The Round Complexity of Verifiable Secret Sharing: The Statistical Case

Ranjit Kumaresan[*]

Dept. of Computer Science
University of Maryland
ranjit@cs.umd.edu

Abstract. We consider the round complexity of a basic cryptographic task: *verifiable secret sharing* (VSS). This well-studied primitive provides a good "test case" for our understanding of round complexity in general; moreover, VSS is important in its own right as a central building block for, e.g., Byzantine agreement and secure multi-party computation.

The round complexity of *perfect* VSS was settled by Gennaro et al. (STOC 2001) and Fitzi et al. (TCC 2006). In a surprising result, Patra et al. (Crypto 2009) recently showed that if a negligible probability of error is allowed, the previous bounds no longer apply. We settle the key questions left open by their work, and in particular determine the *exact* round complexity of statistical VSS with optimal threshold. Let n denote the number of parties, at most t of whom are malicious. Their work showed that 2-round statistical VSS is impossible for $t \geq n/3$. We show that 3-round statistical VSS is possible iff $t < n/2$. We also give an *efficient* 4-round protocol for $t < n/2$.

1 Introduction

The round complexity of cryptographic protocols is a central measure of their efficiency, and has been the subject of intense study. In this work, we are interested in understanding the round complexity of *verifiable secret sharing* (VSS) [2]. Here, there is a *dealer* who shares a secret among a group of n parties, at most t of whom (possibly including the dealer) may be malicious. The requirements (roughly speaking) are that if the dealer is honest, then no information about the dealer's secret is revealed to the t malicious parties by the end of the *sharing phase*; nevertheless, by the end of the sharing phase even a dishonest dealer is irrevocably committed to *some* value that will be recovered by the honest parties in the *reconstruction phase*. Furthermore, if the dealer is honest then this committed value must be identical to the dealer's initial input.

We focus on *information-theoretic* VSS, where the security requirements are required to hold even when the malicious parties have unbounded computational power. Here, two different possibilities can be considered: either the security requirements hold *perfectly* (i.e., always), or the security requirements hold *statistically* but can possibly be violated with negligible probability. Assuming a

[*] Supported by the U.S. DoD/ARO MURI program and NSF award #0627306.

broadcast channel, perfect VSS is possible if and only if $t < n/3$ [1, 4], while statistical VSS is possible up to threshold $t < n/2$ [11].

The round complexity of perfect VSS has been extensively studied. For the case of optimal threshold (i.e., $t < n/3$), Gennaro et al. [6] showed that 3 rounds[1] are necessary and sufficient for perfect VSS, and gave an efficient 4-round protocol for the task. The 3-round VSS protocol by Gennaro et al. requires communication exponential in the number of players, but Fitzi et al. [5] later demonstrated that an *efficient* 3-round protocol is possible. Katz et al. [7] showed that perfect VSS could be achieved with optimal round complexity and, at the same time, optimal use of the broadcast channel.

The 3-round lower bound of Gennaro et al. was generally believed to apply also to the case of statistical VSS. It was therefore relatively surprising when Patra et al. [8] showed recently that *statistical* VSS could be realized in *two* rounds for $t < n/3$. The protocol of Patra et al. does not apply when $n/3 \leq t < n/2$, and finding a minimal-round protocol for the optimal security threshold was left open by their work. On the other hand, the work of Patra et al. proves that 2-round statistical VSS is impossible for $t \geq n/3$, which obviously applies to our setting as well.

Our results and organization of the paper. In this work we resolve the round complexity of statistical VSS with optimal threshold $t < n/2$. We show that 3-round statistical VSS is possible for any $t < n/2$. We also give an efficient 4-round protocol for $t < n/2$.

2 Model and Definitions

We consider the standard communication model where parties communicate in synchronous rounds using pairwise private and authenticated channels. We also assume a broadcast channel. (VSS is impossible for $t \geq n/3$ unless broadcast is assumed.) A broadcast channel allows any party to send the same message to all other parties — and all parties to be assured they have received identical messages — in a single round.

When we say a protocol tolerates t malicious parties, we always mean that it is secure against an adversary who may *adaptively* corrupt up to t parties during an execution of the protocol and coordinate the actions of these parties as they deviate from the protocol in an arbitrary manner. Parties not corrupted by the adversary are called *honest*. We always assume a *rushing* adversary; i.e., in any round the malicious parties receive the messages (including the broadcast messages) sent by the honest parties before deciding on their own messages.

In our protocol descriptions we assume without loss of generality that parties send properly formatted messages, as we may interpret an improper or missing message as some default message.

[1] Following the accepted convention, the round complexity of VSS refers to that of the sharing phase.

We let $\mathbb{F}$ denote a finite field and set $\kappa = \log|\mathbb{F}|$. We require the dealer's secret to lie in $\mathbb{F}$. In the case of statistical VSS, we allow *error* with probability at most $\varepsilon = 2^{-\Theta(\kappa)}$ and so κ can be treated as a security parameter. Note that the dealer's secret can be padded to lie in a larger field, if desired, to reduce the probability of error.

Definition 1. *A two-phase protocol for parties $\mathcal{P} = \{P_1, \ldots, P_n\}$, where a distinguished dealer $D \in \mathcal{P}$ holds initial input $s \in \mathbb{F}$, is a $(1-\varepsilon)$-*statistical VSS protocol tolerating t malicious parties *if the following conditions hold for any adversary controlling at most t parties:*

Privacy: *If the dealer is honest at the end of the first phase (the* sharing phase*), then at the end of this phase the joint view of the malicious parties is independent of the dealer's input s.*

Correctness/Commitment: *Each honest party P_i outputs a value s_i at the end of the second phase (the* reconstruction phase*). Except with probability at most ε, the following hold:*

1. *At the end of the sharing phase, the joint view of the honest parties defines a value s' such that $s_i = s'$ for every honest P_i.*
2. *If the dealer is honest throughout the execution, then $s' = s$.* ♢

Remark: Our definition of statistical VSS relaxes the *correctness/commitment* requirement, but not the *secrecy* requirement. This is the definition that has been considered previously in the literature, and is the definition that our protocols achieve.

3 A Multiple-Verifier Information Checking Protocol

Our protocols rely on what is known as an *information checking (sub)protocol* (ICP), a notion first introduced by Rabin and Ben-Or [11]. The traditional definition of an ICP [11, 3] involves the dealer D, an intermediary INT, and a verifier $\mathcal{V}$. In an initial phase, the dealer gives a secret value $s \in \mathbb{F}$ to INT and some verification information (that reveals nothing about s) to $\mathcal{V}$. Later, INT gives s to $\mathcal{V}$ along with a "proof" that s is indeed the value that INT received initially from D.

The basic definition of ICP involves only a *single* verifier; Patra et al. [10, 9], extend this definition to allow every party in the network to act as a verifier. Defining ICP in this way (i.e., enabling multiple verifiers) will be helpful when we use it as a black box in our VSS protocols. Formally, an *information checking protocol* (ICP) consists of three stages Distr, AuthVal, and RevealVal:

- Distr(D, INT, s) is initiated by D, using as input some value s. The algorithm generates some *authentication information* (which includes s itself) that is given to INT, as well as some *verification information* that is given to each of the verifiers.

- $\mathsf{AuthVal}(D, INT, s)$ is initiated by INT after receiving the authentication information from D. The information held by INT after this stage is called D's *IC-signature* and is denoted by $ICSIG(D, INT, s)$.
- $\mathsf{RevealVal}(D, INT, s)$ is a sub-protocol in which all messages are broadcast. Based on the broadcast messages, either $ICSIG(D, INT, s)$ is accepted or rejected by all honest verifiers (with high probability).

We require ICP to satisfy the following properties:

1. **Correctness1**: If D and INT are honest, then every honest verifier accepts $ICSIG(D, INT, s)$ during RevealVal.
2. **Correctness2**: If INT is honest then at the end of AuthVal, INT possesses an $ICSIG(D, INT, s)$, which will be accepted in RevealVal by each honest verifier, except with probability $2^{-\Omega(\kappa)}$.
3. **Correctness3**: If D is honest then during RevealVal, with probability at least $1 - 2^{-\Omega(\kappa)}$, $ICSIG(D, INT, s)$ revealed as some $s' \neq s$ by a corrupted INT will be rejected by *each* honest verifier.
4. **Secrecy**: If D and INT are honest, then till the end of AuthVal, the adversary has no information about s.

3.1 An ICP protocol

Here we reproduce a simplified version of the ICP protocol (from Patra et al., [10, 9]) tolerating $t < n/2$ malicious parties, such that Distr requires one round and AuthVal and RevealVal require two rounds each. We omit the proofs due to space limitations.

Distr(D, INT, s):
Round 1:

1. D sends the following to INT:
   (a) A random degree-t polynomial $F(x)$ over $\mathbb{F}$, with $F(0) = s$. Let INT receive $F'(x)$ as the polynomial with $F'(0) = s'$. [2]
   (b) A random degree-t polynomial $R(x)$ over $\mathbb{F}$. Let INT receive $R(x)$ as a t-degree polynomial $R'(x)$.
2. D privately sends the following to each verifier P_i:
   (a) (α_i, v_i, r_i), where $\alpha_i \in \mathbb{F}\backslash\{0\}$ is random (all α_i's are distinct), $v_i = F(\alpha_i)$ and $r_i = R(\alpha_i)$.

AuthVal(D, INT, s)**:**
Round 1: INT chooses a random $d \in \mathbb{F} \setminus \{0\}$ and broadcasts $(d, B(x))$ where $B(x) = dF'(x) + R'(x)$.

Round 2: D checks $dv_i + r_i \overset{?}{=} B(\alpha_i)$ for $i = 1, \ldots, n$. If D finds any inconsistency, he broadcasts $s^D = s$.

The polynomial $F'(x)$ (when D does not broadcast s^D in round 2 of AuthVal) or s^D (broadcast by D in round 2 of AuthVal) as held by INT is denoted by $ICSIG(D, INT, s)$.

[2] If INT is honest, then $F'(x) = F(x)$.

$\mathsf{RevealVal}(D, INT, s)$:
Round 1: INT broadcasts $ICSIG(D, INT, s)$ (i.e., he reveals D's secret contained in $ICSIG(D, INT, s)$ as $s' = s^D$ or as $s' = F'(0)$).
Round 2: Verifier P_i broadcasts Accept if one of the following conditions holds. (Otherwise, P_i broadcasts Reject.)

1. $ICSIG(D, INT, s) = s'$, and $s' = s^D$.
2. $ICSIG(D, INT, s) = F'(x)$, and one of the following holds.
   1. *C1*: $v_i = F'(\alpha_i)$; OR
   2. *C2*: $B(\alpha_i) \neq dv_i + r_i$ ($B(x)$ was broadcasted by INT during AuthVal).

Local Computation (By Every Verifier): If at least $t + 1$ verifiers have broadcasted Accept during round 2 of RevealVal then accept $ICSIG(D, INT, s)$ and output s' or $F'(0)$ (depending on whether $ICSIG(D, INT, s)$ is s' or $F'(x)$). Else reject $ICSIG(D, INT, s)$.

In our protocols, we use $\mathsf{AuthVal}^{(1)}$, $\mathsf{AuthVal}^{(2)}$ to denote the first round and second round of AuthVal respectively. Similarly $\mathsf{RevealVal}^{(1)}$, $\mathsf{RevealVal}^{(2)}$ are used for RevealVal. By $\mathsf{ICP_{sh}}(X, Y, s)$, we mean an execution $\mathsf{Distr}(X, Y, s)$ followed by $\mathsf{AuthVal}(X, Y, s)$. In order to make the presentation clearer, we sometimes use $\mathsf{ICP_{rec}}(X, Y, s)$ in place of $\mathsf{RevealVal}(X, Y, s)$. Also, in an execution $\mathsf{ICP_{sh}}(X, Y, s)$, we say that X *conflicts* with Y, if X had to broadcast correctional information in $\mathsf{AuthVal}^{(2)}(X, Y, s)$. Lastly we say that "$(F(x), d, B(x))$ is *consistent* with (α, v, r)" if at least one of the following holds:

1. $F(\alpha) = v$.
2. $B(\alpha) \neq dv + r$.

4 3-Round Statistical VSS with Optimal Resilience

In this section, we present a 3-round statistical VSS protocol with optimal resilience. Although the complexity of the protocol is exponential in terms of the number of parties, the protocol proves optimality of the lower bound from [8]. We also show an efficient 4-round statistical VSS protocol in Section 5.

In our 3-round VSS protocol, the dealer additively shares the secret s into $\binom{n-1}{t}$ shares. Loosely speaking, each of the $\binom{n-1}{t}$ shares correspond to a t-sized subset in $\mathcal{P} - \{D\}$. Then the dealer runs a "VSS-like" subprotocol to share s_m amongst the players in the t-sized subset $S_m \subseteq \mathcal{P} - \{D\}$. In the reconstruction phase, the shares corresponding to each subset are reconstructed first. These shares, in turn, are used to reconstruct the original secret s.

We begin by describing a subroutine that we call U-VSS.

4.1 *U*-VSS

The goal of the U-VSS sub-routine, is to achieve VSS-like functionality for a subset U (with $|U| = t$) of the player set $\mathcal{P}$. In particular, we want correctness and commitment property to hold as in the definition of VSS. However, the privacy requirement needs to met only when all players in $U \cup \{D\}$ are honest.

Informally, the 3 rounds of the U-VSS protocol can be described as follows:

- In Round 1, D sends the secret s to all players in U. Players in U exchange random pads with each other.
- In Round 2, each player in U authenticates his share (via $\mathsf{AuthVal}$). In addition he also broadcasts the secret masked with random pads received from other players in U. Players in U also authenticate random pads received from each other.
- In Round 3, D resolves conflicting broadcasts (if necessary, by broadcasting s to all players). Players finish authenticating their shares with D and their random pads with one another.

Unfortunately the U-VSS protocol described above does not guarantee commitment as such because players in U might (pretend to) have conflicts over random pads, thereby having an option to reveal different random pads in the reconstruction phase. To see this, consider the case when $n = 5$ and $t = 2$. Without loss of generality, let $U = \{P_2, P_3\}$. In round 3, P_2 might (or pretend to) be unhappy (i.e., the $\mathsf{AuthVal}^{(2)}$ check fails) with P_3's authentication of random pad r_{23} (sent by P_2 to P_3). This would result in P_2 broadcasting $F^{(2)}(x)$ and r_{23}. Similarly P_3 might (or pretend to) be unhappy with P_2 over r_{32}. Note that other players have no information about r_{23} and r_{32}. In this case, players in $\mathcal{P} - (U \cup \{D\})$ cannot distinguish (by the end of the sharing phase) between the following 3 cases:

1. (D and P_2 are dishonest.) P_2 broadcasted incorrect authentication information for r_{32} (thereby making P_3 unhappy over r_{32}) and pretends to be unhappy over P_3's broadcast related to r_{23}.
2. (D and P_3 are dishonest.) P_3 broadcasted incorrect authentication information for r_{23} (thereby making P_2 unhappy over r_{23}) and pretends to be unhappy over P_2's broadcast related to r_{32}.
3. (P_2 and P_3 are dishonest.) Both pretend to be unhappy over each other's broadcast related to random pads r_{23} and r_{32}.

Note that in Case (3), an honest D cannot detect any foul play by end of the 2^{nd} round. [3] If we are in Cases (1) or (2), then we have dishonest majority in $U \cup \{D\}$. Thus a dishonest D could take sides with either P_2's reveal or with P_3's reveal in the reconstruction phase. Depending on which player he supports, different secrets could be reconstructed. Note that the players in $\mathcal{P} - (U \cup \{D\})$ *may* not be able to tell whether P_2 or P_3 is honest and whose version of the secret they need to output.

However, in executions where there are no unresolved mutual conflicts, U-VSS does achieve the desired VSS properties. Looking back at the $n = 5, t = 2$ case, we motivate our definition of *mutual conflict* in the general case:

Definition 2. *A* mutual conflict *is said to exist in an execution of U-VSS if*

[3] If we allowed one more round, then Case (3) can be resolved in the following way. When any player broadcasts a "correction" value on a random pad, D will broadcast the secret s in the fourth round. With this modification, commitment can be achieved easily.

1. *Some P_i broadcasted $r_{ij}, F^{(i)}(x)$ for some P_j;* and
2. *P_j also broadcasted $r_{ji}, F^{(j)}(x)$;* and
3. *D did not broadcast s in round 3 of the sharing phase.* $\diamondsuit$

To begin with, we want our U-VSS protocol to satisfy the following weak property: If there is no *mutual conflict* in an execution of U-VSS, then:

- If all players in $U \cup \{D\}$ are honest, then no information about s is revealed to players in $\mathcal{P} - (U \cup \{D\})$ at the end of the sharing phase.
- If D is honest, then D is not discarded in the sharing phase. Also, if there is no mutual conflict then the value shared by D is reconstructed with high probability.
- There exists a value s', such that D is committed to s' at the end of the sharing phase. This s' is reconstructed in the reconstruction phase.

4.2 A Protocol for U-VSS

We present a protocol for U-VSS protocol which satisfies the above requirements.
Inputs: Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ denote the set of players and let $D = P_1$ be the dealer with input s. Let $U \subset \mathcal{P}$ be the target subset with $|U| = t$.
Sharing Phase:
Round 1:

1. Execute $\mathsf{ICP_{sh}}(D, P_i, s)$. for every party P_i in the subset U. Let P_i receive s from D as $s^{(i)}$. Denote the polynomials used in $\mathsf{Distr}(D, P_i, s)$ by $F^{(i)}(x), R^{(i)}(x)$ (both are random t-degree polynomials with $F^{(i)}(0) = s^{(i)}$).
2. For each pair (P_i, P_j) from subset U, party P_i picks a random value r_{ij} and executes $\mathsf{ICP_{sh}}(P_i, P_j, r_{ij})$ for every $P_j \in U \cup \{D\}$. Let P_j receive r_{ij} from P_i as r'_{ij}.

Round 2: Each $P_i \in U \cup \{D\}$ broadcasts $a_{ij} := s^{(i)} + r_{ij}$ and $b_{ij} := s^{(i)} + r'_{ji}$ for every $P_j \in U \cup \{D\}$.
Round 3:

1. If for some $P_i, P_j \in U \cup \{D\}$, $a_{ij} \neq b_{ji}$ or $a_{ji} \neq b_{ij}$, then D broadcasts s.
2. If P_i conflicts with P_j, then he broadcasts $r_{ij}, F^{(i)}(x)$.

Local Computation: D is **discarded** if for some $P_i, P_j \in U \cup \{D\}$, $a_{ij} \neq b_{ji}$ or $a_{ji} \neq b_{ij}$, and D did not broadcast s.
Reconstruction Phase: If D broadcasted s in round 3 of the sharing phase, then each player P_i sets $s^{(i)} := s$ and outputs s and terminates.

If there is a *mutual conflict* then each player (in $\mathcal{P}$) outputs $\perp$ and the reconstruction phase terminates. Else,

1. Each $P_i \in U$ executes $\mathsf{ICP_{rec}}(D, P_i, s)$ and each $P_j \in U \cup \{D\}$ executes $\mathsf{ICP_{rec}}(P_i, P_j, r_{ij})$.
2. D broadcasts the secret s.

Local Computation Construct GOOD in the following way: For $P_i \in U$, include P_i in GOOD if

1. P_i is successful in revealing $s^{(i)}$.
2. For each P_j that did not conflict with P_i, P_i is successful in revealing r'_{ji}.
3. For every r'_{ji} revealed by P_i in the previous step, $a_{ji} = s^{(i)} + r'_{ji}$ holds.
4. If r'_{ij} was successfully revealed by any P_j, $a_{ij} = s^{(i)} + r'_{ij}$ holds.

Compute s' as follows:

1. If GOOD is empty, then $s' := s$, where s is D's broadcast in Step 2.
2. Else pick any $P_i \in \mathsf{GOOD}$ and assign $s' := s^{(i)}$.

Output s'.

4.3 Proofs

We show that the U-VSS protocol presented above satisfies the necessary requirements through a series of claims.

The following claim is proved by means of a standard argument. We omit the proof due to space limitations.

Claim 1. If all players in $U \cup \{D\}$ are honest, then no information about s is revealed to players in $\mathcal{P} - (U \cup \{D\})$ at the end of the sharing phase.

It is easy to see that an honest D is never discarded in the sharing phase.

Claim 2. If there is no mutual conflict then the value shared by honest D, say s, is reconstructed with high probability.

Proof. Since only the values held by $P_i \in \mathsf{GOOD}$ are reconstructed, we need to argue that a dishonest P_i is contained in GOOD only if he reveals $s^{(i)} = s$. This is easily shown since when D is honest, by **Correctness3**, every successful reveal is equal to s.

Claim 3. If D is not discarded, then for all honest P_i, $s^{(i)} = s'$ for some s'.

Proof. Assume that honest players $P_i, P_j \in U$ received shares $s^{(i)}, s^{(j)}$, with $s^{(i)} \neq s^{(j)}$. Then in round 2, a_{ij} is not equal to b_{ji}. Therefore, D has to broadcast s, otherwise he is discarded. Consequently every P_i sets $s^{(i)} := s'$ (see Local Computation).

The following claim can be easily verified.

Claim 4. If D is not discarded, and does not broadcast s in the sharing phase, then with high probability, all honest players in U are contained in GOOD.

Claim 5. If there is no mutual conflict, then there exists a value s' such that D is committed to s' at the end of the sharing phase. This s' is reconstructed in the reconstruction phase.

Proof. When D is honest, the claim follows from Claim 2. Assume D is dishonest. If D is discarded in the sharing phase, then the claim trivially holds. In the following, we assume that D is not discarded. Since D is dishonest and $U \cup \{D\}$ contains $(t+1)$ players, there exists an honest $P_j \in U$. From Claim 3, we have that all honest players received the same share $s' = s^{(j)}$ (P_j's share) from D. We now show that if there is no *mutual conflict*, then s' is reconstructed.

The idea is to show that any $P_i \in U$ is contained in GOOD only if he reveals $s^{(i)}$ as s'. This would prove the claim, since all honest players are already contained in GOOD (follows from Claim 4).

For the sake of reaching a contradiction, assume that $P_i \in U$ successfully reveals $s^{(i)} \neq s'$. We consider two cases:
Case 1: P_j did not conflict with P_i.
By **Correctness3**, with high probability, P_i can successfully reveal r'_{ji} only as r_{ji}. Since P_j used r_{ji} to compute a_{ji}, it holds that $a_{ji} \neq s^{(i)} + r'_{ji}$ for $s^{(i)} \neq s'$. Hence in this case, P_i will not be included in GOOD.
Case 2: P_i did not conflict with P_i.
By **Correctness2**, with very high probability, it holds that P_j successfully revealed r'_{ij} that he received. Since D is not discarded, $a_{ij} = b_{ji} = s' + r'_{ij}$. Observe that the condition "$a_{ij} = s^{(i)} + r'_{ij}$" will not be satisfied for $s^{(i)} \neq s'$. Hence in this case, P_i will not be included in GOOD.

The cases discussed above are sufficient since there are no *mutually conflicting* parties in U, i.e., we do not have to consider the case when both P_i and P_j broadcast the random pads which they had used.

4.4 Building Statistical VSS for $t < n/2$ from U-VSS

In the previous section we saw how U-VSS gives us the desired VSS properties when there is no *mutual conflict.* In this section, we'll develop techniques to cope up with executions in which there is *mutual conflict.* Let's first look at the $n = 5, t = 2$ case. There's a small trick that we can use to achieve commitment: First observe that a *mutual conflict* arises when at least 2 parties in $U \cup \{D\}$ are corrupted. Since $U = \{P_2, P_3\}$ and $t = 2$, all players in $\mathcal{P} - (U \cup \{D\})$ are honest. (For higher n, this is not the case, and hence the difficulty is amplified.) Since conflicting P_2, P_3 would have revealed their polynomials $F^{(2)}(x), F^{(3)}(x)$ (with $F^{(2)}(0) \neq F^{(3)}(0)$) respectively, the reveals for the set U is fixed. Since P_4 and P_5 are honest, the "check points" are also fixed! The key observation is that for an honest D (Case (3)), dishonest P_2, P_3 will not be able to guess the honest "check points" correctly. If D is honest then at least one of the revealed polynomials is not *consistent* with any of the honest "check points" except with negligible probability. So one of P_2, P_3's reveal will not be $\mathsf{Accepted}$.

For general t, n, when we encounter a *mutual conflict* in an U-VSS execution, all players in $\mathcal{P} - (U \cup \{D\})$ are not necessarily honest. So instead of assigning a "check point" to each player, we assign a "check point" to each t-sized subset via an U-VSS protocol. In addition, to avoid the problems caused by *mutual conflicts*, only those U-VSS executions in which is no *mutual conflict* are used to generate the verification points in the reconstruction phase. The reason behind using U-VSS to share the "check points" is that now checking for Consistency is made public (i.e., dishonest players can no longer arbitrarily broadcast Accept or Reject to force a favorable outcome). U-VSS executions with no *mutual conflict*, guarantee agreement over the revealed check points. This results in an agreement over which of the revealed polynomials are actually *consistent*. There might be two conflicting polynomials both of which satisfy all the check points. However at the end of the sharing phase, the outcome of the check for Consistency is fixed! If two conflicting polynomials do pass the Consistency test, then $\perp$ is reconstructed. Note that this does not violate the commitment property of VSS since whether $\perp$ is reconstructed is fixed at the end of the sharing phase. (We assume that $\perp$ represents a default element in $\mathbb{F}$). Also, dishonest players could possibly reveal incorrect polynomials in the reconstruction phase. We prove that our statistical VSS protocol is robust against such adversarial behavior.

4.5 A 3-round protocol for VSS

Inputs: Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ denote the set of players and let $D = P_1$ be the dealer with input s. Let $T \overset{\text{def}}{=} 2^t - 1$.
Sharing Phase: D additively shares s into $s_1, \ldots, s_K$ where $s_1, \ldots, s_K$ are random subject to $s = s_1 + s_2 + \ldots + s_K$. The following U-VSS executions are run in parallel.

1. Iterate over all t-sized subsets S_m: Execute $U\text{-VSS}(D, S_m, s_m)$.
2. For each player subset S_k of size t, D picks "check points" $(\alpha_k^{(m,i)}$, $v_k^{(m,i)} = F_m(\alpha_k^{(m,i)})$, $r_k^{(m,i)} = R_{m,i}(\alpha_k^{(m,i)}))$ and sends it to S_k (to check for the polynomials revealed by each $P_i \in S_m$). Execute $U\text{-VSS}(D, S_k, (\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)}))$ for all $P_i \in S_m$, and for every t-sized subset S_m.

Local Computation: D is **discarded** if at least one of the following hold:

1. D is discarded in some execution of $U\text{-VSS}(D, S_k, (\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)}))$.
2. D is discarded in some execution of $U\text{-VSS}(D, S_m, s_m)$.

Reconstruction Phase: Let $\mathcal{B} \overset{\text{def}}{=} \{S_m \mid D \text{ broadcasted } s_m\}$. Let

$$\mathcal{A}_{m,i} \overset{\text{def}}{=} \{S_k \mid \text{There are no } \textit{mutual conflicts} \text{ in an execution of } U\text{-VSS}(D, S_k, (\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)}))\}$$

Reconstruction phase consists of the following 2 rounds:
Round 1: Iterate over all S_m, and every $P_i \in S_m$: Execute reconstruction phase of $U\text{-VSS}(D, S_m, s_m)$, and $U\text{-VSS}(D, S_k, (\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)}))$ (for each $S_k \in \mathcal{A}_{m,i}$).
Round 2: Reveals started in round 1 are completed in this round. Also D broadcasts s_m for each S_m.

Local Computation: Let

$$\mathcal{C}_m \overset{\text{def}}{=} \{F_m^{(i)}(x) \mid P_i \in S_m \text{ broadcasted } F_m^{(i)}(x) \text{ and } \textit{mutually conflicted} \\ \text{with some } P_j \in S_m \text{ in the sharing phase}\}$$

All players reconstruct $\perp$ if for any S_m:

1. There is a player $P_i \in S_m$ with $F_m^{(i)}(x) \in \mathcal{C}_m$ and $(F_m^{(i)}(x), d_m^{(i)}, B_m^{(i)}(x))$ *consistent* with $(\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)})$, for all $S_k \in \mathcal{A}_{m,i}$; AND
2. There is a player P_j $(\neq P_i) \in S_m$ with $F_m^{(j)}(x) \in \mathcal{C}_m$, $F_m^{(i)}(0) \neq F_m^{(j)}(0)$ and $(F_m^{(j)}(x), d_m^{(j)}, B_m^{(j)}(x))$ *consistent* with $(\alpha_k^{(m,j)}, v_k^{(m,j)}, r_k^{(m,j)})$ for all $S_k \in \mathcal{A}_{m,j}$.

If $\perp$ is not reconstructed, then for each $S_m \notin \mathcal{B}$ construct $\mathsf{GOOD_m}$ in the following way: Include $P_i \in S_m$ in $\mathsf{GOOD_m}$ if

1. P_i is contained in GOOD corresponding to the execution U-VSS(D, S_m, s_m).
2. $(F_m^{(i)}(x), d_m^{(i)}, B_m^{(i)}(x))$ is *consistent* with $(\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)})$ for all $S_k \in \mathcal{A}_{m,i}$ (where $F_m^{(i)}(x), d_m^{(i)}, B_m^{(i)}(x)$ are internal variables in $\mathsf{ICP_{sh}}(D, P_i, s_m)$ corresponding to U-VSS(D, S_m, s_m) with $P_i \in S_m$). Let $s_m^{(i)} = F_m^{(i)}(0)$.

Compute s'_m (which is D's commitment to S_m) as follows:

1. For $S_m \in \mathcal{B}$, set s'_m to be the one broadcasted by D during round 3 of the sharing phase.
2. For $S_m \notin \mathcal{B}$, pick any $P_i \in \mathsf{GOOD_m}$ and set $s'_m = s_m^{(i)}$. If $\mathsf{GOOD_m}$ is empty, then $s'_m = s_m$, where s_m is D's broadcast in round 2 of reconstruction phase.

Reconstruct D's secret as $s' = \sum_{m=1}^{K} s'_m$.

4.6 Proof of Correctness for 3-Round-VSS

We now prove that 3-Round-VSS satisfies all the properties required of a statistical VSS protocol. Let $T \overset{\text{def}}{=} 2^t - 1$.

The following lemma is proved by means of a standard argument. We omit the proof due to space limitations.

Lemma 1. *(**Secrecy**) Protocol 3-round-VSS satisfies perfect secrecy.*

Lemma 2. *(**Correctness**) Protocol 3-Round-VSS satisfies $(1-\varepsilon)$-correctness property.*

Proof. It is easy to see that an honest D is never discarded in the sharing phase. We now show that with high probability, $\perp$ is not reconstructed whenever D is honest.

The only possibility of $\perp$ getting reconstructed is when there exist two *mutually conflicting* players $P_i, P_j \in S_m$ (with $S_m \notin \mathcal{B}$) such that $(F_m^{(i)}(x), d_m^{(i)}, B_m^{(i)}(x))$, $(F_m^{(j)}(x), d_m^{(j)}, B_m^{(j)}(x))$ are *consistent* with $(\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)})$, $(\alpha_l^{(m,j)}, v_l^{(m,j)}, r_l^{(m,j)})$ (respectively) for all $S_k \in \mathcal{A}_{m,i}$ and $S_l \in \mathcal{A}_{m,j}$. Since D is honest, at least one of P_i, P_j has to be dishonest (otherwise they wouldn't conflict on random pads and broadcast their polynomials).

The key point is that there is at least one set, say $S_l (\neq S_m)$ which contains only honest players. Since all the players are honest, there is no *mutually conflicting* pair in S_l. As a result, $S_l \in \mathcal{A}_{m,i} \cap \mathcal{A}_{m,j}$. By Claim 1, no information is revealed about $(\alpha_l^{(m,i)}, v_l^{(m,i)}, r_l^{(m,i)}), (\alpha_l^{(m,j)}, v_l^{(m,j)}, r_l^{(m,j)})$. Also the correct values $(\alpha_l^{(m,i)}, v_l^{(m,i)}, r_l^{(m,i)}), (\alpha_l^{(m,j)}, v_l^{(m,j)}, r_l^{(m,j)})$, as shared by D, are revealed in the reconstruction phase of the corresponding U-VSS protocols (follows from Claim 2). So if a dishonest player, say P_i is able to discard an honest D by revealing $F_m^{(i)}(x) \neq F_m(x)$, then he must have guessed $\alpha_l^{(m,i)}$ (follows from the proof of **Correctness3**). This happens with negligible probability.

Given that $\perp$ is not reconstructed, a dishonest P_i revealing $F_m^{(i)}(x) \neq F_m(x)$ can be in $\mathsf{GOOD_m}$ only if he guessed $\alpha_l^{(m,i)}$ where S_l is the set of honest players (as described above). Again, this happens with negligible probability. Correctness follows immediately.

Claim 6. If a corrupted D is not discarded, then for every S_m, at least one honest player is contained in $\mathsf{GOOD_m}$ with very high probability.

Proof. First, let us fix an S_m. By Claim 5 (commitment property for U-VSS), we have that for every tuple $(\alpha_k^{(m,i)}, v_k^{(m,i)}, r_k^{(m,i)}) \in \mathcal{C}_m$, the exact tuple was held by (all) the honest player(s) in S_k. This essentially makes every verification "check point" behave as if it were possessed by an honest player. Now, from the proof of **Correctness2** for *ICP* [4] , each honest player in S_m is *consistent* with "check points" in $\mathcal{C}_m$ with high probability $(1 - \frac{1}{|\mathbb{F}|-1})$.

Suppose there are k honest players in S_m. By the above argument, the claim can fail for a given S_m, only if it fails for each honest player in S_m. This happens with probability at most $\frac{1}{(|\mathbb{F}|-1)^k}$ [5]. Since there are $\binom{t+1}{k}\binom{t-1}{t-k}$ such S_m, the probability that the claim fails for any one such S_m is bounded by $\frac{t^{2k}}{|\mathbb{F}|^k}$. Summing over all k, we see that D can cause the claim to fail for any one S_m with probability at most $\frac{2t^2}{|\mathbb{F}|} = 2^{-\Theta(\kappa)}$. Hence the claim holds.

Lemma 3. *(**Commitment**) Protocol 3-Round-VSS satisfies* $(1-\varepsilon)$*-commitment property.*

Proof. For an honest D, the lemma follows from Lemma 2. In the following, we assume that D is dishonest. First we show that whether or not, $\perp$ is reconstructed, is fixed at the end of the sharing phase. Note that the polynomials in

[4] The proof is identical since in both cases we are dealing with a dishonest D and an honest intermediary. In both cases, the dealer wasn't unhappy with $\mathsf{AuthVal}^{(1)}(D, P_i, s)$, where s is the dealer's secret.

[5] We have used the fact that a corrupt D's ability to cause failure for a particular honest player is *independent* of his ability to cause failure for a different honest player. This is true because D can cause failure for an honest $P_i \in S_m$, only by guessing $d_m^{(i)}$ (follows from the proof of **Correctness2**). A different honest player $P_j \in S_m$, chooses $d_m^{(j)}$ independent of $d_m^{(i)}$. Hence our argument is justified.

$\mathcal{C}_m$ are taken from the sharing phase. Also, the "check points" for these polynomials are fixed at the end of the sharing phase (by the commitment property of U-VSS proved in Claim 5). Therefore, the decision of whether $\perp$ is reconstructed, is fixed at the end of the sharing phase. Since $\perp\in \mathbb{F}$ (by our assumption), we achieve commitment even when $\perp$ is reconstructed.

We prove commitment in the case when $\perp$ is not reconstructed. By Claim 6, we now need to prove that for each $S_m \notin \mathcal{B}$, the share held by honest player(s), say $s'_m = s_m^{(j)}$ for some honest P_j, will be reconstructed with high probability (Recall that, by Claim 3, all honest players in $U = S_m(\notin \mathcal{B})$ have the same share).

Let us assume (for the sake of reaching a contradiction) that some dishonest $P_i \in S_m$ successfully reveals some $s_m^{(i)} \neq s_m^{(j)}$. Let $a_{ij}^m, b_{ij}^m, r_{ij}^m$ be the internal variables used in U-VSS(D, S_m, s_m) with $P_i, P_j \in S_m$. We consider two cases:
Case 1: P_j did not broadcast r_{ji}^m in round 3 of the sharing phase.
By **Correctness3**, with very high probability, P_i can successfully reveal $r_{ji}^{m\prime}$ only as r_{ji}^m. Since P_j computed $a_{ji}^m := s_m^{(j)} + r_{ji}^m$, it holds (with high probability) that $a_{ji}^m \neq s_m^{(i)} + r_{ij}^{m\prime}$ for $s_m^{(i)} \neq s_m^{(j)}$. Hence in this case, P_i will not be included in $\mathsf{GOOD_m}$.
Case 2: P_i did not broadcast r_{ij}^m in round 3 of the sharing phase.
By **Correctness2**, with very high probability, it holds that P_j revealed $r_{ij}^{m\prime}$ as the random pad that he used in computing $b_{ji}^m := s_m^{(j)} + r_{ij}^{m\prime}$. Since $S_m \notin \mathcal{B}$, and since D is not discarded, we have $a_{ij}^m = b_{ji}^m$. Therefore, the condition "$a_{ij}^m = s_m^{(i)} + r_{ij}^{m\prime}$" will not be satisfied for any $s_m^{(i)} \neq s_m^{(j)}$. Hence in this case, P_i will not be included in $\mathsf{GOOD_m}$.

We do not have to consider the case when both P_i, P_j broadcasted the random pads which they had used (in round 3). This is because if some P_i revealed $F_m^{(i)}(x)$ (with $F_m^{(i)}(0) \neq s'_m$) *consistent* with the all the revealed "check points", then $\perp$ will be reconstructed. Hence an honest P_j's share (i.e., $s_m^{(j)} = s'_m$) is reconstructed always. Given this, commitment follows immediately.

The theorem follows from Lemmas 1, 2 and 3.

Theorem 1. *There exists a 3-round statistical VSS protocol tolerating $t < n/2$ malicious parties.*

5 Efficient 4-Round Statistical VSS with Optimal Resilience

We now design a 4-round sharing, 2-round reconstruction $(2t + 1, t)$ statistical VSS with polynomial communication complexity. In the protocol, D selects a random symmetric bivariate polynomial $F(x, y)$ such that $F(0, 0) = s$ and sends $f_i(x)$ to P_i. At the end of the sharing phase, if D is not discarded then every honest P_i holds a degree t polynomial $f_i(x)$ such that for every pair of honest parties (P_i, P_j), $f_i(j) = f_j(i)$. This implies that if D is not discarded, then the

$f_i(x)$ polynomials of the honest parties define a symmetric bivariate polynomial $F^H(x, y)$. Moreover in the protocol, it is ensured by using the properties of $ICSig$ that no corrupted P_i will be able to disclose $f'_i(x) \neq f_i(x)$ in the reconstruction phase. Hence irrespective of whether D is honest or corrupted, reconstruction of $s = F^H(0, 0)$ is enforced. To achieve all the properties of VSS, D gives $ICSig$ to individual parties, and concurrently every individual party gives $ICSig$ to every other party. The protocol is somewhat inspired by the VSS protocol of [3]. As the ICP proposed in [3] takes one round in Distr, 3 rounds in $\mathsf{AuthVal}$ and 2 rounds in $\mathsf{RevealVal}$, the VSS of [3] takes at most eleven rounds in the sharing phase.

5.1 The Protocol

Inputs: The dealer has a secret s. Let D be the dealer and let $F(x, y)$ be a *symmetric* bivariate polynomial of degree t in each variable. Let $F(0, 0) = s$.
Sharing Phase
Round 1: Let $f_i(x)$ be defined as $F(i, x)$. Let $r_{ij} \in_R \mathbb{F}$ for each P_i, P_j. Execute $\mathsf{ICP}_{\mathsf{sh}}(D, P_i, f_i(j))$, $\mathsf{ICP}_{\mathsf{sh}}(P_i, P_j, r_{ij})$ and $\mathsf{ICP}_{\mathsf{sh}}(P_i, D, r_{ij})$. Let the corresponding values received be $f'_i(j)$, r'_{ij} and r^D_{ij}.
Round 2:

1. P_i broadcasts $a_{ij} = f'_i(j) + r_{ij}$ and $b_{ij} = f'_i(j) + r'_{ji}$.
2. D broadcasts $a^D_{ij} = f_i(j) + r^D_{ij}$ and $b^D_{ij} = f_i(j) + r^D_{ji}$.
3. If P_i received $f'_i(x)$ which is not a polynomial of degree t, then P_i executes $\mathsf{ICP}_{\mathsf{rec}}(D, P_i, f'_i(j)$ for all j.

Round 3:

1. If D *conflicts* with P_i or $a_{ij} \neq a^D_{ij}$ or $a_{ij} = \perp$, then D broadcasts $f^D_i(x) = f_i(x)$ and executes $\mathsf{ICP}_{\mathsf{rec}}(P_i, D, r^D_{ik})$ and $\mathsf{ICP}_{\mathsf{rec}}(P_k, D, r^D_{ki})$ for all k.
2. If P_i *conflicts* with P_j or $a_{ij} \neq b_{ji}$ or $a_{ji} \neq b_{ij}$ or $a_{ij} \neq a^D_{ij}$ or $b_{ij} \neq b^D_{ij}$, then P_i executes $\mathsf{ICP}_{\mathsf{rec}}(D, P_i, f'_i(j))$ and $\mathsf{ICP}_{\mathsf{rec}}(P_j, P_i, r'_{ji})$.
3. If P_i *conflicts* with D, then he executes $\mathsf{ICP}_{\mathsf{rec}}(D, P_i, f'_i(k))$, for all k.

Round 4: Corresponding $\mathsf{ICP}_{\mathsf{rec}}$ executions are completed in this round.
Local Computation: D is **discarded** if for some P_i, P_j, at least one of the following does not hold:

1. $\{f'_i(k)\}_k$ lie on a t-degree polynomial.
2. $f^D_i(j) = f^D_j(i) = f'_i(j) = f'_j(i)$.
3. $a^D_{ij} = b^D_{ji} = f^D_j(i) + r^D_{ij}$.
4. All $\mathsf{ICP}_{\mathsf{rec}}(D, P_i, r^D_{ij})$ reveals were successful (i.e., at least $t + 1$ accepts were broadcasted).

Reconstruction Phase: Every P_i executes (if they haven't already) $\mathsf{ICP}_{\mathsf{rec}}(D, P_i, f_i(j))$, $\mathsf{ICP}_{\mathsf{rec}}(P_j, P_i, r_{ji})$ for all P_j.
Local Computation: Let $P_i \in \mathsf{U}$ if D broadcasted $f^D_i(x)$. Construct Rec in the following way:

1. $P_i \in \mathsf{Rec}$ if $P_i \in \mathsf{U}$. In this case, define $f_i'(x) = f_i^D(x)$.
2. $P_i \in \mathsf{Rec}$ if he successfully executed $\mathsf{ICP}_{\mathsf{rec}}(D, P_i, f_i(j))$ for all j, and they lie on a t-degree polynomial.

Delete $P_i \notin \mathsf{U}$ from Rec if

1. P_i successfully revealed $f_i'(j)$ and $f_i'(j) \neq f_j^D(i)$ for some $P_j \in \mathsf{U}$.
2. P_j successfully revealed r_{ij}' and $f_i'(j) + r_{ij}' \neq a_{ij}$.
3. If for some P_j, P_j did not *conflict* with P_i and $b_{ij} - r_{ji}' \neq f_i'(j)$.

Reconstruct a symmetric bivariate polynomial $F'(x,y)$ of degree t from $\{f_i'(x)\}_{P_i \in \mathsf{Rec}}$. Output $s' = F'(0,0)$.

5.2 Proofs

Note that in our 4-Round-VSS protocol, ICP properties **Correctness1**, **Correctness2**, **Correctness3** hold for concurrent executions of ICP(P_i, P_j, r_{ij}) and ICP(P_i, D, r_{ij}). Also when D is honest, **Secrecy** holds for concurrent executions of ICP(P_i, P_j, r_{ij}) and ICP(P_i, D, r_{ij}).

The following lemma is proved by means of a standard argument.

Lemma 4. *(**Secrecy**) Protocol 4-round-VSS satisfies perfect secrecy.*

Claim 7. If D is not discarded and P_i is honest, then for every $P_j \in \mathsf{U}$, $f_i'(j) = f_j^D(i)$.

Proof. If $P_i \in \mathsf{U}$, then $f_i'(x) = f_i^D(x)$, and since D is not discarded, the claim holds. Now let $P_i \notin \mathsf{U}$. Recall that $P_j \in \mathsf{U}$ because D *conflicted* with P_j (over some value $f_j(k)$) OR because $a_{jk} \neq a_{jk}^D$ OR $a_{jk} = \perp$. As a result D reveals r_{ij} (Round 3 Step 1). Recall that $P_i \notin \mathsf{U}$. Therefore, w.h.p, his reveals are successful. Now there are two cases to consider. First, if P_i conflicts with D, then he reveals $f_i'(k)$ as well (Round 3 Step 3). If $f_i'(j) \neq f_j^D(i)$, then D is discarded (see Local Computation). On the other hand, if P_i did not conflict with D, then D has to reveal the correct value of r_{ij} (follows from **Correctness3**), i.e. $r_{ij}^D = r_{ij}$. Since $P_i \notin \mathsf{U}$, we have $a_{ij}^D = a_{ij}$. Therefore, for an honest P_i, we have $a_{ij}^D - r_{ij}^D = a_{ij} - r_{ij} = f_i'(j)$. If $a_{ij}^D - r_{ij}^D \neq f_j^D(i)$, then D is discarded (see Local Computation). Therefore, $f_i'(j) = f_j^D(i)$.

Claim 8. If D is not discarded and P_i is honest, then $P_i \in \mathsf{Rec}$.

Proof. If $P_i \in \mathsf{U}$, then $P_i \in \mathsf{Rec}$ by construction. Honest $P_i \notin \mathsf{U}$ successfully reveals $f_i'(j))$ for all j. We now show that none of rules that delete P_i from Rec apply to an honest P_i.

1. By Claim 7, we have that for each $P_j \in \mathsf{U}$, $f_i'(j) = f_j^D(i)$.
2. Since revealed r_{ij}' is equal to r_{ij} w.h.p (by **Correctness3**), $a_{ij} = f_i'(j) + r_{ij}'$.
3. If P_j did not conflict with P_i, then an honest P_i will be successful in revealing the pad r_{ji}' (by **Correctness2**). Hence $b_{ij} - r_{ji}' = f_i'(j)$.

Claim 9. If D is not discarded, then $f_i'(j) = f_j'(i)$ for every honest P_i, P_j.

Proof. Recall that when $P_i \in \mathsf{U}$, $f_i'(x) = f_i^D(x)$. When both P_i and P_j are in U, then the claim follows directly. Now suppose $P_i, P_j \notin \mathsf{U}$. For honest P_i, P_j , if $f_i'(j) \neq f_j'(i)$, then $a_{ij} \neq b_{ji}$ and $a_{ji} \neq b_{ij}$. Consequently, P_i, P_j would have successfully revealed $f_i'(j), f_j'(i)$ respectively (by **Correctness2**). Since we assume that D is not discarded, the claim follows in this case too.

Lastly, consider the case when exactly one of P_i, P_j is contained in U. W.l.o.g, let $P_i \notin \mathsf{U}$,$P_j \in \mathsf{U}$. If $f_i'(j) \neq f_j^D(i)$, then P_i would have been deleted from Rec. But by Claim 8, we have honest $P_i \in \mathsf{Rec}$. Therefore, the claim must hold.

Recall that there are at least $t+1$ honest players, and by Claim 8 all of them are contained in Rec. By Claim 9, the shares of these honest players are consistent. The following claim is now easy to see:

Claim 10. If D is not discarded then all honest parties are consistent with an unique t-degree symmetric bivariate polynomial, say $F^H(x, y)$.

Claim 11. If D is not discarded and $P_i \in \mathsf{Rec}$, then $f_i'(x)$ is consistent with $F^H(x, y)$.

Proof. By Claim 7, for every $P_i \in \mathsf{U}$, $f_i^D(x)$ is consistent with all the honest players' shares. This implies that $f_i'(x)$ is consistent with $F^H(x, y)$.

Now let $P_i \notin \mathsf{U}$. Since $P_i \in \mathsf{Rec}$, we have $f_i'(j) = f_j^D(i)$ for every $P_j \in \mathsf{U}$ (otherwise, P_i is deleted from Rec). Therefore, if $f_i'(x)$ is inconsistent with $F^H(x, y)$, then $f_i'(j) \neq f_j'(i)$ must hold for some *honest* $P_j \notin \mathsf{U}$. If $a_{ij} \neq b_{ji}$ or $a_{ji} \neq b_{ij}$, then P_i, P_j would reveal $f_i'(j), f_j'(i)$ respectively. Since D was not discarded, we have $f_i'(j) = f_j'(i)$. For the rest of the proof, we assume $a_{ij} = b_{ji}$ and $a_{ji} = b_{ij}$.

If P_i had a conflict with P_j, then P_i reveals $f_i'(j)$. If P_j also had a conflict with P_i, then P_j would have revealed $f_j'(i)$. Since D was not discarded, we have $f_i'(j) = f_j'(i)$. On the other hand, if P_j did not have a conflict with P_i, then P_i would have to reveal $r_{ji}' = r_{ji}$ (follows from **Correctness3**) Since P_j is honest, $b_{ij} - r_{ji} = f_j'(i)$. If $P_i \in \mathsf{Rec}$, then $b_{ij} - r_{ji}' = f_i'(j)$. Since $r_{ji}' = r_{ji}$, this shows that $f_i'(j) = f_j'(i)$. Hence $f_i'(x)$ is consistent with $F^H(x, y)$.

On the other hand if P_i did not have a conflict with P_j, an honest P_j would successfully reveal r_{ij}' . Since $a_{ij} = b_{ji} = f_j'(i) + r_{ij}'$, P_i would have to reveal $f_i'(x)$ such that $f_i'(j) = f_j'(i)$, otherwise $a_{ij} \neq f_i'(j) + r_{ij}'$, and P_i will be deleted from Rec.

Since $F^H(x, y)$ can be computed from the joint view of the honest parties at the end of the sharing phase, the following claim holds.

Claim 12. If D is not discarded, then $F^H(x, y)$ will be reconstructed in the reconstruction phase. Moreover, this $F^H(x, y)$ is fixed at the end of the sharing phase.

It is easy to see that an honest D is never disqualified. Given this, the next two lemmas follow directly from Claim 12, and the theorem follows from Lemmas 4, 5 and 6.

Lemma 5. *(**Correctness**) Protocol 4-Round-VSS satisfies* $(1-\varepsilon)$*-correctness property.*

Lemma 6. *(**Strong Commitment**) Protocol 4-Round-VSS satisfies* $(1-\varepsilon)$*-strong commitment property.*

Theorem 2. *There exists an efficient 4-round sharing, 2-round reconstruction* $(2t+1, t)$ *statistical VSS protocol.*

Acknowledgements

This is joint work with Arpita Patra and C. Pandu Rangan. We thank Jonathan Katz for fruitful collaboration during early stages of this research.

References

1. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *20th Annual ACM Symposium on Theory of Computing (STOC)*, pages 1–10. ACM Press, 1988.
2. B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *26th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 383–395. IEEE, 1985.
3. Ronald Cramer, Ivan Damgård, Stefan Dziembowski, Martin Hirt, and Tal Rabin. Efficient multiparty computations secure against an adaptive adversary. In Jacques Stern, editor, *Advances in Cryptology — Eurocrypt '99*, volume 1592 of *LNCS*, pages 311–326. Springer, 1999.
4. D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly secure message transmission. *Journal of the ACM*, 40(1):17–47, 1993.
5. Matthias Fitzi, Juan A. Garay, Shyamnath Gollakota, C. Pandu Rangan, and K. Srinathan. Round-optimal and efficient verifiable secret sharing. In *3rd Theory of Cryptography Conference — TCC 2006*, volume 3876 of *LNCS*, pages 329–342. Springer, 2006.
6. Rosario Gennaro, Yuval Ishai, Eyal Kushilevitz, and Tal Rabin. The round complexity of verifiable secret sharing and secure multicast. In *33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 580–589. ACM Press, 2001.
7. J. Katz, C.-Y. Koo, and R. Kumaresan. Improving the round complexity of VSS in point-to-point networks. *Information and Computation*, 207(8):889–899, 2009.
8. A. Patra, A. Choudhary, T. Rabin, and C.P. Rangan. The round complexity of verifiable secret sharing revisited. In *Advances in Cryptology — Crypto 2009*, volume 5677 of *LNCS*, pages 487–504. Springer, 2009.
9. Arpita Patra, Ashish Choudhary, and C. Pandu Rangan. Round efficient unconditionally secure multiparty computation protocol. In Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das, editors, *INDOCRYPT*, volume 5365 of *Lecture Notes in Computer Science*, pages 185–199. Springer, 2008.
10. Arpita Patra, Ashish Choudhary, and C. Pandu Rangan. Simple and efficient asynchronous byzantine agreement with optimal resilience. In Srikanta Tirthapura and Lorenzo Alvisi, editors, *PODC*, pages 92–101. ACM, 2009.
11. Tal Rabin and Michael Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority. In *21st Annual ACM Symposium on Theory of Computing (STOC)*, pages 73–85. ACM Press, 1989.