorrect r_{ji}. Moreover, from Property 7 of Theorem 8, P_i has committed $g_i(j)$ by agreeing with $f_j(i)$. Also, from Property 6 and Property 8 of Theorem 8, the publicly known shares of $g_i(y)$ are consistent with $F(x, y)$. So if during reconstruction phase, the recovered g_{ij}'s are t-consistent then it implies that except with an error probability of at most $2^{-O(k)}$, it is consistent with $F(x, y)$ also. Hence the lemma holds. □

Lemma 18 *The five round UVSS protocol satisfies strong commitment property except with an error probability of at most* 2^{-k}.

PROOF: We have to only consider the case when **D** is dishonest. If **D** is discarded during sharing phase, then the lemma holds. On the other hand if **D** is not discarded, then from Lemma 15, except with an error probability of 2^{-k}, none of the honest players (at least $t + 1$) are discarded. Since **D** is corrupted, the honest players may be distributed in sets $\mathbf{D}^B$ and $\mathbf{D'}^{NB}$. However, from Property 5 of Theorem 8, all honest players, along with the players in $\mathbf{D}^B$ are consistent with each other and hence define a unique bivariate polynomial $F^H(x, y)$ of degree at most t in both x and y. Moreover, from the properties given in Theorem 8, each corrupted player (either in $\mathbf{D}^B$ or in $\mathbf{D'}^{NB}$) is consistent with all the honest players, who in turn are consistent with $F^H(x, y)$. So if a corrupted $P_i \in \mathbf{D'}^{NB}$ is not discarded in the reconstruction phase, then the recovered $g_i(y)$ will be consistent with $F^H(x, y)$. Hence the strong commitment on $s' = F^H(0, 0)$ is satisfied. □

Lemma 19 *The five round UVSS protocol communicates* $O(n^3k)$ *bits and broadcasts* $O(n^3k)$ *bits.*

PROOF: In the protocol, **D** executes $2n^2$ instances of **IC** protocol to give its signature on the n shares of $f_i(x)$ and $g_i(y), 1 \leq i \leq n$. Similarly, each P_i executes n instances of **IC** protocol to give its signature on r_{ij}'s to P_j's. So total number of **IC** protocols executed by the players (as a dealer) is n^2. Thus, the total number of **IC** protocol executed in the UVSS protocol is $3n^2$, where in each execution, $\ell = 1$ length secret is signed. The lemma now follows from Theorem 7). □

Theorem 9 *The four round UVSS protocol satisfies the properties of UVSS with an error probability of at most* $2^{-O(k)}$.

PROOF: The proof follows from Lemma 16, Lemma 17 and Lemma 18. □

8.1 Five Round UVSS to Share $\ell > 1$ Length Secret

We now informally show how to adapt the above protocol to share the secret $S = [s_1\ s_2\ \ldots\ s_\ell] \in \mathbb{F}^\ell$, where $\ell > 1$. **D** generates ℓ random bivariate polynomials $F^k(x, y), 1 \leq k \leq \ell$, each of degree t in both x, y, such that $F^k(0, 0) = s_k$. Let $f_i^k(x) = F^k(x, i)$ and $g_i^k(y) = F^k(i, y)$. **D** gives its IC signature on shares of $f_i^k(x)$ and $g_i^k(y)$ to player P_i. Recall that **IC** protocol can be used to generate IC signature of a player on $\ell 1$ length secret in a single execution. Hence, **D** can give its IC signature on the shares of $f_i^k(x)$ and $g_i^k(y)$ to P_i by executing $2n$ instance of **IC**, where in each instance, it signs on an ℓ length message. Now each pair of distinct players (P_i, P_j) will have 2ℓ shares in common. Player P_i (P_j), in order to check the consistency of common shares with P_j (P_i), will give ℓ random values to P_j (P_i), along with its IC signature on these values. To generate the signatures, P_i (P_j) will execute a single instance of **IC** protocol to sign on ℓ length message. The protocol now proceeds as in the above protocol. All the claims, lemmas and theorems of previous protocol will hold here. It is easy to see that $3n^2$ instances of **IC** will be executed, where in each instance, an ℓ length message (secret) is signed. So, we get the following theorem:

Theorem 10 *There exists a five round $(2t+1, t)$ UVSS scheme with agreement, which shares an $\ell \geq 1$ length secret by communicating (both private and broadcast) $O(n^2(\ell+n)k)$ bits.*

Comparison with the UVSS Protocol of [5]: In [5], the authors have given a $(2t+1, t)$ UVSS protocol, whose sharing phase takes at most eleven rounds. Moreover, the protocol shares a single length secret; i.e., $\ell = 1$ by communicating and broadcasting $O(n^3k)$ bits. The protocol needs to be executed ℓ times to share ℓ length secret, incurring a communication overhead (both private and broadcast) of $O(n^3\ell k)$ bits. Comparing this with Theorem 10, we find that our UVSS protocol performs better than the UVSS protocol of [5], both in terms of communication and round complexity.

9 Lower Bound on Single Round UWSS

In this section, we prove that any single round UWSS is possible only if $n > 3t$.

Theorem 11 *There is no single round (n, t)-UWSS protocol when $n \leq 3t$.*

PROOF: By player-partitioning argument [7, 8], Theorem 11 reduced to the following lemma.

Lemma 20 *There is no single round $(3, 1)$-UWSS protocol.*

PROOF (SKETCH): Let Π be a $(3, 1)$-UWSS protocol with players P_1, P_2, P_3, with P_1 as dealer ($\mathbf{D}$). The execution of Π can be viewed as follows: (a) **Sharing Phase: D**, on input secret s and random input r_D, sends α, β, γ to P_1, P_2 and P_3 respectively and broadcasts b_D. Each other player $P_i, i \in \{2, 3\}$, on random input r_i, sends a message p_{ij} to each player P_j and broadcasts b_i; (b) **Reconstruction Phase:** Every player produces it's entire view generated in sharing phase.

In Π, the broadcasts done by dealer and individual players have no information about the secret s, otherwise Π violates the secrecy property of UWSS. The secrecy property also implies that when $\mathbf{D}$ is honest, any one of α, β and γ must not have any information about s. According to the correctness property of Π, when $\mathbf{D}$ is honest, if either P_2 or P_3 deviates from the protocol during reconstruction phase, then all the honest players must output s with very high probability.

Let s_1 and s_2 be two independent secrets and $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$ be the share corresponding to s_1 and s_2 respectively. Consider two execution E_1^h and E_2^h of Π, where $\mathbf{D}$ is honest. In E_1^h, secret is s_1 and $\mathbf{D}$ distributes α_1, β_1 and γ_1 to P_1, P_2 and P_3 respectively. Assume that in E_1^h, P_2 is corrupted and further assume that P_2 produces β_2 in reconstruction phase. So according to correctness property of Π, each honest player should reconstruct s_1 with very high probability.

In E_2^h, secret is s_2 and $\mathbf{D}$ distributes α_2, β_2 and γ_2 to P_1, P_2 and P_3 respectively. Assume that in E_2^h, P_3 is corrupted and further assume that P_3 produces γ_1 in reconstruction phase. So according to correctness property of Π, each honest player should reconstruct s_2 with very high probability.

Now consider another execution E_3^c of Π where $\mathbf{D}$ is corrupted and distributes α_1, β_2 and γ_1 to P_1, P_2 and P_3 respectively. Now in reconstruction phase if every player behaves honestly, then view of the honest players in the reconstruction phase of E_3^c will exactly match with the view of the honest players in the reconstruction phase of E_1^h. Since the honest players reconstructs s_1 in E_1^h, they do the same in E_3^c also. Now according to the weak commitment property, if in E_3^c, the corrupted player (which is $\mathbf{D} = P_1$) deviates from the protocol and broadcasts α_2 during reconstruction phase, then with very high probability all honest players must reconstruct either s_1 or $NULL$. But notice that now, the view of the honest players will be identical as in E_2^h and thus s_2 should be reconstructed with very high probability. This is a contradiction. Hence Π does not exist. Thus there is no single round $(3, 1)$-UWSS and hence single round $(3t, t)$ UWSS protocol. □

10 Lower Bound on Single Round UVSS

Theorem 12 *There is no one round UVSS protocol with ($t = 1$ and $n < 4$) or with $t > 1$.*

PROOF: From Theorem 11, we know that there is no single round UWSS protocol for $n \leq 3$. Since UVSS has stronger properties than UWSS, the above implication holds for UVSS. Now we prove that there is no single round UVSS protocol for $t > 1$ and $n \geq 4$. For that, we show that if there exist a single round $(n, 2)$ UVSS protocol with $n \geq 4$, then its error probability P_{error} must satisfy $P_{error} \geq \frac{1}{n}$. But according to the definition of UVSS, P_{error} should be exponentially small. So this shows a contradiction. Let Π be a single round $(n, 2)$ UVSS, where P_1 is the dealer. According to secrecy property of Π, for an honest $\mathbf{D}$, the broadcasts done by $\mathbf{D}$ during sharing phase is independent of the secret. Since for an honest $\mathbf{D}$, other players do not know the secret in sharing phase (which has a single round), the broadcasts done by individual players and the private communications done between any two players from $\mathcal{P} - \{\mathbf{D}\}$ during sharing phase are completely independent of the secret. Hence, we can neglect all of the above mentioned communications and concentrate on the communications done by $\mathbf{D}$ to the players in $\mathcal{P}$. Since Π is single round UVSS, the values given by $\mathbf{D}$ to individual players can be thought as shares of $\mathbf{D}$'s secret s. Now, let s and s^* be two secrets where $(\beta_1, \beta_2, \ldots, \beta_n)$ and $(\theta_1, \theta_2, \ldots, \theta_n)$ be the n shares corresponding to s and s^*, respectively. Let in Π, P_1 $(= \mathbf{D})$ and P_2 be the two corrupted players. Now consider three different type of executions of Π. In each execution, the random coin tosses of all the players are same.

1. In execution type E_i^A, $0 \leq i \leq n-2$, during sharing phase, $\mathbf{D}$ gives the shares corresponding to the secret s^* to $P_2, P_3, \ldots, P_{2+i}$. To the remaining players, $\mathbf{D}$ gives the shares corresponding to s. During reconstruction phase, each player behaves honestly and correctly produces the shares.

2. In execution type $E_i^B, 0 \leq i \leq n-3$, during sharing phase, $\mathbf{D}$ gives the shares corresponding to s^* to $P_3, P_4, \ldots, P_{3+i}$. To the remaining players, $\mathbf{D}$ gives the shares corresponding to s. During reconstruction phase, each player behaves honestly and correctly produces the shares.

3. In execution type $E_i^C, 0 \leq i \leq n-3$, the sharing phase is same as in E_i^B, but during reconstruction phase, P_2 (corrupted player) produces share corresponding to s^*. This he can do because $\mathbf{D}$ is also corrupted and hence can collude with P_2.

For pictorial representation of these three different types of executions see Figure 1. Let $P(s, E)$ be the probability that secret s is reconstructed during reconstruction phase of an execution E. Notice that execution E_i^A and E_i^B are same in the sense that in both the executions, during sharing phase, $\mathbf{D}$ distributes $i + 1$ shares corresponding to s^* and $n - i - 1$ shares corresponding to s and during reconstruction phase, each player honestly broadcast the shares received during sharing phase. Hence

$$P(s, E_i^A) = P(s, E_i^B) \tag{7}$$

Next notice that E_i^B and E_i^C differs only in the behavior of faulty player (P_2) during reconstruction phase. So according to the strong commitment property of Π, if s can be reconstructed in E_i^B with probability p, then s should also be reconstructed with probability at least $(1 - P_{error}) \times p$ in E_i^C (from Baye's Theorem and neglecting the other terms which are positive). This implies that

$$P(s, E_i^C) \geq (1 - P_{error}) \times P(s, E_i^B) \tag{8}$$

Finally in E_i^C and E_{i+1}^A, the view of the honest players during reconstruction phase is same. Hence

$$P(s, E_{i+1}^A) = P(s, E_i^C) \tag{9}$$

$$\text{Now by correctness property of } \Pi, P(s, E_0^A) \geq 1 - P_{error} \tag{10}$$

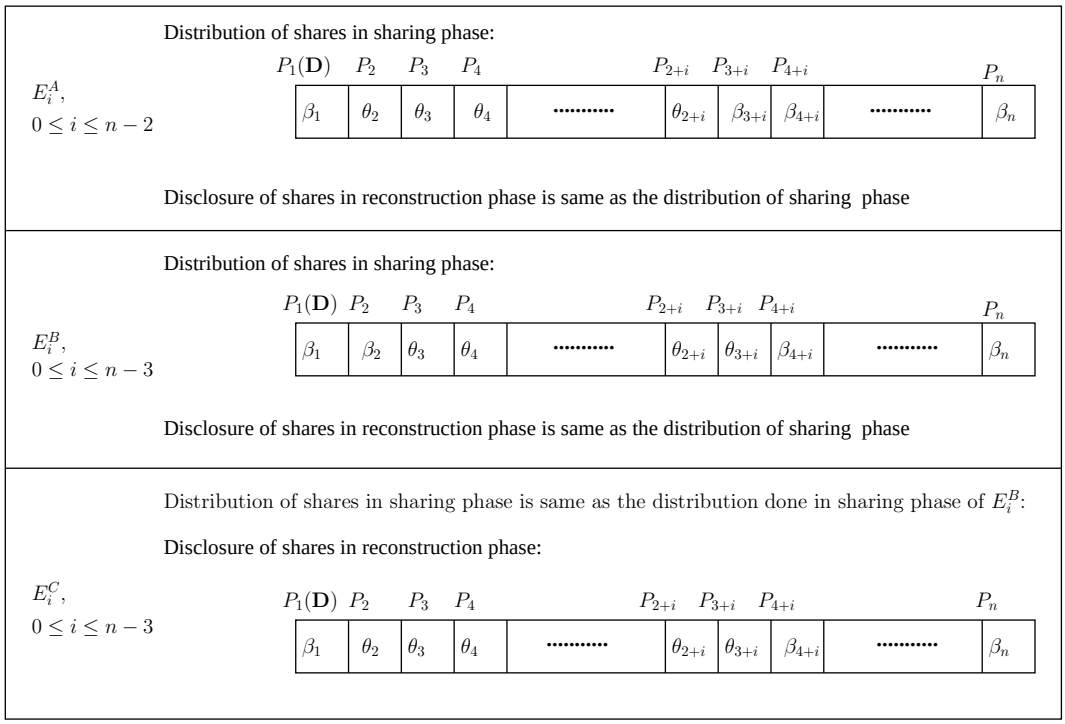

Figure 1: Pictorial Representation of three differerent type of executions E_i^A,E_i^B and E_i^C

$$\begin{aligned} \text{Now From Equation (9) ,} \quad P(s, E_1^A) &= P(s, E_0^C) \\ &\geq (1 - P_{error}) \times P(s, E_0^B) \text{ by (8)} \\ &= (1 - P_{error}) \times P(s, E_0^A) \text{ by (7)} \\ &\geq (1 - P_{error})^2 \text{ by (10)} \end{aligned} \tag{11}$$

Hence by induction, $P(s, E_{n-2}^A) \geq (1-P_{error})^{n-1}$. However, E_{n-2}^A denotes an execution sequence, where during sharing phase, $\mathbf{D}$ has distributed $n-1$ shares corresponding to s^* and one share corresponding to s. Moreover, during reconstruction phase, all players honestly broadcast the shares received during sharing phase. From the correctness and commitment property of Π, we get

$$P(s^*, E_{n-2}^A) \geq (1 - P_{error}) \tag{12}$$

Notice that

$$\begin{aligned} 1 &\geq P(s, E_{n-2}^A) + P(s^*, E_{n-2}^A) \\ &\geq (1 - P_{error})^{n-1} + (1 - P_{error}) \\ &\geq 1 - (n-1) \times P_{error} + 1 - P_{error} \end{aligned}$$

This implies that $P_{error} \geq \frac{1}{n}$. But this is a contradiction because according to the definition of PVSS, P_{error} is exponentially small. Hence Π does not exist. □

11 Conclusion and Open Problems

In this work, we have shown that probabilistically relaxing the conditions of VSS and WSS helps to increases the the fault tolerance significantly. The following are the challenging problems left open in this paper: (a) Is $n > 3t$ necessary for two round UVSS and two round UWSS? (we have proved only sufficiency) (b) Is $n > 2t$ is sufficient for four round UVSS? (necessity is obvious from [10]). We conjecture that it is impossible to design 2-round $(3t, t)$ UVSS and 4-round $(2t + 1, t)$ UVSS protocol.

References

[1] Z. Beerliová-Trubíniová and M. Hirt. Efficient multi-party computation with dispute control. In *Proc. of TCC*, pages 305–328, 2006.

[2] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *STOC*, pages 1–10, 1988.

[3] D. Chaum, C. Crpeau, and I. Damgård. Multiparty unconditionally secure protocols. In *Proc. of FOCS 1988*, pages 11–19, 1988.

[4] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *Proc. of STOC 1985*, pages 383–395, 1985.

[5] R. Cramer, I. Damgård, S. Dziembowski, M. Hirt, and T. Rabin. Efficient multiparty computations secure against an adaptive adversary. In *Proc. of EUROCRYPT 1999*, pages 311–326.

[6] R. Cramer, I. Damgård, and S. Fehr. On the cost of reconstructing a secret, or vss with optimal reconstruction phase. In *Proc. of CRYPTO 2001*, pages 503–523, 2001.

[7] M. Fitzi, J. Garay, S. Gollakota, C. Pandu Rangan, and K. Srinathan. Round-optimal and efficient verifiable secret sharing. In *Proc. of TCC 2006*, pages 329–342.

[8] Rosario Gennaro, Yuval Ishai, Eyal Kushilevitz, and Tal Rabin. The round complexity of verifiable secret sharing and secure multicast. In *STOC*, pages 580–589, 2001.

[9] J. Katz, C. Koo, and R. Kumaresan. Improving the round complexity of vss in point-to-point networks. Cryptology ePrint Archive, Report 2007/358. To appear in Proc. of ICALP 2008.

[10] T. Rabin and M. Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority (extended abstract). In *STOC*, pages 73–85, 1989.

[11] A. Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.