Efficient Protocol for Generating IC Signature and its Application to Unconditional Verifiable Secret Sharing

Ashish Choudhary* **Arpita Patra** **Ashwinkumar B. V** **C. Pandu Rangan**

Department of Computer Science and Engineering
Indian Institute of Technology Madras
Chennai India 600036
Email:{ ashishc,arpita,ashwin }@cse.iitm.ernet.in, rangan@iitm.ernet.in

Abstract

Verifiable Secret Sharing (VSS) is a very important secure distributed computation task which allows a dealer to share a secret s from a finite field $\mathbb{F}$, among n players, in a way that would later allow a unique reconstruction of the secret. Essentially, VSS extends secret sharing to counter active Byzantine corrupted players, who try to fail VSS protocol by deviating from the protocol. Unconditional VSS (UVSS) attains the same goal of VSS, except with negligible error probability. An essential building-block for UVSS is unconditionally secure Information Checking (IC) protocol, which is used to generate unconditional secure IC signature and is of independent interest. IC signature attains the goal of cryptographic digital signatures, while providing unconditional (information theoretic with negligible error probability) security. We focus on the standard *secure channel model*, where all players have access to secure point-to-point channels and a common broadcast medium. In this model, it is known that UVSS protocol can be designed only with $n \geq 2t + 1$ players, where t denotes the fault tolerance threshold and bounds the total number of malicious (Byzantine) players having *unbounded computing power*. We propose a new IC protocol, which allows a dealer to sign on ℓ secrets simultaneously by communicating $O(\ell)$ field elements and broadcasting $O(n)$ field elements in three rounds, provided $\ell = \Omega(n)$. Hence our IC protocol is communication optimal and perhaps round optimal also. This can be compared with the previous IC protocols of [5] and [9], which takes four rounds of communication and signs on a *single secret*, with a communication overhead of $O(n)$ field elements and broadcasting $O(n)$ field elements. Using our new IC protocol, we then propose a four round UVSS protocol, where dealer can share $\Omega(n)$ secrets at once by communicating $O(n^3)$ field elements. This can be compared with UVSS protocol of [9] which shares a *single secret* by communicating $O(n^3)$ fields elements in four rounds. Thus the IC and UVSS protocol presented in this paper try to simultaneously improve both the communication and round complexity.

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

1 Introduction

In *secret sharing* [11], a dealer **D** wants to share a secret s from a finite field $\mathbb{F}$, among a set of n players, such that no set of t players will be able to reconstruct s from their shares, while any set of $t + 1$ or more players will be able to reconstruct s by combining 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**, who can be under the control of an active adversary, having *unbounded computing power*. In unconditional VSS (UVSS), each property of VSS holds, but with a negligible error probability [10]. UVSS is essentially a 2-phase protocol, consisting of a **Sharing Phase** and **Reconstruction Phase**. In the **Sharing Phase**, **D** distributes s among n players in a way that no t of them can infer any information on the secret. In the **Reconstruction Phase**, the players pool together their shares to reconstruct s. UVSS must satisfy the following with negligible error

[*] Work supported by Project no. CSE/O5-6/076/DITX/CPAN on Protocols for Secure Communication and Computation sponsored by Department of Information Technology, Govt of India.

probability: (a) if **D** is honest, then collusion of t actively corrupted players should not be able to prevent honest players from correctly reconstructing **D**'s secret, (b) a collusion of dishonest **D** and additional $(t-1)$ dishonest players cannot change the reconstructed secret, once it is decided in sharing phase. As in [7], we define the round complexity of UVSS protocol as the number of communication rounds in its sharing phase. Reconstruction can be done in a single round, wherein every player reveals its entire view generated during sharing phase. UVSS has wide application in secure multiparty computation (UMPC) [3, 5, 8, 1, 6]. UVSS has also many stand alone applications, like Byzantine agreement, generating global random coin, etc. A basic building block for designing UVSS protocols is *information checking* protocol (IC), a concept introduced in [10]. It can be informally described as an unconditionally secure signature scheme for authenticating data and is a sequence of three sub-protocols, namely **Distr, AuthVal** and **RevealVal** protocol (formal definition is given in Section 2). We define the round complexity of IC protocol as the number of communication rounds in **Distr** and **AuthVal** protocols.

Extant Literature: In *secure channel* model (point-to-point private channel and a common Broadcast channel), perfectly secure (zero error) VSS was studied in [2], where it is proved that perfect VSS is possible iff $n > 3t$. The exact round complexity of perfect VSS and tight trade-offs between the round complexity and fault tolerance threshold was established by Gennaro et. al. [7]. In VSS, it is possible to obtain better fault tolerance when negligible error probability of 2^{-k} is allowed, where k is the error/security parameter. In [10], it is shown that unconditionally secure VSS with negligible probability of error (UVSS) can be realized iff $n > 2t$. In [10] an IC and UVSS protocol is proposed. However, these protocols were quiet cumbersome. Later Cramer et. al. [5] proposed a more efficient UVSS protocol which requires 9 rounds, by using a new 4 round IC protocol. In [1], the authors have given a protocol, which does not completely satisfy the properties of IC protocol. Depending on the adversary behavior, the protocol may either terminate successfully, satisfying the properties of IC, or may terminate unsuccessfully, without satisfying the properties of IC, but detecting a pair of players where at least one of them is corrupted. In the same paper, the authors have designed another protocol, which does not completely satisfy the properties of UVSS. We do not compare our protocols with the protocols of [1]. Recently, the trade-off between the round complexity and fault tolerance threshold for UVSS is captured in [9]. The authors in [9] have termed UVSS as PVSS. Among several interesting results, the authors in [9] have given a four round UVSS protocol with $n = 2t+1$. The following table summarizes the communication complexity and round complexity of known UVSS and IC protocols with $n = 2t+1$. In the table, communication complexity denotes the number of bits communicated privately among the players.

UVSS				IC			
No. of Shared Secret(s)	# Rounds	Communication Complexity (bits)	Ref.	No. of Signed Secret(s)	Rounds	Communication Complexity (bits)	Ref.
1	9	$O((k+\log n)n^3)$	[5]	1	4	$O((k+\log n)n)$	[5]
1	4	$O((k+\log n)n^3)$	[9]	1	4	$O((k+\log n)n)$	[9]
$(n-t)$	4	$O((k+\log n)n^3)$	This paper	$(n-t)$	3	$O((k+\log n)n)$	This paper

Our Contribution: We propose a new IC protocol, which allows to simultaneously sign on $(n-t) = \Omega(n)$ secrets, by communicating $O((k+\log n)n)$ bits and broadcasting $O((k+\log n)n)$ bits in three rounds. Thus our IC protocol is communication optimal and perhaps round optimal also. This can be compared with the previous IC protocol of [5] and [9] with same communication overhead, which takes four rounds and allows to sign *only a single secret.* Using our new IC protocol, we then propose a four round UVSS protocol where **D** can share $(n-t) = \Omega(n)$ secrets at once, by communicating $O((k+\log n)n^3)$ bits and broadcasting $O((k+\log n)n^3)$ bits. This can be compared with four round UVSS protocol of [9] with the same communication complexity, which shares *only a single secret.* Thus our IC and UVSS protocol tries to simultaneously improve two important complexity measures (a) communication complexity and (b) round complexity.

2 Model and Definitions

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 $\mathbf{D}$ to be any one of the players from $\mathcal{P}$. Our protocols will *also* work for an external dealer where $\mathbf{D}$ is an entity outside the set $\mathcal{P}$. We assume the system to be synchronous. The adversary model is same as in [5]. The adversary, denoted as $\mathcal{A}_t$ has *unbounded computing power* and can actively control at most t of the n players (possibly including $\mathbf{D}$) during the protocol. Thus there exists at least $t+1$ honest players in the system. To actively corrupt a player means to take full control of the player; i.e. to make the player (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 collect all the messages addressed to the corrupted players and exploit this information to decide on what the corrupted players send during the same round. A player under the control of $\mathcal{A}_t$ will remain so throughout the protocol. The error probability in our protocols is expressed in terms of an error parameter k. The protocols operate in a finite field $\mathbb{F} = GF(q)$, where $q = \max(n, 2^k)$.

UVSS [5, 10]: A (n,t)-UVSS scheme for sharing secret $S = [S_1\ S_2\ \ldots\ S_\ell] \in \mathbb{F}^\ell$, with $\ell \geq 1$, consists of **Sharing Phase** and **Reconstruction Phase** and satisfies the following properties, with a negligible error probability (except SECRECY which is perfect) 2^{-k}, even in presence of $\mathcal{A}_t$:

1. TERMINATION: If $\mathbf{D}$ is honest then all honest players will complete sharing phase and if the honest players invoke reconstruction phase, then each honest player will complete it.

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

3. Once all currently uncorrupted players complete sharing phase, there exists a $S' \in \mathbb{F}^\ell$ which is *fixed*, such that the following holds:
   CORRECTNESS: If $\mathbf{D}$ is uncorrupted throughout sharing and reconstruction phase, then S' is the shared secret, i.e. $S' = S$ and each honest players will output S at the end of reconstruction phase.
   STRONG COMMITMENT: If $\mathbf{D}$ is corrupted then each honest player outputs S' upon completion of reconstruction phase.

As in [7], we define the round complexity of UVSS protocol as the number of communication rounds in sharing phase. Reconstruction phase can always be executed in a single round, where each player reveals its entire view generated during sharing phase. Using convention of [7], we assume that if $\mathbf{D}$ is discarded in sharing phase, then a pre-defined $S^* \in \mathbb{F}^\ell$ will be taken as $\mathbf{D}$'s secret.

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. The IC signatures can be used as semi "digital signatures". When a player $INT \in \mathcal{P}$ receives an IC signature from a dealer $\mathbf{D} \in \mathcal{P}$, then INT can later produce the signature and have the players in $\mathcal{P}$ verify that it is in fact 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 $S = [S_1\ S_2\ \ldots\ S_\ell] \in \mathbb{F}^\ell$, where $\ell \geq 1$ 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.

3. **RevealVal** $(\mathbf{D}, INT, \mathcal{P}, S)$ is carried out by INT and the receivers in $\mathcal{P}$, 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 **D**'s IC signature on S, obtained by INT. The IC signature must satisfy the following properties:

1. If **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 a S, which will be accepted in **RevealVal**, except with probability 2^{-k}.
3. If **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 **D** and INT are uncorrupted, then at the end of **AuthVal**, S is information theoretically secure from $\mathcal{A}_t$.

We define the round complexity of an IC scheme as the number of communication rounds in the **Distr** and **AuthVal** protocol. The **RevealVal** can be executed in a single round.

Remark 1 *As in [5], for the ease of exposition, in our protocols we adopt the convention that whenever a player P_i expects to receive some value from another player P_j in the next step and either no value or some syntactically incorrect value (such as higher degree polynomial, etc) arrives from P_j, then P_i replaces the received value by some fixed default syntactically correct value. Thus we do not treat separately the case when no value or syntactically incorrect value arrives.*

Remark 2 *Although, the definitions of IC and UVSS demand the error probability to be at most 2^{-k}, our protocols have an error probability of at most $2^{-k+O(\log n)}$. This does not violate the properties of UVSS and IC. By appropriately increasing the size of $\mathbb{F}$, we can achieve the error probability of at most 2^{-k} in our protocols, without any modification. In fact, the UVSS protocol of [5] has an error probability of at most $2^{-k+O(\log n)}$, but it can be made 2^{-k} by setting $|\mathbb{F}| = poly(n)2^k$, instead of $|\mathbb{F}| = max(n, 2^k)$.*

3 Efficient Information Checking Protocol with $n = 2t + 1$

We now present a three round IC protocol, called **IC**, which allows **D** to sign on $n-t = \Omega(n)$ secrets by communicating $O(n)$ field elements and broadcasting $O(n)$ field elements, where $n = 2t+1$. This is a significant improvement over the existing *four round* IC protocol of [5] and [9] (the IC protocol given in [9] is a slight modification of [5]), with same communication overhead and which allows **D** to sign on *only one* secret. Thus, our **IC** protocol takes one less round and allows **D** to sign on more secrets, without incurring additional communication overhead. Let $S = [S_1\ S_2\ \ldots\ S_{n-t}] \in \mathbb{F}^{n-t}$ denotes the $n - t$ secrets on which **D** wants to give his IC signature. The protocol **IC** is given in Table 1

Lemma 1 *If* **D** *and INT are honest, then* **D** *will not broadcast $F(x)$ during* **Round 3**. *Moreover, V^{Sh} and V^{Rec}_{FR} will match at atleast $t + 1$ locations.*

Proof: If INT is honest, then it will broadcast correct information during **Round 2**. So if **D** is also honest, then **D** will not broadcast $F(x)$ during **Round 3**. It is easy to see that in this case, V^{Sh} and V^{Rec}_{FR} will contain 1 at atleast $t + 1$ locations corresponding to honest receivers. □

Claim 1 *Let INT be honest and $P_i \in \mathcal{P}$ be an honest receiver. If* **D** *does not broadcast $F(x)$ during* **Round 3** *and if V^{sh} contains 0 at i^{th} position, then V^{Rec}_{FR} will also contain 0 at i^{th} position.*

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

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

Round 1: **D** selects a random $n-1$ degree polynomial $F(x)$ over $\mathbb{F}$, whose lower order $n-t$ coefficients are $S_1, S_2, \ldots, S_{n-t}$. In addition, **D** selects another random $n-1$ degree polynomial $R(x)$, over $\mathbb{F}$, which is independent of $F(x)$. **D** selects n distinct random elements $\alpha_1, \alpha_2, \ldots, \alpha_n$ from $\mathbb{F}$ such that each $\alpha_i \in \mathbb{F} - \{0, 1, \ldots, n-1\}$. **D** privately gives $F(x)$ and $R(x)$ to INT. To receiver $P_i \in \mathcal{P}$, **D** privately gives α_i, v_i and r_i, where $v_i = F(\alpha_i)$ and $r_i = R(\alpha_i)$. The polynomial $R(x)$ is called **authentication information**. The values $\alpha_i, v_i, r_i, 1 \leq i \leq n$ are called **verification information**.

AuthVal(D, $INT, \mathcal{P}, S$):

Round 2: Player INT chooses a random $d \in \mathbb{F} \setminus \{0\}$ and broadcasts $B(x) = dF(x) + R(x)$, along with d.

Round 3: **D** and the receivers in $\mathcal{P}$ parallely do the following:

- **D** checks the correctness of the information broadcasted by INT in **Round 2**. In addition, **D** also checks $dv_j + r_j \stackrel{?}{=} B(\alpha_j)$, for $1 \leq j \leq n$. If **D** finds any inconsistency, he broadcasts $F(x)$.
- 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 during **Round 3**, **D** has broadcasted $F(x)$, then accept the lower order $n-t$ coefficients of $F(x)$ as **D**'s secret and terminate the protocol, irrespective of the computation and communication done so far.
- If during **Round 3**, **D** has not broadcasted $F(x)$, then construct an n length bit vector, denoted by 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 **D** has not broadcasted $F(x)$ during **Round 3**, then we call the pair $(F(x), R(x))$ as **D**'s IC signature on the secret S (which are lower order $n-t$ coefficients of $F(x)$) given to INT.

RevealVal(D, $INT, \mathcal{P}, S$): INT broadcasts $F(x)$ and $R(x)$. Parallely, $P_i \in \mathcal{P}$ broadcasts α_i, v_i and r_i.

Local Computation (by each player):

1. For the polynomial $F(x)$ broadcasted by INT, construct an n length vector $V^{Rec}_{F(x)}$ where the j^{th} bit of $V^{Rec}_{F(x)}$ contains 1(0) if $F(\alpha_j) = v_j$ $(F(\alpha_j) \neq v_j)$. 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}.
2. If V^{Rec}_{FR} and V^{Sh} matches at atleast $t+1$ locations (irrespective of bit value at these locations), then accept the lower order $n-t$ coefficients of $F(x)$ as S. In this case, we say that **D**'s signature on S is correct. Else reject $F(x)$ broadcasted by INT and we say that INT has failed to produce **D**'s signature.

Table 1: A Three Round IC Protocol to Sign $n-t$ Secrets where $n = 2t+1$

Proof: An honest INT will correctly broadcast $B(x) = dF(x) + R(x)$ with respect to $F(x)$ and $R(x)$ which it has received from **D**. Also, since $P_i \in \mathcal{P}$ is an honest receiver, V^{sh} containing 0 at i^{th} position implies that $B(\alpha_i) \neq dv_i + r_i$. This further implies that **D** has distributed $F(x), R(x), \alpha_i, v_i$ and r_i in such a way that either $F(\alpha_i) \neq v_i$ or $R(\alpha_i) \neq r_i$ or both. So, during **RevealVal**, when INT broadcasts $F(x), R(x)$ and P_i broadcasts α_i, v_i, r_i, at least one of the vectors $V^{Rec}_{F(x)}, V^{Rec}_{R(x)}$ will contain 0 at i^{th} position. Hence V^{Rec}_{FR} will also contain 0 at i^{th} position. □

Claim 2 *Let INT be honest and $P_i \in \mathcal{P}$ be an honest receiver. If* **D** *does not broadcast $F(x)$ during* **Round 3** *and if V^{sh} contains 1 at i^{th} position, then $V^{Rec}_{FR} = V^{Rec}_{F(x)} \otimes V^{Rec}_{R(x)}$ will also contain 1 at i^{th} location, except with an error probability of at most 2^{-k}.*

Proof: An honest INT will correctly broadcast $B(x) = dF(x) + R(x)$ with respect to $F(x)$ and $R(x)$, received from **D**. Also since $P_i \in \mathcal{P}$ is an honest receiver, V^{sh} containing 1 at i^{th} position implies that $B(\alpha_i) = dv_i + r_i$. Now this equality holds in the following 2 cases:

1. **D** *has distributed $F(x), R(x)$ to INT and α_i, v_i, r_i to P_i in such a way that $F(\alpha_i) = v_i$ and $R(\alpha_i) = r_i$*: In this case, it is easy to see that during **RevealVal**, both $V^{Rec}_{F(x)}$ and $V^{Rec}_{R(x)}$ will contain 1 at i^{th} position. So $V^{Rec}_{FR} = V^{Rec}_{F(x)} \otimes V^{Rec}_{R(x)}$ will also contain 1 at i^{th} position.

2. **D** *has distributed $F(x), R(x)$ to INT and α_i, v_i, r_i to P_i in such a way that $F(\alpha_i) \neq v_i$ and $R(\alpha_i) \neq r_i$, but $B(\alpha_i) = dF(\alpha_i) + R(\alpha_i) = dv_i + r_i$*: We claim that this can happen for an unique $d \in \mathbb{F} - \{0\}$, which **D** (in this case, **D** is dishonest) must guess with probability $\frac{1}{|\mathbb{F}|-1} \approx 2^{-k}$ during **Round 1**. For otherwise, let there exist another $e(\neq d) \in \mathbb{F} - \{0\}$, such that $eF(\alpha_i) + R(\alpha_i) = ev_i + r_i$. This implies that $(d-e)F(\alpha_i) = (d-e)v_i$, which further implies that $F(\alpha_i) = v_i$, which is a contradiction. So in this case, if **D** can correctly guess the unique d (with probability at most 2^{-k}), then during **RevealVal**, both $V_{F(x)}^{Rec}$ and $V_{R(x)}^{Rec}$ and hence V_{FR}^{Rec} will contain 0 at i^{th} position. So V^{Sh} and V_{FR}^{Rec} will mismatch at i^{th} position. However, this can happen with probability at most 2^{-k}.

It is easy to see that above two are the only two possibilities, where $B(\alpha_i)$ will be equal to $dv_i + r_i$. If $F(\alpha_i) \neq v_i$ but $R(\alpha_i) = r_i$, then $B(\alpha_i) \neq dv_i + r_i$ because $d \neq 0$. Similarly, if $F(\alpha_i) = v_i$ but $R(\alpha_i) \neq r_i$, then $B(\alpha_i) \neq dv_i + r_i$). Hence the claim holds. □

Lemma 2 *If INT is honest and* **D** *has not broadcasted $F(x)$ during* **Round 3**, *then V^{Sh} and $V_{FR}^{Rec} = V_{F(x)}^{Rec} \otimes V_{R(x)}^{Rec}$ 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^{-k}.*

Proof: In the worst case, there exists $t+1$ honest receivers. Without loss of generality, let $P_1, P_2, \ldots, P_{t+1} \in \mathcal{P}$ be the $t+1$ honest receivers. From Claim 1, if V^{Sh} contains 0 at i^{th} position, for $i \in \{1, 2, \ldots, t+1\}$, then V_{FR}^{Rec} will also contain 0 at i^{th} position and hence they will match. Similarly, from Claim 2, if V^{Sh} contains 1 at i^{th} position, for $i \in \{1, 2, \ldots, t+1\}$, then except with probability at most 2^{-k}, V_{FR}^{Rec} will also contain 1 at i^{th} position. Thus, except with an error probability of at most 2^{-k}, V^{Sh} and V_{FR}^{Rec} will match at first $t+1$ positions. □

Lemma 3 *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 $2^{-k+O(\log n)}$.*

Proof: Without loss of generality, let $P_1, P_2, \ldots, P_{t+1} \in \mathcal{P}$ be the $t+1$ honest receivers and $P_{t+2}, P_{t+3}, \ldots, P_n$ be corrupted receivers. It is clear that if **D** is honest, then in order to forge **D**'s signature on $S' \neq S$, a corrupted INT should broadcast $F'(x) \neq F(x)$ during **RevealVal**, such that lower order $n-t$ coefficients of $F'(x) = S'$. Now to prevent **D** from broadcasting $F(x)$ in **Round 3**, INT must broadcast $B(x) = dF(x) + R(x)$ ($= dF'(x) + R'(x)$ possibly). Hence, V^{Sh} will contain 1 at the first $t+1$ locations corresponding to the $t+1$ honest receivers.

Now in **RevealVal**, INT broadcasts $F'(x) \neq F(x)$. In order that **D**'s signature on S' gets accepted, V_{FR}^{Rec} should match with V^{Sh} at atleast $t+1$ locations. Now V^{Sh} and V_{FR}^{Rec} can always match at the last t locations, corresponding to t corrupted receivers. Hence it is required that V^{Sh} and V_{FR}^{Rec} also matches at atleast one of the first $t+1$ locations. As mentioned above, V^{Sh} will contain 1 at first $t+1$ locations. So, in order that V^{Sh} and V_{FR}^{Rec} matches at atleast one of the first $t+1$ locations, V_{FR}^{Rec} should contain 1 at atleast one of the first $t+1$ locations. Let this be i^{th} location, where $i \in \{1, 2, \ldots, t+1\}$. This implies that INT should broadcast $F'(x) \neq F(x)$ during **GenRevealVal**, such that $F'(\alpha_i) = F'(\alpha_i)$. However, α_i is randomly selected from $\mathbb{F} - \{0, 1, \ldots, n\}$ and unknown to INT and $\mathcal{A}_t$. In addition, $\alpha_i \notin \{\alpha_{t+2}, \alpha_{t+3}, \ldots, \alpha_n\}$. Since, $F(x)$ and $F'(x)$ are both of degree $n-1$, they can have same value at atmost $n-1$ values of x. So the probability that INT could correctly guess α_k, such that $F'(x) \neq F(x)$, but $F(\alpha_i) = F'(\alpha_i)$ is at most $\frac{n-1}{|\mathbb{F}|-n-t} \approx 2^{-k}$. Since i can be any one of the first $t+1 = O(n)$ locations, the total error probability that a corrupted INT will be able to forge honest **D**'s signature on $S' \neq S$ is at most $O(n)2^{-k} = 2^{-k+O(\log n)}$ because $O(n) = 2^{O(\log n)}$. □

Lemma 4 *If* **D** *and INT are uncorrupted, then adversary $\mathcal{A}_t$ controlling t receivers in $\mathcal{P}$ does not get any information about the secret S before* **RevealVal**.

Proof: During **Round 1**, $\mathcal{A}_t$ will know t distinct point on the polynomials $F(x)$ and $R(x)$, implying information theoretic security for the lower order $n-t$ coefficients of both $F(x)$ and $R(x)$.

During **Round 2**, $\mathcal{A}_t$ will know d and the polynomial $dF(x) + R(x)$. Since both $F(x)$ and $R(x)$ are random and independent of each other, it still holds that lower order $n - t$ coefficients of $F(x)$ are information theoretically secure. Also, if **D** and INT are honest, then **D** will never broadcast $F(x)$ during **Round 3**. Hence the lemma is true. □

Theorem 1 *Protocol* **IC** *is a three round IC scheme and correctly generates IC signatures except with an error probability of at most* $2^{-k+O(\log n)}$.

PROOF: The proof follows from Lemma 1, 2, 3, 4 and the working of the protocol. □

Theorem 2 *Protocol* **IC** *communicates* $O(n(\log n + k))$ *bits and broadcasts* $O(n(\log n + k))$ *bits.*

PROOF: Follows from the fact that in **IC**, $O(n)$ field elements are communicated and $O(n)$ field elements are broadcasted and each field element is represented by $\log |\mathbb{F}| = \log n + k$ bits. □

4 Efficient Four Round UVSS with $n = 2t + 1$ Players

We first briefly outline the UVSS protocol proposed in [9] and [5].

UVSS Protocol of Cramer et. al: In [5], Cramer et.al have proposed a nine round UVSS protocol with $n = 2t + 1$. The protocol sequentially executes two set of IC protocols. The first set of IC protocols are executed by **D** to give its signatures on the shares to individual players. The second set of IC protocols are executed by individual players, where each individual player acts as a dealer. This is followed by other consistency checks, which take one additional round. Since the IC protocol proposed in [5] takes four rounds, the UVSS protocol given in [5] takes at most nine rounds. The UVSS protocol of [5] shares one secret from $\mathbb{F}$ by communicating $O((k+\log n)n^3)$ bits and broadcasting $O((k + \log n)n^3)$ bits, with an error probability of at most $2^{-k+O(\log n)}$.

UVSS Protocol of Arpita et. al: In [9], a four round UVSS protocol with $n = 2t+1$ is proposed. The protocol shares one secret from $\mathbb{F}$ by communicating $O((k + \log n)n^3)$ bits and broadcasting $O((k+\log n)n^3)$ bits, with error probability of at most 2^{-k}, where $|\mathbb{F}| = n^2(n-1)2^k$. The protocol makes use of a three round UWSS protocol proposed in the same paper, as a black-box, where UWSS is a weaker version of UVSS. The first set of IC protocols executed by **D** in the UVSS protocol of [5] are executed as such. However, the second set of IC protocols, executed by individual players in the UVSS protocol of [5] is replaced by a set of three round UWSS protocols, executed by individual players, acting as a dealer. Moreover, **D**'s actions are parallely executed and overlapped with the steps of the UWSS protocols, thus reducing the sharing phase to only four rounds.

We now present a four round UVSS protocol with $n = 2t + 1$ which shares $n - t = \Omega(n)$ secrets by communicating $O((k + \log n)n^3)$ bits and broadcasting $O((k + \log n)n^3)$ bits, with error probability of at most $2^{-k+O(\log n)}$. Thus, our protocol allows to share multiple secrets at the same time, incurring the same communication complexity as in the existing UVSS protocols of [9, 5]. The protocol makes use of our new three round **IC** protocol, which provides us with higher efficiency. We first explain the protocol to share only one secret value $s \in \mathbb{F}$. Later, we explain the modifications needed to share $\Omega(n)$ field elements with same communication overhead.

4.1 Four Round UVSS to Share Single Secret

The protocol is inspired by the principle used in the UVSS protocol of [5]. In our protocol, we use the following definition:

Definition 1 *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* $(e_0, e_1, \ldots, e_{n-1}) \in \mathbb{F}^n$ *is t-consistent if there exists a polynomial* $w(x)$ *of degree at most t such that* $w(i) = e_i$*, for* $0 \leq i \leq n - 1$.

The **Sharing Phase** of our four round UVSS protocol is presented in Table 2 and Table 3. In the protocol, **IC** protocols are used to sign on *only one* secret value, instead of $n-t$ secret values. Suppose P_i wants to give IC signature on a single secret value r to P_j, then P_i constructs an $n-t$ tuple S, where the first value is r and the remaining values are randomly selected from $\mathbb{F}$. Now P_i can initiate **IC** protocol to give signature on S.

Claim 3 *If $|\mathbf{D}^B| > t$, 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 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 4 *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 5 *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 $2^{-k+O(\log n)}$, $\mathbf{D}$ 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 3), corrupted P_i cannot forge honest $\mathbf{D}$'s signature on incorrect $f_i(j)$, except with an error probability of at most $2^{-k+O(\log n)}$. 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 $2^{-k+O(\log n)}$, $\mathbf{D}$ is corrupted. □

Claim 6 *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 $2^{-k+O(\log n)}$, $\mathbf{D}$ 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 3, this can happen with probability at most $2^{-k+O(\log n)}$. □

Claim 7 *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 $2^{-k+O(\log n)}$, $\mathbf{D}$ 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 $2^{-k+O(\log n)}$. □

Sharing Phase: Round I	
1. **D** chooses a random bivariate polynomial $F(x, y)$ of degree t in both variable, such that $F(0, 0) = s$. **D** computes $f_i(x) = F(x, i)$ and $g_i(y) = F(i, y)$ for $1 \leq i \leq n$. Considering player P_i as INT, **D** executes **Round 1** of $\mathbf{IC(D}, P_i, \mathcal{P}, f_i(j)\mathbf{)}$ and $\mathbf{IC(D}, P_i, \mathcal{P}, g_i(j)\mathbf{)}$, for $1 \leq j \leq n$, 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 $\mathbf{IC(}P_i, P_j, \mathcal{P}, r_{ij}\mathbf{)}$. The random value r_{ij} will be used by P_i and P_j to check the equality of $f_i(j)$ and $g_j(i)$, which will be common to both of them.
Sharing Phase: Round II	
1. In response to **Round 1** of $\mathbf{IC(D},P_i, \mathcal{P}, f_i(j))$ and $\mathbf{IC(D},P_i, \mathcal{P}, g_i(j))$, P_i acting as INT, executes **Round 2** of $\mathbf{IC(D}, P_i, \mathcal{P}, f_i(j)\mathbf{)}$ and $\mathbf{IC(D}, P_i, \mathcal{P}, g_i(j)\mathbf{)}$, for $1 \leq j \leq n$. Thus P_i tries to check the validity of **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 $\mathbf{IC(}P_i, P_j, \mathcal{P}, r_{ij}\mathbf{)}$, player P_j, acting as INT, executes **Round 2** of $\mathbf{IC(}P_i, P_j, \mathcal{P}, r_{ij}\mathbf{)}$ to check the validity of P_i's signature on r_{ij}, received during **Round I**. 3. Each player $P_i, 1 \leq i \leq n$ broadcasts: (a) $a_{ij} = f_i(j) + r_{ij}$, (b) $b_{ij} = g_i(j) + r_{ji}$.
Sharing Phase: Round III	
1. If **D** is not satisfied by the broadcast of P_i (as INT) in previous round (during the execution of **Round 2** of $\mathbf{IC(D}, P_i, \mathcal{P}, f_i(j)\mathbf{)}$ and $\mathbf{IC(D}, P_i, \mathcal{P}, g_i(j)\mathbf{)}$ for $1 \leq j \leq n$), then **D** broadcasts $f_i(x) = F(x, i)$ and $g_i(y) = F(i, y)$. This completes **D**'s actions of **Sharing Phase**. If **D** has not broadcasted $f_i(x)$ and $g_i(y)$, then P_i has obtained a valid IC signature of **D** on the n shares of $f_i(x)$ and $g_i(y)$.	2. If P_i is not satisfied by the broadcast of P_j (as INT) in previous round (during the execution of **Round 2** of $\mathbf{IC(}P_i, P_j, \mathcal{P}, r_{ij}\mathbf{)}$, then P_i broadcasts the signal $Unhappy_i^j$. We call the pair (P_i, P_j) as an **accusing pair**. 3. If P_i finds that the value $f_i(j)$'s or $g_i(j)$'s, $1 \leq j \leq n$, which are given to it by **D** during **Round I** of **IC** protocol are not t-consistent, then P_i broadcasts $Complaint_i^{\mathbf{D}}$ signal. In this case, 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 **D** has broadcasted $f_i(x)$ and $g_i(y)$ during **Round III**. If $\lvert\mathbf{D}^B\rvert > t$, then discard **D** and terminate **(see Claim 3)**. 2. For every $(P_i, P_j) \in \mathbf{D}^B$, check whether they are consistent (see Definition 1) with respect to $f(x)$ and $g(y)$ polynomials corresponding to them, which **D** has broadcasted during **Round III**. If not, then discard **D** and terminate **(see Claim 4)**. 3. If $P_i \in \mathbf{D}^B$, then the values $f_i(j)$'s and $g_i(j)$'s are known publicly. So terminate the execution of protocols $\mathbf{IC(D}, P_i, \mathcal{P}, f_i(j)\mathbf{)}$ and $\mathbf{IC(D}, P_i, \mathcal{P}, g_i(j)\mathbf{)}$, for $1 \leq j \leq n$. In addition, also terminate the execution of the protocols $\mathbf{IC(}P_i, P_j, \mathcal{P}, r_{ij}\mathbf{)}$, 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 defends himself by broadcasting $f_i(j)$ and $g_i(j)$, along with **D**'s signature on them. Similarly, P_j defends himself by broadcasting $f_j(i)$ and $g_j(i)$, along with **D**'s signature on them. Parallely, the receivers in $\mathcal{P}$ broadcasts the **verification information** corresponding to these signatures. 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 and the receivers in $\mathcal{P}$ 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 **D** during **Round I**, along with **D**'s signature on these values. Parallely, the receivers in $\mathcal{P}$ broadcasts the **verification information** corresponding to these signatures. 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 **D**'s signature on these values. Parallely, the receivers in $\mathcal{P}$ broadcasts the **verification information** corresponding to these signatures. 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.	

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

Local Computation by Each Player at the End of Round IV (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 **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 **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 **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 **D** and terminate (**see claim 5**)). 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 3)**.
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 **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 **D** and terminate (**see Claim 6**). 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 **D**'s valid signature on them. If not, then discard P_j from $\mathbf{D}^{NB}$.
   - (b) If **D**'s 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 **D**, corresponding to P_i during **Round III**). In case of any inconsistency, discard **D** and terminate (**see Claim 7**). 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 **D** and terminate the protocol (**see Claim 8**).

Table 3: Sharing Phase of four round UVSS with $n = 2t+1$ to share a secret value $s \in \mathbb{F}$ Contd ...

Claim 8 *At the end of* **Sharing Phase**, *if the size of final* $\mathbf{D}^{NB} < t+1$, *then* **D** *is corrupted.*

PROOF: The proof follows from the fact that if **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 **D** can be discarded and show that none can occur except with an error probability of at most $2^{-k+O(\log n)}$.

Lemma 5 *An honest* **D** *can be discarded during* **Sharing Phase** *only with an error probability of at most* $2^{-k+O(\log n)}$.

PROOF: It is easy to see that if **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 **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* **D***'s valid signature and either* $(f_i(j) \neq g_j(i))$ *or* $(f_j(i) \neq g_i(j))$: From Claim 5, this can happen for an honest **D** with an error probability of at most $2^{-k+O(\log n)}$. Since there can be $O(n^2)$ such pairs, the total error probability is $O(n^2)2^{-k+O(\log n)} = 2^{-k+O(\log n)}$.

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* **D** *on these values*: From Claim 6, this can happen for an honest **D** with an error probability of at most $2^{-k+O(\log n)}$. Since here can be $O(n)$ such corrupted players, the total error probability is $O(n)2^{-k+O(\log n)} = 2^{-k+O(\log n)}$.

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 7, this can happen for an honest $\mathbf{D}$ with an error probability of at most $2^{-k+O(\log n)}$. Since there can be $O(n^2)$ such pairs, the total error probability is $O(n^2)2^{-k+O(\log n)} = 2^{-k+O(\log n)}$. □

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^{-k}.

Lemma 6 *An honest player P_j can be discarded during* **Sharing Phase** *only with an error probability of at most 2^{-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 $g_j(i)$ (see Claim 1). 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 $1 - 2^{-k}$ (see Lemma 2). But, in the later case, P_j can be discarded, but this happens with an error probability of at most 2^{-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^{-k}, $\mathbf{D}$'s signature on these values are valid and will be accepted. However, with an error probability of at most 2^{-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^{-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 3 are true.

Theorem 3 *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 **D** *during* **Round III**.

Proof: The proof follows from the proof of the all the previous claims and working of the protocol. For space constraint, we give the proof in **APPENDIX**. □

If **D** is not discarded during **Sharing Phase**, the protocol proceeds to **Reconstruction Phase** as shown in Table 4. 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: 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 the random r_{ji}, which P_i has received from P_j during **Round I**, provided $P_j \in CORE$ and the share $g_i(j)$ is not known publicly. In addition, P_i also produces P_j's signature on r_{ji}. Parallely, the receivers in $\mathcal{P}$ broadcasts **verification information** corresponding to r_{ji}. Note that P_i does this for all such possible P_j's. 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:

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

   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 corresponding to the players in $CORE$ (who are not discarded from $CORE$), 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 (see Lemma 9).

Table 4: Reconstruction Phase of Four Round UVSS with $n = 2t + 1$.

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

Lemma 7 *The four 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 4). □

Lemma 8 *The UVSS protocol satisfies correctness property except with error probability of* $2^{-k+O(\log n)}$.

Proof: We have to only consider the case when **D** is honest. From Lemma 5, the probability that an honest **D** might get discarded during sharing phase is at most $2^{-k+O(\log n)}$. 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 $2^{-k+O(\log n)}$, P_i cannot forge an honest P_j's signature on incorrect r_{ji}. Moreover, from Property 7 of Theorem 3, P_i has committed $g_i(j)$ by agreeing with $f_j(i)$. Also, from Property 6 and Property 8 of Theorem 3, 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 $2^{-k+O(\log n)}$, it is consistent with $F(x, y)$ also. Hence the lemma holds. □

Lemma 9 *The four 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 6, 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 3, 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 3, 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 10 *The four round UVSS protocol communicates* $O(n^3(\log n + k))$ *bits and broadcasts* $O(n^3(\log n + k))$ *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 $3n^2$. As each execution of **IC** communicates $O(n(\log n + k))$ bits and broadcasts $O(n(\log n + k))$ bits (see Theorem 2), the UVSS protocol communicates $O(n^3(\log n + k))$ bits and broadcasts $O(n^3(\log n + k))$ bits. □

Theorem 4 *The four round UVSS protocol satisfies the properties of UVSS with an error probability of at most* $2^{-k+O(\log n)}$.

PROOF: The proof follows from Lemma 7, Lemma 8 and Lemma 9. □

4.2 Four Round UVSS to Share $(n - t)$ Secrets

Let $n = 2t + 1$. We now show how to adapt our four round UVSS protocol proposed in the previous section to share $n - t = t + 1$ secrets at the same time without incurring any extra communication overhead. Let $S = [s_1\ s_2\ \ldots\ s_{t+1}] \in \mathbb{F}^{t+1}$ denotes the secret that **D** wants to share. **D** generates $t + 1$ random and independent bivariate polynomials $F^k(x, y), 1 \leq k \leq t + 1$, 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 $g_i^k(y) = F^k(i, y)$, for $1 \leq k \leq t + 1$. **D** gives its IC signature on n shares of $f_i^k(x)$ and $g_i^k(y)$ to player P_i. More formally, **D** give its IC signature on the shares $f_i^k(j)$ and $g_i^k(j)$, for $1 \leq k \leq t + 1, 1 \leq j \leq n$ to player $P_i, 1 \leq i \leq n$. Recall that **IC** protocol can be used to generate IC signature of a player on $n - t = t + 1$ random secret values in a single execution. Hence, **D** can give its IC signature on the shares to P_i by parallely executing $2n$ instances of **IC** protocol (**D** has to give IC signature on $2n(t + 1)$ shares to P_i). Now each pair of distinct players (P_i, P_j) will have $2(t + 1)$ shares in common (in the previous protocol, they have 2 common shares). Player P_i, in order to check the consistency of common shares with P_j, will give

$t + 1$ random values to P_j, along with its IC signature on these values. Similarly P_j, in order to check the consistency of common shares with P_i, will give $t + 1$ random values to 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. The rest of the protocol now proceeds as in the UVSS protocol of previous section. If **D** is discarded during **Sharing Phase**, then a set of standard $t + 1$ values from $\mathbb{F}$ will be taken as **D**'s secrets. All the claims, lemmas and theorems of previous protocol will hold now also. The number of **IC** protocols executed by **D** is $2n^2$ ($2n$ executions per player). Similarly, the number of **IC** protocols executed by the players (as a dealer) is n^2. So the total number of **IC** protocols executed is still $3n^2$ and hence the communication complexity of the protocol is $O(n^3(\log n + k))$ bits. So we have the following theorem.

Theorem 5 *If $n = 2t+1$, then there exists a four round UVSS protocol that shares $n-t$ secrets and satisfies the properties of UVSS, except with an error probability of at most $2^{-k+O(\log n)}$. Moreover, the protocol communicates $O(n^3(\log n + k))$ bits and broadcasts $O(n^3(\log n + k))$ bits.*

Proof: Follows from the above discussion. □

5 Conclusion and Open Problems

All the existing *unconditional verifiable secret sharing* (UVSS) scheme shares only a *single secret* by communicating too many bits. In many practical distributed computing tasks there arises a need to share multiple secrets. In such a situation, the existing UVSS protocols will be very communication inefficient to use. In this paper, we have proposed a new four round UVSS protocol, which allows to share $n - t$ secrets simultaneously. Moreover, its communication complexity is same as the most efficient four round UVSS protocol of [9] which allows to share *only a single* secret. To design our UVSS protocol, we have also designed a new three round *information checking* (IC) protocol, which allows to simultaneously generate IC signature on multiple secrets. Moreover, it does so by incurring the same communication complexity as the existing IC protocols of [5, 9] which allows to generate IC signature on a single secret. We leave the issue of reducing the communication and round complexity of our protocols as open problem.

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. In *Proc. of 20th ACM STOC*, pages 1–10, 1988.

[3] D. Chaum, C. Crpeau, and I. Damgård. Multiparty unconditionally secure protocols (extended abstract). 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 (extended abstract). 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*, volume 1592 of *LNCS*, pages 311–326. Springer Verlag, 1999.

[6] I. Damgård and J. B. Nielsen. Scalable and unconditionally secure multiparty computation. In *Proc. of CRYPTO*, pages 572–590, 2007.

[7] 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.

[8] M. Hirt and U. M. Maurer. Robustness for free in unconditional multi-party computation. In *Proc. of CRYPTO 2001*, pages 101–118, 2001.

[9] Arpita Patra, Ashish Choudhary, AshwinKumar B.V, and C. Pandu Rangan. Probabilistic verifiable secret sharing tolerating adaptive adversary. Cryptology ePrint Archive, Report 2008/101, 2008. http://eprint.iacr.org/.

[10] Tal Rabin and Michael 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.

APPENDIX: Proof of Theorem 3

(1) Property 1 follows from Lemma 6. (2) If Property 2 is false, then from Claim 4, **D** would have been discarded, which is a contradiction. (3) From Claim 8, if $|\mathbf{D'}^{NB}| < t+1$, then **D** would have been discarded, which is a contradiction. So $|\mathbf{D'}^{NB}| \geq t+1$ and it contains at least one honest player and hence Property 3 is satisfied.

(4) If $P_i \in \mathbf{D'}^{NB}$ is an honest player and has received either t-inconsistent $f_i(j)$'s or $g_i(j)$'s during **Round I**, then from Claim 6, except with an error probability of at most $2^{-k+O(\log n)}$, **D** would have been discarded. Since **D** is not discarded, Property 4 holds.

(5) If there exists two honest players $P_i, P_j \in \mathbf{D'}^{NB}$, who are not consistent with each other, then (P_i, P_j) is a **conflicting pair**. So from Claim 5, **D** would have been discarded, which is a contradiction. So all honest players in $\mathbf{D'}^{NB}$ are consistent with each other. Similarly, if an honest player $P_j \in \mathbf{D'}^{NB}$ is inconsistent with some player $P_i \in \mathbf{D}^B$, then the inconsistency would have been revealed during $P_i - P_j - B - NB - Consistency - Checking - Broadcast$ and from Claim 7, **D** would have been discarded, which is a contradiction. Thus each honest player in $\mathbf{D'}^{NB}$ is consistent with all the players in $\mathbf{D}^B$. So, Property 5 is true.

(6) If $P_i, P_j \in \mathbf{D'}^{NB}$ and the pair (P_i, P_j) is either an **accusing** or **conflicting pair**, then during **Round IV**, they had broadcasted their common shares along with **D**'s signature on them. If the signatures would have been invalid, then at least one of P_i or P_j would have been discarded. On the other hand, if the signatures would have been valid, but the shares are unequal then **D** would have been discarded (see Claim 5). Since neither **D** is discarded, nor P_i, P_j are discarded, this implies that both P_i, P_j have produced the same common shares, along with valid **D**'s signature on them. Hence these shares are known publicly and Property 6 is true.

(7) If $P_i, P_j \in \mathbf{D'}^{NB}$, such that P_i is corrupted and P_j is honest and the blinded (XORed) common shares broadcasted by P_i, P_j during **Round II** contradict each other, then (P_i, P_j) would be a **conflicting pair**. Moreover, while resolving this confliction, either **D** or at least one of P_i, P_j would have been discarded. But neither **D** nor P_i, P_j is discarded. This implies that either the corrupted $P_i \in \mathbf{D'}^{NB}$ have broadcasted correct blinded common shares during **Round II**, that matches the corresponding blinded common share broadcasted by the honest $P_i \in \mathbf{D'}^{NB}$ (in which case (P_i, P_j) is not a **conflicting pair**) or the common shares produced by P_i, P_j to resolve the conflicting pair (P_i, P_j) are same (see Property 6). Thus, in any case, corrupted P_i has approved his commitment on $g_i(j)$ to everybody by agreeing with $f_j(i)$. So, Property 7 is true.

(8) If $P_k \in \mathbf{D}^B$, then it implies that D has broadcasted $f_k(x) = F(x,k)$ during **Round III**, thus publicly committing the shares $f_k(j), 1 \leq j \leq n$. Now if a corrupted $P_i \in \mathbf{D'}^{NB}$ does not agrees with this $f_k(x)$, then the values broadcasted by P_i during $P_k - P_i - B - NB - Consistency - Checking - Broadcast$ would have revealed the inconsistency between P_i and P_k. Moreover, either **D** or P_i would have been discarded while resolving the inconsistency (see Claim 7). Since neither has happened, it implies that the values broadcasted by P_i during $P_k - P_i - B - NB - Consistency - Checking - Broadcast$ are consistent with $f_k(x)$; i.e., $f_k(i) = g_i(k)$, which implies that P_i is approving his commitment on $g_i(k) = f_k(i) = F(i,k)$ publicly. Hence the last property holds.