Lower Bounds for Round and Communication Complexities of Unconditional Verifiable Secret Sharing

Srivatsan Narayanan, Ananth Raghunathan, and Pandu Rangan

Dept. of Computer Science and Engineering
Indian Institute of Technology Madras
{nsrivats,ananthr}@cse.iitm.ac.in, prangan@iitm.ac.in

Abstract. Verifiable Secret Sharing (VSS) is a fundamental primitive used as a building block in many distributed cryptographic tasks, such as Multiparty Computation and Byzantine Agreement. VSS is a two phase (Share, Reconstruct) protocol carried out among n parties in the presence of a computationally unbounded adversary who can corrupt up to t parties. We assume that every two parties in the network are connected by a pairwise secure channel and a public broadcast channel is available to all.

We analyze two important notions of complexity of Verifiable Secret Sharing schemes—round complexity and communication complexity. The round complexity of VSS is a well studied problem [12, 11] and lower bounds and corresponding optimal protocols have been discovered. But recent work by Patra et al., in [19] have provided *new* protocols that break traditional resilience bounds on VSS by introducing a negligible error in the properties of VSS (statistically-secure VSS).

In this paper, we provide the first formal definition and framework to study the lower bound of SVSS schemes. We also show the impossibility of a weaker notion of (n, t)-Byzantine Agreement, a 2-round (n, t)-VSS scheme for $n \leq 3t$, and provide an overview of the impossibility of 3-round (n, t)-SVSS for $n < 3t$.

We relax the assumption that we have a pair-wise secure channel and show some interesting lower bounds on the *connectivity* of the underlying *communication graphs* of (n, t)-VSS schemes for $n \geq 3t + 1$. We use these results to study the communication complexity and establish a lower bound for *perfectly-secure* (n, t)-VSS schemes ($n \geq 3t + 1$). This shows the optimality of the 4-round protocol in [12].

1 Introduction

VSS (Verifiable Secret Sharing) [8] is a fundamental primitive used as a building block in many distributed cryptographic tasks. VSS is a two phase protocol (sharing and reconstruction) carried among n parties in the presence of computational unbounded adversaries.

The round complexity of interactive protocols is one of their most important complexity measures. Interaction over computer networks is usually the most time consuming operation (because of lagging or network congestion). It is thus very important to devise protocols which require the minimal number of rounds to complete. Consequently substantial research efforts has been invested into characterizing the rounds complexity of various tasks, particularly, [1, 4, 3, 10, 15, 22, 2, 7, 16].

Another important measure of the performance of distributed protocols is the *communication complexity* between different nodes. In particular, related to Multi-party Computation, and VSS, the following papers deal with communication complexity – [6, 14, 5, 13, 20].

1.1 Previous Work

The round complexity of *perfectly-secure* VSS was studied by Gennaro et. al in [12] where they established lower bounds for the cases when $n \geq 3t + 1$ and $n \geq 4t + 1$. Recently, [19] provided *upper* bounds by

relaxing the security requirements of VSS (statistically secure VSS)—to include a negligible probability of error. They showed that we are able to break the traditional $n \geq 3t + 1$ resilience, and provide protocols for $n \leq 3t$.

Fitzi et. al in [11] provided the first round-optimal VSS protocol ($n \geq 3t + 1$), whose communication complexity is $O(n^3)$. However, the 4-round protocol in [12] has a complexity of $O(n^2)$.

1.2 Our contributions

In this paper, we provide the first formal parameterized definition of statistically-secure VSS and show that the protocols in [19] are round-optimal. In particular, we show that we require at least 3 rounds for a $(3, 1)$-SVSS protocol, and can extend this to 4 rounds if $t > 1, n \leq 3t - 1$. We also show that a weaker notion of Byzantine Agreement, where non-dealer players are allowed to broadcast is *impossible* for $n \leq 3t$, for *any number* of rounds. This is the first formal, and rigorous proof of the above lower bounds to the best of the authors' knowledge.

We also explore for the first time the *connectivity* requirements for VSS and establish lower bound on the connectivity of the underlying *communication graph*. We use these results to show a quadratic lower bound on the communication complexity of VSS/WSS, showing that the (round optimal) WSS protocol in [11] and the VSS protocol in [12] are communication optimal at optimal resilience.

Organization. In section 2, we provide a formal definition of statistically secure VSS, with several variants. In section 3, we show that if we restrict the dealer from broadcasting, even statistically-secure VSS is *impossible* for any rounds, when $n \leq 3t$. In section 4, we show a 3-round lower bound for $(3, 1)$-SVSS. Section 5 deals with a 4-round lower bound for the case when $t > 1$ and $n < 3t$. In section 6, we establish the quadratic lower bound on communication complexity, and allied results in the connectivity requirements for VSS.

2 Definition of SVSS

In this section, we give the formal definition of statistically-secure VSS. Here, we introduce several *new notions* of commitment, which are obtained by "relaxing" the definition of commitment in *perfect* VSS schemes in different ways.

2.1 Player and Adversary Model for SVSS

We consider a synchronous network of n processors, denoted by $\mathcal{P} = \{P_1, \ldots, P_n\}$, which will be referred to as *players*. Each pair of players is connected via a private, authenticated, point-to-point channel. In addition, all players share a common *broadcast channel* (unless specified otherwise).

In VSS protocols, a distinguished player $\mathcal{D}$ (say $\mathcal{D} = P_1$) is referred to as *dealer*. The protocol consists of two phases: a *sharing phase* and a *reconstruction phase*.

Sharing: Initially, the dealer $\mathcal{D}$ holds an input s referred to as the *secret*, and each player P_i holds an independent random input r_i. The sharing phase might comprise of several *rounds*. At each round, each player communicates privately, and can broadcast a message. Each message sent or broadcasted by P_i is determined by its input, its randomness, and messages received from other players in previous rounds.

Reconstruction: In this phase, each player P_i provides its entire view $\mathcal{V}_i$ from the sharing phase, and a reconstruction function $\mathbf{Rec}(\mathcal{V}_1, \cdots, \mathcal{V}_n)$ is applied and taken as the protocol's output.[1]

Let t be a *security threshold*. A *t-adversary* may choose any set of t players to be corrupted during the entire execution of the protocol (including both phases). We assume that the adversary is computationally unbounded, and that it attains full control of corrupted players. In particular, the corrupted players might deviate from the protocol in a co-ordinated manner in an attempt to violate its guaranteed properties (defined below). Moreover, at each round, the adversary may wait until it receives all messages from honest players before sending its own messages.

2.2 Properties of SVSS

A two-phase n-player protocol as above is called a *statistical* (n, t)-VSS protocol if, for any t-adversary, the following requirements hold:
Perfect Secrecy or 0-secrecy: If the dealer $\mathcal{D}$ is honest, then the adversary does not learn any information about the secret s, during the Sharing Phase. The view of the adversary at the end of the Sharing Phase is distributed identically for all secrets s. We use 0-secrecy to denote that the probability of error is zero in this case, to make it consistent with the definitions below. For a note on relaxing secrecy, refer to Appendix C.
ε-Correctness: If the dealer is honest, s is always reconstructed by the honest players with overwhelmingly high probability, irrespective of what the adversary does. To put it mathematically, for some negligible quantity ε,

$$\mathop{\mathbf{Pr}}_{r_{\text{dealer}}, r_{\text{adversary}}, r_{\text{honest players}}} [s \text{ is reconstructed}] \geq 1 - \varepsilon.$$

Commitment: We consider a state at the end of the *Sharing Phase* to be **committed** if the view of all the players defines a unique secret s^* that can be reconstructed. As can be seen below, we introduce a probability of error in both phases.

Definition 1 ((δ_s, δ_r)-commitment) *At the end of the sharing phase, we arrive at a committed state with probability $\geq 1 - \delta_s$. However, once we arrive at a committed state, unlike in perfect VSS, the probability that s^* is reconstructed subsequently is $\geq 1 - \delta_r$. Mathematically:*

$$\mathop{\mathbf{Pr}}_{r_{\text{Sharing}}} [\textit{ A committed state is reached }] \geq 1 - \delta_s,$$

$$\mathop{\mathbf{Pr}}_{r_{\text{Reconstruction}}} [\textit{ } s^* \textit{ is reconstructed } | \textit{ A committed state is reached }] \geq 1 - \delta_r.$$

Here, r_{Sharing} and $r_{\text{Reconstruction}}$ are the random coins of all players in each phase.

There are two (stronger) variants of commitment, where δ_s and δ_r are respectively zero. This leads to two other notions which can be referred to as "always almost-committed" and "almost-always committed" protocols. We can see that by setting both δ_s and δ_r to zero, we get the definition of commitment used in perfect VSS, which is the strongest notion of commitment.
Note: For our proofs, we'll assume a general (δ_s, δ_r)-commitment, since it is the weakest of the four variants of commitment, and for lower bounds, it suffices to establish them for the weakest variants.

[1] A physical realization of the reconstruction stage may use either private channels or broadcast channels to exchange the views v_i between the players, so that **Rec** can be locally computed by each player. It will follow from the semantics of VSS that the local outputs of all good players will be identical, even when the views are exchanged via private channels.

3 Impossibility of $n \leq 3t$ SVSS without dealer broadcast

In this section, we show that if a protocol Π does not have any dealer broadcast (the players can however broadcast), then $(3, 1)$-SVSS is impossible. We can then use the standard *player partitioning argument* [18, 8] to get the following result.

Theorem 1. *(n, t)-SVSS is impossible for $n \leq 3t$ for any protocol Π without dealer broadcast.*

3.1 Proof Outline

We provide a proof by contradiction. We assume that there is a protocol Π for $(3, 1)$-SVSS and show that ε-correctness implies that (δ_s, δ_r)-commitment (for some parameters δ_s, δ_r) is violated, hence contradicting the fact that Π, a $(3, 1)$-SVSS protocol satisfies the two conditions simultaneously.

We employ the *hybrid games* argument to construct a series of games that are indistinguishable to an honest player. However, these games lead to a situation where commitment is violated. In games $\mathcal{H}$ and $\mathcal{H}'$, a corrupt dealer $\mathcal{D}$ pretends to share the secret s with the first player A, and the secret s' with the second player B. In the proof that follows, we show that *as long as $\mathcal{D}$ does not broadcast*, he does not ever need to commit to one of the two secrets.

In particular, as far as A is concerned, his behavior is identical to an honest $\mathcal{D}$ sharing s and a corrupt B (described in game $\mathcal{G}$). And similarly, B is unable to distinguish his behavior from a game $\mathcal{G}'$ where an honest $\mathcal{D}$ shares s' and A is corrupt. However, since Π satisfies correctness (albeit with a negligible probability of error), in games $\mathcal{G}$ (and $\mathcal{G}'$), honest players reconstruct s (and s') during reconstruction.

Now it is easy to see that, if in game $\mathcal{H}$, during reconstruction $\mathcal{D}$ behaves like the honest dealer $\mathcal{D}$ in $\mathcal{G}$ and hence s is reconstructed. But if during reconstruction, he behaves like the honest dealer $\mathcal{D}$ in $\mathcal{G}'$, s' must be reconstructed. This violates even the weakest notion of commitment. The section below describes the games in detail, and analyzes the required probabilities.

3.2 A Detailed Proof

Games $\mathcal{H}$ and $\mathcal{H}'$: The dealer $\mathcal{D}$ maintains two internal states $\mathcal{T}$ and $\mathcal{T}'$ (which are referred to as transcripts) that detail the interactions with A and B in all the rounds so far. The sharing phase is identical for $\mathcal{H}$ and $\mathcal{H}'$. We assume that each transcript is *initialized* with the dealer randomness ($r_{\mathcal{D}}$ or $r'_{\mathcal{D}}$) and the secret to be shared (s or s' respectively). All other internal variables are functions of the data in each transcript. Hence, the data in the transcript completely describes the player's behavior.

The dealer $\mathcal{D}$ chooses random coins $r_{\mathcal{D}}$ and $r'_{\mathcal{D}}$. In transcript $\mathcal{T}$ he pretends to follow the protocol Π correctly, with his initial randomness $r_{\mathcal{D}}$ and secret s. However, for every message received from B, he replaces it with $\mathbf{0}$ – a default all-zero message. His transcript is as below. Every time he sends a private message to A, he uses the information in $\mathcal{T}$. Here $m_{\mathsf{A}}^{(i)}$ and $m_{\mathsf{B}}^{(i)}$ are private messages from players A and B in the i^{th} round. And $\beta^{(i)}$ is the collective broadcasts by players in the i^{th} round.

$\mathcal{T}$	
Round	Messages
0	$r_{\mathcal{D}}, s$
1	$m_{\mathsf{A}}^{(1)}, \mathbf{0}, \beta^{(1)}$
2	$m_{\mathsf{A}}^{(2)}, \mathbf{0}, \beta^{(2)}$
$\vdots$	$\vdots$

$\mathcal{T}'$	
Round	Messages
0	$r'_{\mathcal{D}}, s'$
1	$\mathbf{0}, m_{\mathsf{B}}^{(1)}, \beta^{(1)}$
2	$\mathbf{0}, m_{\mathsf{B}}^{(2)}, \beta^{(2)}$
$\vdots$	$\vdots$

Similarly, he replaces messages from A with $\mathbf{0}$ and uses randomness $r'_{\mathcal{D}}$ and secret s' in transcript $\mathcal{T}'$ to interact with B. This is to make sure that his malicious behavior is indistinguishable from a corrupt player's behavior (as in the game described below).

During reconstruction, in game $\mathcal{H}$, the dealer $\mathcal{D}$ discards the transcript $\mathcal{T}'$ and pretends he was an honest dealer sharing s. The view revealed during reconstruction is the view got from $\mathcal{T}$. In game $\mathcal{H}'$ similarly, he discards $\mathcal{T}$ and reveals a view based on $\mathcal{T}'$.

Game $\mathcal{G}$: As mentioned earlier, we need to show a game where B is corrupt and yet A is unable to distinguish between the behavior of a corrupt dealer $\mathcal{D}$ (as above) and a corrupt B (with an honest dealer). The dealer $\mathcal{D}$ holds the secret s.

The player B, being corrupt, must simulate the transcript $\mathcal{T}'$. Whenever he sends private messages to A or broadcasts a message, he discards all the information $\mathcal{D}$ gives him about the protocol so far. Instead, he randomly chooses $r'_{\mathcal{D}}$ and s' to initialize the transcript $\mathcal{T}'$. Since B doesn't know what the private messages A sent to $\mathcal{D}$, he replaces them by $\mathbf{0}$. This allows B to simulate $\mathcal{T}'$, and hence $\mathcal{D}$'s behavior as if it were game $\mathcal{H}$.

Therefore, according to the above construction, as far as player A is concerned, $\mathcal{H}$ and $\mathcal{G}$ are two identical games. In one, the dealer is corrupt, whereas in the other, the player B successfully simulates a corrupt dealer.

From correctness, we get that:

$$\mathop{\mathbf{Pr}}_{r_{\mathcal{D}},r_{\mathsf{A}},r_{\mathcal{D}'},r_{\mathsf{B}}}[s \text{ is reconstructed in game } \mathcal{G}] \geq 1-\varepsilon, \tag{1}$$

where ε is a negligible quantity and $r_{\mathcal{D}'}, r_{\mathsf{B}}$ is the player randomness.

Game $\mathcal{G}'$: In this game, we assume that player A is corrupt. The dealer shares the secret s'. As described above, the player A behaves as how B does in game $\mathcal{G}$. (Except that, in this case, $r_{\mathcal{D}}$ and r_{A} are the player's random coins).

From correctness, we once again get that:

$$\mathop{\mathbf{Pr}}_{r_{\mathcal{D}},r_{\mathsf{A}},r_{\mathcal{D}'},r_{\mathsf{B}}}\left[s' \text{ is reconstructed in game } \mathcal{G}'\right] \geq 1-\varepsilon, \tag{2}$$

where ε is a negligible quantity and $r_{\mathcal{D}}, r_{\mathsf{A}}$ is the player randomness.

Extracting the probability of reconstruction. The above result for correctness is over all possible random coins during sharing and reconstruction. If we define sets $\mathcal{R}_s$ and $\mathcal{R}_r$ to denote the randomness during sharing and reconstruction[2], equation (1) implies that there exists a set $\mathcal{W} \subseteq \mathcal{R}_s$ of cardinality at least $(1-\sqrt{\varepsilon}) \cdot |\mathcal{R}_s|$ such that:

$$\mathop{\mathbf{Pr}}_{r_{\text{Reconstruction}}}[s \text{ is reconstructed} | r_{\text{Sharing}} \in \mathcal{W}] \geq 1-\sqrt{\varepsilon}. \tag{3}$$

This is easy to see, because if $|\mathcal{W}|$ is $< (1-\sqrt{\varepsilon}) \cdot |\mathcal{R}_s|$, we can come up with at least $\sqrt{\varepsilon} \cdot \sqrt{\varepsilon} = \varepsilon$ fraction of instances where correctness is violated, which leads to a contradiction.

Similarly, we can construct a set $\mathcal{W}'$ corresponding to equation (2). Now, we note that $\mathcal{W}$ and $\mathcal{W}'$ are defined over the same set of random coins. Since $|\mathcal{W}| \geq (1-\sqrt{\varepsilon}) \cdot |\mathcal{R}_s|$, and $|\mathcal{W}'| \geq (1-\sqrt{\varepsilon}) \cdot |\mathcal{R}_s|$, $|\mathcal{W} \cap \mathcal{W}'| \geq (1-2\sqrt{\varepsilon}) \cdot |\mathcal{R}_s|$. We can conclude thus that, with probability at least $(1-2\sqrt{\varepsilon})$ we arrive at a state (after Sharing Phase) such that:

[2] Note that $(r_{\mathcal{D}}, r_{\mathsf{A}}, r'_{\mathcal{D}}, r_{\mathsf{B}}) \in \mathcal{R}_s \times \mathcal{R}_r$

$$\underset{r_{\text{Reconstruction}}}{\mathbf{Pr}}\left[s \text{ (or } s') \text{ is reconstructed}\right] \geq 1 - \sqrt{\varepsilon}. \tag{4}$$

Breaking commitment. At the end of the sharing phase, $\mathcal{D}$ has a choice between s and s'. Let us assume he randomly chooses to side with either A or B. Therefore, we get (when the dealer is corrupt):

$$\begin{aligned}
\underset{r_{\text{Reconstruction}}}{\mathbf{Pr}}\left[s \text{ is reconstructed}\right] &= \underset{r_{\text{Reconstruction}}}{\mathbf{Pr}}\left[s \text{ is reconstructed in } \mathcal{H}\right] \cdot \mathbf{Pr}\left[\mathcal{H} \text{ is played}\right] \\
&= \underset{r_{\text{Reconstruction}}}{\mathbf{Pr}}\left[s \text{ is reconstructed in } \mathcal{H}\right] \cdot \left(\frac{1}{2}\right) \\
&\geq \frac{1}{2}\left(1 - \sqrt{\varepsilon}\right) \qquad \text{(from equation(4))}
\end{aligned}$$

Along the same lines, we also get that $\mathbf{Pr}_{r_{\text{Reconstruction}}}[s' \text{ is reconstructed}] \geq \frac{1}{2}(1 - \sqrt{\varepsilon})$. Thus the probability that either s or s' is reconstructed is $\geq 1/2 - \sqrt{\varepsilon}/2$, and hence neither of them can be considered s^* (because the other option occurs with the above non-negligible probability). From the previous discussion, we have seen that we arrive at such a state for randomness $\in \mathcal{W}_s \cap \mathcal{W}'_s$, translating to a probability of $(1 - 2\sqrt{\varepsilon})$. Therefore, we see that we are also able to break (δ_s, δ_r)-commitment for any $0 \leq \delta_s \leq 1 - 2\sqrt{\varepsilon}$ and $0 \leq \delta_r \leq \sqrt{\varepsilon}$. (If we define δ_r, δ_s and ε all to be negligible quantities, then this result is true for *all* δ_r, δ_s and ε).

3.3 Extending this to Byzantine Agreement

We can actually show something stronger with the above proof. In particular, if we define weakened Byzantine Agreement to be Byzantine Agreement with the additional property that every player $P \neq \mathcal{D}$ can broadcast, we get:

Theorem 2. *Statistically-secure weakened (n, t)-Byzantine Agreement is impossible for $n \leq 3t$.*

For proof, please see Appendix A.

3.4 Deterministic Reconstruction

We can in fact modify the definition of commitment so that during reconstruction, the protocol is deterministic. We claim that:

Lemma 1. *For every protocol Π that is a (n, t)-SVSS scheme with parameters $(\varepsilon, \delta_s, \delta_r)$, we can create a new scheme Π' such that all players who follow the protocol honestly will not use any randomness during reconstruction. Also, Π' will also be a (n, t)-SVSS scheme with slightly weaker parameters $(\varepsilon, \delta_s + \delta_r(1 - \delta_s), \delta_r)$.*

For proof, please see Appendix B. In light of this lemma, we only consider protocols whose honest reconstruction is deterministic.

4 Lower bound for (n, t)-SVSS for $n \leq 3t$

In this section, we show that for (n, t)-SVSS protocols where $n \leq 3t$ we require at least 3 rounds during the sharing phase. By the standard player partitioning argument [18, 8] this reduces to showing the following theorem:

Theorem 3. *There exists no 2-round* $(3, 1)$*-SVSS scheme.*

4.1 Proof Outline

The proof proceeds with four games $\mathcal{G}_1$ through $\mathcal{G}_4$. In game $\mathcal{G}_1$, the dealer honestly tries to share the secret s, and B is corrupt. Irrespective of whether B behaves honestly or not, honest players will reconstruct s with high probability (due to correctness). $\mathcal{D}$ in the next game will try to simulate this B.

In game $\mathcal{G}_2$, the dealer is corrupt. He behaves adversarially, so as to make it impossible for A to distinguish whether it is B or $\mathcal{D}$ who is corrupt. But during reconstruction, $\mathcal{D}$ behaves honestly, and reconstructs s by correctness.

In game $\mathcal{G}_3$, the dealer is corrupt, and behaves identically to the dealer in $\mathcal{G}_2$ during the sharing phase. But, during reconstruction, he chooses to output a random or "nonsensical" view. By commitment, since in the previous game, s is reconstructed, s must be reconstructed in this game as well.

In game $\mathcal{G}_4$, it is A who is corrupt this time. He waits for an honest execution from $\mathcal{D}$ and B till the end of the sharing phase. But *before* reconstruction, he is going to mentally simulate the other two players so that in his mind, the game $\mathcal{G}_3$ is played out. We will show that this breaks secrecy.

Overall, we see that in games $\mathcal{G}_1$ and $\mathcal{G}_2$ we are making sure that B is irrelevant to the protocol (i.e., B does not contribute to reconstructing the secret). Then in game $\mathcal{G}_3$ we also see that this must be possible if $\mathcal{D}$ himself also does not participate in reconstruction. This logically means that A himself has all the information required to construct the secret, breaking secrecy, which is what $\mathcal{G}_4$ is all about.

2 rounds is important here because in round 2, B is going to complain (as per the protocol) that he has been dealt a bad share, but without a third round, $\mathcal{D}$ need not defend himself, and A will not know whether to trust B or $\mathcal{D}$.

4.2 Game $\mathcal{G}_1$

In game $\mathcal{G}_1$, the player B is corrupt. As in the previous section, we consider a transcript $\mathcal{T}$ of all the information a player sees as the rounds proceed. In this game, the player B, irrespective of private messages from $\mathcal{D}$, replaces them with a default all-zero $\mathbf{0}$ string. Therefore, B's (modified) transcript is as follows.

$\mathcal{T}^*_{\mathsf{B}}$	
Round	Messages
0	r_{B}
1	$m^{(1)}_{\mathsf{A}}, \mathbf{0}, \beta^{(1)}$
2	$m^{(2)}_{\mathsf{A}}, \mathbf{0}, \beta^{(2)}$

Since he is also malicious, he sends $\mathbf{0}$ messages to $\mathcal{D}$ whenever it is his turn to communicate. And, whenever he sends a private message to A or broadcasts, he follows the protocol with $\mathcal{T}^*_{\mathsf{B}}$ as the transcript. Effectively, he acts as if he *does not* get a share of the secret s. A's view (and behavior) are identical to an honest execution of the protocol.

Therefore, to summarize, in this game, A and $\mathcal{D}$ proceed to interact honestly according to Π. B interacts with $\mathcal{D}$ as above, and A and B interact as if $\mathcal{D}$ were sending $\mathbf{0}$ messages to B. We'll see that this game can be simulated by a corrupt $\mathcal{D}$. We get from correctness:

$$\Pr_{r_{\mathsf{A}}, r_{\mathsf{B}}, r_{\mathcal{D}}} [s \text{ is reconstructed in } \mathcal{G}_1] \geq 1 - \varepsilon \tag{5}$$

4.3 Game $\mathcal{G}_2$

In game 2, we see that the dealer is corrupt. As in previous games, he must somehow set it up so that it looks like B is corrupt, as far as A is concerned. He does this as follows. In his transcript $\mathcal{T}_{\mathcal{D}}$, he replaces all messages sent by B with $\mathbf{0}$. This is in accordance with how a corrupt B would behave in game $\mathcal{G}_1$-type of scenario. And instead of communicating truthfully with B as the protocol dictates, the dealer $\mathcal{D}$ instead sends $\mathbf{0}$ messages with B. Note that in this case, an *honest* B's transcript would appear identical to the one in $\mathcal{G}_1$. And as above, $\mathcal{D}$ behaves as an honest dealer would with A.

Note 1: This does not preclude the honest player B from communicating with A (either privately or via broadcast) to inform him that there is some inconsistency. This is referred to as a "complaint". However, in 2-rounds, we see that $\mathcal{D}$ need not *address* this complaint. Also, whatever an honest B might do can be replicated by a dishonest B in game $\mathcal{G}_1$ (including the false complaint).

Note 2: In the first round, private communication between A and B is independent of the message, and hence, only in the second round can B complain to A. Also, unlike the earlier proof, where we claim that each of A and B are satisfied with their interaction with $\mathcal{D}$ because $\mathcal{D}$ never has to broadcast, in this case, the broadcast from $\mathcal{D}$ in the first and second rounds *can and might* be inconsistent with the view of B. But this doesn't matter, because, as we mentioned earlier, B doesn't have an extra round to force $\mathcal{D}$ to commit.

During reconstruction, $\mathcal{D}$ proceeds to behave identically to the honest dealer in $\mathcal{G}_1$. Since as far as A is concerned, the above two games are indistinguishable, s is reconstructed with the same high probability.

$$\Pr_{r_{\mathsf{A}}, r_{\mathsf{B}}, r_{\mathcal{D}}} [s \text{ is reconstructed in } \mathcal{G}_2] \geq 1 - \varepsilon$$

We have assumed, from Lemma 1 that reconstruction phase in Π does not involve any randomness (by honest players). This implies that with $(1 - \varepsilon)$ probability, we arrive at a committed state where s is reconstructed with probability 1.

4.4 Game $\mathcal{G}_3$

In this game, the dealer behaves identically to the one on $\mathcal{G}_2$. However, during reconstruction, he reveals a view that is completely "nonsensical". Actually, since we want to use a deterministic adversary (cf. Lemma 1), $\mathcal{D}$ outputs a default view $\mathcal{V}_0$. By commitment, we know that there is a s^* that is shared such that s^* is reconstructed with high probability.

Now we consider a corrupt dealer $\mathcal{D}$ who randomly chooses whether to play as in $\mathcal{G}_2$ or $\mathcal{G}_3$. If the two options reconstruct different secrets s_1^* and $s_2^* \neq s_1^*$, then they will each occur with (non-negligible) probability $\frac{1}{2}$. Such a state violates commitment.

So by commitment and correctness, with probability $(1 - \delta_s - \varepsilon)$ after sharing, both the options *must* reconstruct s.

$$\Pr_{r_{\text{Reconstruction}}} \left[s \text{ is reconstructed in game } \mathcal{G}_3 | r_{\text{Sharing}} \in \mathcal{W} \cap \mathcal{W}' \right] = 1. \tag{6}$$

4.5 Game $\mathcal{G}_4$

In this game, we utilize information from the third game, and show that actually, secrecy is broken. We know that in the previous game $\mathcal{D}$ being corrupt gives B no information about the share. We also have shown from commitment that even if $\mathcal{D}$ were to give a default "nonsensical" view $\mathcal{V}_0$, independent of the secret, VSS still claims that s must be reconstructed. Clearly, this means A himself has all the information to reconstruct s. Assume $\mathcal{D}$ is honest and wants to share s. After the two rounds are completed, A's transcript has the following information:

$\mathcal{T}_\mathsf{A}$	
Round	Messages
0	r_A
1	$m_\mathsf{B}^{(1)}, m_\mathcal{D}^{(1)}, \beta$
2	$m_\mathsf{B}^{(2)}, m_\mathcal{D}^{(2)}, \beta$

$\mathcal{T}_\mathsf{B}$	
Round	Messages
0	r_B
1	$m_\mathsf{A}^{(1)}, \mathbf{0}, \beta^{(1)}$
2	$m_\mathsf{A}^{(2)}, \mathbf{0}, \beta^{(2)}$

Using this information, A will try to now simulate $\mathcal{G}_3$ where the dealer is corrupt. A can create *his own* messages trivially, because his behavior in $\mathcal{G}_3$ and $\mathcal{G}_4$, during sharing is identical (and honest). Also, we assume that A knows what protocol B and $\mathcal{D}$ follows.

In order to simulate $\mathcal{G}_3$, A must generate transcripts $\mathcal{T}_\mathsf{B}$. Since $\mathcal{D}$ behaves in a default manner during Reconstruction, we do not need transcript $\mathcal{T}_\mathcal{D}$. In game $\mathcal{G}_3$, $\mathcal{T}_\mathsf{B}$ is as described above. In order to do this, A picks a random r_B, and initializes $\mathcal{T}_\mathsf{B}$. Since $\mathcal{D}$ in any case makes B's private messages to him $\mathbf{0}$, it is independent of r_B. Also, messages that $\mathcal{D}$ sends to A, or $\mathcal{D}$ broadcasts are actually identical to an honest execution, even if $\mathcal{D}$ were corrupt (as in $\mathcal{G}_3$), therefore he simply uses $\mathcal{D}$'s information from the *current, actual* execution of $\mathcal{G}_4$. Also, since $\mathcal{D}$ sends $\mathbf{0}$ to B, A simulates an honest B initialized with r_B that he chooses.

Now, A uses his secret share (i.e., communication from an honest $\mathcal{D}$) and reconstructs his view as an honest player would do for the Reconstruction Phase. Let's call this view $\mathcal{V}_\mathsf{A}$. Since $\mathcal{T}_\mathsf{B}$ is known, and identical to the one in game $\mathcal{G}_3$, $\mathcal{V}_\mathsf{B}$ can also be computed (here is where we assume that the protocol B follows is known to A). Now, A chooses the default view $\mathcal{V}_0$ to be $\mathcal{D}$'s view, and proceeds with the reconstruction in his simulation.

As per the construction, since s is legitimately shared with A (in both $\mathcal{G}_3$ and $\mathcal{G}_4$), $\mathcal{V}_\mathsf{A}$ in either case will be equal. Also, by construction, $\mathcal{T}_\mathsf{B}$ and hence $\mathcal{V}_\mathsf{B}$ will be equal in each game. Therefore, the reconstruction simulated by A is identical to the one in game $\mathcal{G}_3$.

4.6 Breaking Secrecy

From equation (6) we get that if $r_{\text{Sharing}} \in \mathcal{W} \cap \mathcal{W}'$ then: $\mathbf{Pr}\left[s \text{ is reconstructed in game } \mathcal{G}_4\right] = 1$. And we've seen earlier that $|\mathcal{W} \cap \mathcal{W}'| \geq (1 - \varepsilon - \delta_s) \cdot |\mathcal{R}_s|$. Therefore, as far as A is concerned, the total probability that s is reconstructed in game $\mathcal{G}_4$ is $(1 - \varepsilon - \delta_s)$. Now, for negligible ε, δ_r is $(1 - \varepsilon - \delta_s)$ is clearly non-negligible. This is the probability that s is reconstructed by one corrupt player A during *sharing phase*. However, from *perfect* secrecy, we require that the probability that this happens for any set of up to t players is $= 1/|\mathbb{F}|$, which is negligible. Hence, we arrive at a contradiction. Therefore, we can conclude:

Theorem 4. *There exists no $(3, 1)$-SVSS scheme over 2 rounds, with secrets from a field $\mathbb{F}$, satisfying ε-correctness and (δ_s, δ_r)-commitment if $(1 - \varepsilon - \delta_s) \geq 1/|\mathbb{F}|$ and $\delta_r < \frac{1}{2}$.*

5 Impossibility of a 3-round (n, t)-SVSS for $n < 3t$

In this section, we show that any (n, t)-SVSS protocol requires at least 4 rounds for $n \leq 3t - 1$.

Theorem 5. *There exists no 3-round (n, t)-SVSS protocol, when $n \leq 3t - 1$.*

Proof Outline. We first describe a high level description of the working of a typical (n, t)-SVSS, for $n \leq 3t - 1$, and arrive at a break in security, when we allow only 3 rounds. We remark that the proof works only for $2t + 1 \leq n \leq 3t - 1$, for $t \geq 2$. This leaves open the question of round complexity of SVSS scheme when there are exactly $n = 3t$ players.

1. In the first round, the dealer $\mathcal{D}$ sends the "shares" to the players.
2. In the second round, the players "check" their shares mutually, using probabilistic checks, which will expose "inconsistencies" or "conflicts" in the shares with high probability.
3. The dealer $\mathcal{D}$ is asked to "resolve the conflicts" by broadcasting (a part of) the view of one of the players who caused the conflict.
4. At the end of the conflict resolution by the dealer, the view broadcast by him may not match some of the (honest) players. We call such a player "unhappy". Actually, a SVSS protocol needs an *additional* 4^{th} round, in which the unhappy players will broadcast their unhappiness with the dealer, and make it known to all the other players.

We show how a corrupt $\mathcal{D}$ may exploit the lack of a 4^{th} round (to expose unhappiness) to cause a break in commitment.

We consider an adversary, who corrupts the dealer during the sharing phase in the following way. The dealer holds some secret, but he deals out "nonsensical" shares to t players in set C in the first round. This is exposed at the end of the second round, and the dealer is required to broadcast conflict resolution information.

If the dealer is honest, this information may be relied upon. However, if the dealer is dishonest, it may happen that while the views of some players are consistent with the dealer broadcast (whom we assign to set A), some of the players (whom we assign to set B) are left unhappy with the broadcast meant to resolve conflicts. It turns out that the unhappy players may either be honest or corrupt themselves, therefore, unless we have a *strict majority* of honest players among the remaining players ($A \cup B$), it would be impossible to say whether the happy players or the unhappy players should be trusted. Since $A \cup B$ contains up to $t - 1$ dishonest players, we need at least t honest players in $A \cup B$, making $|A \cup B| \geq 2t - 1$. Therefore, the total number of players $n \geq t + 1 + (2t - 1) = 3t$.

The above proof idea does not work in the case of statistical WSS, since, a WSS protocol is allowed to fail in case we discover any unhappy players in the reconstruction phase (apart from the t players in C).

For the full proof, kindly refer to appendix D.

6 Communication Complexity and Connectivity Requirements of VSS

In this section, we establish a lower bound on the communication complexity of *perfectly-secure* WSS and VSS scheme and therefore in turn show that the 3-round WSS scheme in [11] and the 4-round VSS scheme in [12] are communication-optimal (although the 4-round scheme is not *round* optimal). In particular, we show a $\Omega\left(t^2/(n - 3t)\right)$ lower bound on the number of private messages communicated by any PVSS protocol, when $n < 4t$.

6.1 Connectivity Requirements

Private Communication Graph. For any execution E of any protocol, define the *(undirected) private communication graph* G on the set of players ($\mathcal{P}$) in the following way. For any pair of players P^1 and P^2, draw an edge between P^1 and P^2, if either P^1 communicated a message to P^2 through a private channel, or vice versa. (We can also define the more natural *directed* version of the communication graph, but we will use only the undirected graph in this paper.)

We use the concept of a communication graph to establish the lower-bound on the complexity of private communication in VSS schemes. We do this by studying the connectivity requirements (for the first time) of a general VSS scheme, and obtain the following independent result (Lemma 2). This result can then be used to establish Theorem 7:

Lemma 2. *Suppose Π is a VSS protocol, with dealer $\mathcal{D}$, on $n = 3t + 1$ players, tolerating up to t corrupt players. Let E be a corruption-free execution of the protocol, when the input to $\mathcal{D}$ is s. Suppose G is the private communication graph of E. Then, $G' = G \backslash \{\mathcal{D}\}$ must be $2t$-vertex connected.*

We defer this proof to Appendix F.

Theorem 6. *In any corruption-free execution E of a $(n = 3t + 1, t)$-VSS protocol, the non-dealers must communicate at least $3t^2$ private messages among themselves.*

Proof. Let G and G' be as in Lemma 2. Then, the vertex connectivity of G', and hence the minimum degree of G', is at least $\kappa' = 2t$. Thus, the number of edges in G' is at least $2t \cdot (n-1)/2 = 3t^2$. Since every edge of G' implies a private communication between the two vertices (in either direction), the number of private messages communicated between the players is at least $3t^2$.

We direct the reader to Appendix F.1 for a discussion on the lower bound of private messages.

For a general $n < 4t$, we get the following lower bound on the number of private messages communicated by any PVSS protocol.

Theorem 7. *Any (n, t)-VSS protocol, with $n < 4t$, must communicate at least $\Omega\left(t^2/(n-3t)\right)$ messages privately among the players.*

We refer the reader to Appendix F.2 for the proof of the above theorem.

6.2 Upper Bounds on Connectivity Requirements

In this section, we shall study the possibility of VSS in an incomplete network. We model the network in the following way. There are n players $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$. The players are equipped with a broadcast channel. Further, there exists a secure link between selected pairs of processors.

We specify the connectivity of the underlying network using the *undirected connectivity graph* G. The vertices of the graph are the n players in the protocol. We add an edge between two players if there is a secure channel between the two players. In general, it is possible that there are *one-way secure channels* between a pair of processors; in such a case, one of the processors is allowed to send private messages through the channel to another, but not in the opposite direction. We can capture such general networks using a *directed connectivity graph* G.

We first show that there is essentially no difference between one-way and two-way links: two-way private communication can be securely and efficiently simulated using one-way links, assuming a broadcast

channel (refer to Appendix E). Therefore, we assume that the underlying network only possesses two-way private channels between the players.

The problem of message transmission over incomplete graphs is a well-studied problem [9, 23]. We know that, over a given network, perfectly-secure transmission of a message m is possible between two (honest) nodes ($\mathcal{S}$,$\mathcal{R}$) iff the connectivity between the two nodes is at least $2t + 1$.

Amazingly, this result is tight for VSS scheme with optimal resilience. In Lemma 2, we established a lower bound of $2t$ on the vertex connectivity of G' for the existence of (n, t)-PVSS, for $n = 3t + 1$. Indeed, when the connectivity of G is at least $2t + 1$, then a (n, t)-VSS protocol can be securely simulated using a perfectly secure message transmission (PSMT) scheme. On the other hand, for the sake of completeness, and a better understanding of the requirements of the VSS protocols, we indicate a direct proof for the possibility of implementing a VSS in graphs with sufficient connectivity in Appendix G.

7 Conclusions

In this paper, we have explored for the first time the formal definition behind statistically-secure VSS schemes. In particular, we have shown four variants of commitment and also cast the traditional *perfectly-secure* commitment as one of the above. We have shown lower bounds assuming the weakest notion of commitment for (n, t)-SVSS schemes ($n \leq 3t$) and extended this result for the slightly sub-optimal resilience bound of $n \leq 3t - 1$. We have also considered the specialized case when the dealer doesn't broadcast and shown that in that case, irrespective of the number of rounds, VSS (and more generally, *weakened* Byzantine Agreement) is *impossible* for $n \leq 3t$.

Continuing on the theme of lower bounds, we have also explored the communication complexity, and established a quadratic lower bound for the same. We also show that for incomplete graphs, we require at least $2t$ connectivity. We have also shown this result to be tight by providing an explicit construction without message transmission (PSMT) and also noting that PSMT protocols also provide tight solutions.

Currently, we do have 2-round $(3, 1)$-SVSS and in general (n, t) for $n \leq 3t$ protocols in [19], which shows our results to be optimal. However, it still remains open to show a lower-bound on $(3, 1)$-SWSS protocol. We have also seen several variants of commitment, and a natural interesting question is whether or not we can show any separation among these variants.

In the direction of communication complexity, the main open questions are to characterize the number of messages communicated by a (n, t)-VSS. Also, we would like to establish possible trade-offs between the round and communication complexities of VSS schemes.

References

1. Judit Bar-Ilan and Donald Beaver. Non-cryptographic fault-tolerant computing in constant number of rounds of interaction. In *PODC*, pages 201–209, 1989.
2. Donald Beaver. Minimal-latency secure function evaluation. In *EUROCRYPT*, pages 335–350, 2000.
3. Donald Beaver, Joan Feigenbaum, Joe Kilian, and Phillip Rogaway. Locally random reductions: Improvements and applications. *J. Cryptology*, 10(1):17–36, 1997.
4. Donald Beaver, Silvio Micali, and Phillip Rogaway. The round complexity of secure protocols (extended abstract). In *STOC*, pages 503–513, 1990.
5. Zuzana Beerliova-Trubiniova and Martin Hirt. Efficient multi-party computation with dispute control. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography Conference — TCC 2006*, volume 3876 of *Lecture Notes in Computer Science*, pages 305–328. Springer-Verlag, March 2006.
6. Zuzana Beerliova-Trubiniova and Martin Hirt. Perfectly-secure MPC with linear communication complexity. In Ran Canetti, editor, *Theory of Cryptography Conference — TCC 2008*, volume 4948 of *Lecture Notes in Computer Science*, pages 213–230. Springer-Verlag, March 2008.

7. Christian Cachin, Jan Camenisch, Joe Kilian, and Joy Müller. One-round secure computation and secure autonomous mobile agents. In Ugo Montanari, José D. P. Rolim, and Emo Welzl, editors, *ICALP*, volume 1853 of *Lecture Notes in Computer Science*, pages 512–523. Springer, 2000.
8. Benny Chor, Shafi Goldwasser, Silvio Micali, and Baruch Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults (extended abstract). In *FOCS*, pages 383–395, 1985.
9. D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly secure message transmission. In *SFCS '90: Proceedings of the 31st Annual Symposium on Foundations of Computer Science*, pages 36–45 vol.1, Washington, DC, USA, 1990. IEEE Computer Society.
10. Uriel Feige, Joe Kilian, and Moni Naor. A minimal model for secure computation (extended abstract). In *STOC*, pages 554–563, 1994.
11. Matthias Fitzi, Juan A. Garay, Shyamnath Gollakota, C. Pandu Rangan, and K. Srinathan. Round-optimal and efficient verifiable secret sharing. In Shai Halevi and Tal Rabin, editors, *TCC*, volume 3876 of *Lecture Notes in Computer Science*, pages 329–342. Springer, 2006.
12. 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.
13. Martin Hirt, Ueli Maurer, and Bartosz Przydatek. Efficient secure multi-party computation. In Tatsuaki Okamoto, editor, *Advances in Cryptology — ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 143–161. Springer-Verlag, December 2000.
14. Martin Hirt and Jesper Buus Nielsen. Robust multiparty computation with linear communication complexity. In Cynthia Dwork, editor, *Advances in Cryptology — CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 463–482. Springer-Verlag, August 2006.
15. Yuval Ishai and Eyal Kushilevitz. Private simultaneous messages protocols with applications. In *ISTCS*, pages 174–184, 1997.
16. Yuval Ishai and Eyal Kushilevitz. Randomizing polynomials: A new representation with applications to round-efficient secure computation. In *FOCS*, pages 294–304, 2000.
17. Mitsuru Ito, Akira Saito, and Takao Nishizeki. Secret sharing scheme realizing general access structure. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, 72(9):56–64, 1989.
18. Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4:382–401, 1982.
19. Arpita Patra, Ashish Choudhary, Tal Rabin, and C. Pandu Rangan. The round complexity of verifiable secret sharing revisted. In *CRYPTO*, 2009.
20. B. Prabhu, K. Srinathan, and C. Pandu Rangan. Asynchronous unconditionally secure computation: An efficiency improvement. In Alfred Menezes and Palash Sarkar, editors, *INDOCRYPT*, volume 2551 of *Lecture Notes in Computer Science*, pages 93–107. Springer, 2002.
21. Tal Rabin and Michael Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority (extended abstract). In *STOC*, pages 73–85, 1989.
22. Tomas Sander, Adam Young, and Moti Yung. Non-interactive cryptocomputing for nc^1. In *FOCS*, pages 554–567, 1999.
23. K. Srinathan, A. Narayanan, and C.P. Rangan. Optimal perfectly secure message transmission. *Lecture notes in computer science*, pages 545–561, 2004.

A Impossibility of Byzantine Agreement

Theorem 8. *Statistically-secure weakened (n, t)-Byzantine Agreement is impossible for $n \leq 3t$.*

Proof. In the proof in section 3, note that we haven't explicitly used the *secrecy* property of SVSS. Therefore, this lower bound holds even for protocols Π that do not maintain secrecy. In this case, SVSS reduces to Byzantine Agreement, where the dealer must, in the absence of broadcast, communicate a secret s to the players. The above proof shows that (n, t)-Byzantine Agreement, even if the players are given additionally the power to broadcast information, cannot be solved for $n \leq 3t$. This completes the proof.

A.1 Adjusting parameters

In the above proof, we show that there is no $(3, 1)$-SVSS protocol, without dealer broadcast, and with ε-correctness, (δ_s, δ_r)-commitment for any $0 \leq \delta_s \leq 1 - 2\sqrt{\varepsilon}$ and $0 \leq \delta_r \leq \sqrt{\varepsilon}$. This is because when we

extract the probability for reconstruction, we look for a $\mathcal{W}$ of cardinality $(1-\sqrt{\varepsilon})\cdot|\mathcal{R}_s|$. Instead, if we look for a $\mathcal{W}$ of cardinality $(1-\varepsilon_1)\cdot|\mathcal{R}_s|$, then we can easily see that equation (3) should be modified to:

$$\mathop{\mathbf{Pr}}_{r_{\text{Reconstruction}}}[s \text{ is reconstructed}|r_{\text{Sharing}} \in \mathcal{W}] \geq 1 - \frac{\varepsilon}{\varepsilon_1}. \tag{7}$$

We see that immediately we can change the parameter constraints as follows. If there exists ε_1 and ε_2 ($\varepsilon_i \leq \frac{1}{2}$) such that $\varepsilon_1 \cdot \varepsilon_2 \leq \varepsilon$, $0 \leq \delta_s \leq 1-2\varepsilon_1$ and $0 \leq \delta_r \leq \varepsilon_2$, then there is no $(3,1)$-SVSS protocol (without broadcast) such that it is simultaneously satisfies ε-correctness and (δ_s, δ_r)-commitment. This generalizes the previous constraints (albeit with a lot more parameters).

B Proof of Lemma 1

Lemma 3. *For every protocol Π that is a (n,t)-SVSS scheme with parameters $(\varepsilon, \delta_s, \delta_r)$, we can create a new scheme Π' such that all players who follow the protocol honestly will not use any randomness during reconstruction. Also, Π' will also be a (n,t)-SVSS scheme with slightly weaker parameters $(\varepsilon, \delta_s + \delta_r - \delta_s\delta_r, \delta_r)$.*

Proof. In order to do this, we simply modify the protocol Π such that all players toss their random coins during the sharing phase itself, and therefore fix their views (to be given during reconstruction). It is easy to see that Π' and Π have the same round complexity and resilience. Also, correctness is not affected. However, we lose a factor of $\delta_r - \delta_s\delta_r$ in commitment (because our error probability can now be as much as $(1-\delta_s)\cdot(1-\delta_r)$). Since we are dealing with only the weakest form of commitment, this loss in commitment doesn't affect our results, because (δ_s, δ_r)-committed schemes are still (δ'_s, δ_r)-committed for (still) negligible parameters δ'_s and δ_r.

The advantage with this definition is that it allows us to simplify the earlier proof. This is because our adversaries do not (in any of the games) employ randomness *at all* in the reconstruction phase. Although they do use randomness (to choose $r'_{\mathcal{D}}$ say), this happens during the sharing phase itself. In fact, section 4, we will assume that the protocol Π is a $(\varepsilon, \delta_s, \delta_r)$ protocol (for negligible parameters) where all honest players are deterministic during reconstruction.

C A small note on relaxing secrecy requirements

In section 4, we have seen the following theorem:

Theorem 9. *There exists no $(3,1)$-SVSS scheme over 2 rounds, with secrets from a field $\mathbb{F}$, satisfying ε-correctness and (δ_s, δ_r)-commitment if $(1-\varepsilon-\delta_s) \geq 1/|\mathbb{F}|$ and $\delta_r < \frac{1}{2}$.*

So far, even in the definition of SVSS, we have not relaxed the notion of secrecy. We assumed that protocols satisfy 0-secrecy, because all protocols so far in literature have only considered 0-secrecy [19, 21]. However, we can relax this definition. We say that an adversary has an advantage ν if:

$$\mathop{\mathbf{Pr}}_{r_{\text{Adversary}}, r_{\text{HonestPlayers}}}[s \text{ is reconstructed}] > \frac{1}{|\mathbb{F}|} + \nu,$$

where $s \in \mathbb{F}$ is the secret shared in the execution of the protocol. Therefore, naturally, we can extend 0-secrecy, to ν-secrecy as follows:

Definition 2 (ν-secrecy) *A SVSS protocol satisfies ν-secrecy iff for every execution of the protocol where the dealer $\mathcal{D}$ is honest no adversary has an advantage (as defined above) $\geq \nu$.*

As we've seen from the previous theorem, for a 2-round $(3,1)$-SVSS protocol that satisfies ε-correctness, and (δ_s, δ_r)-commitment, we are able to reconstruct s with a probability $(1-\varepsilon-\delta_s)$. Therefore for all ν such that $\nu \leq 1-\varepsilon-\delta_s-1/|\mathbb{F}|$, a 2-round $(3,1)$-SVSS protocol satisfying ν-secrecy (as opposed to 0-secrecy) is impossible. In particular, if ν is negligible in the security parameter, then the above inequality is trivially satisfied (for negligible ε and δ_s). Therefore, relaxing secrecy, by any negligible amount is impossible.

D Proof of impossibility of 3-round (n,t)-SVSS for $n \leq 3t-1$

Our proof proceeds according to multiple games. We assume that the number of players $n \leq 3t-1$. For convenience, we partition the players $\mathcal{P} \setminus \{\mathcal{D}\}$ into 3 groups or sets A,B, and C, such that $|A| = |B| = t-1$, and $|C| = n-(t-1)-(t-1)-1 = n+1-2t \leq t$.

D.1 Game $\mathcal{H}_1$ and $\mathcal{H}'_1$

In the game $\mathcal{H}_1$, the players in C are corrupt, and behaves as if they does not get any share from the dealer $\mathcal{D}$; whenever they receive any message from $\mathcal{D}$, they replace it by a default $\mathbf{0}$ message. Similarly, when they communicate with $\mathcal{D}$, they simply send $\mathbf{0}$. But, when communicating with the other players, they follow the steps of the protocol. The behavior of these corrupt players is exactly as explained in section 4.2 .
Note: A typical VSS protocol detects that the shares of C are not consistent with the remaining players at the end of the 2^{nd} round, and requires the dealer to broadcast some conflict resolution information (probably the share given to C by $\mathcal{D}$) in the 3^{rd} round, which the dealer follows honestly.

The dealer holds the secret s. Since the dealer is honest, we get that the probability of reconstruction of the secret is at least $1-\varepsilon$, taken over the randomness of the dealer $\mathcal{D}$, and the players in A, B and C.

The game $\mathcal{H}'_1$ is identical to the previous game $\mathcal{H}_1$, except that the secret held by the dealer is $s' \neq s$.

D.2 Game $\mathcal{H}_2$ and $\mathcal{H}'_2$

In the game $\mathcal{H}_2$, the dealer is corrupt, and he sends "nonsensical" shares to the players in C in the first round of the sharing phase. However, he ensures that the execution of the protocol is identical to that of game $\mathcal{H}_1$. That is, the dealer replaces all private communication with C by $\mathbf{0}$. He follows the protocol while sending messages privately to the other players, or broadcasting any message. Therefore the probability of reconstructing s is at least $1-\varepsilon$.

The game $\mathcal{H}'_2$ is identical to $\mathcal{H}_2$, except that the secret held by $\mathcal{D}$ is s'. Therefore, this is identical to $\mathcal{H}'_1$. In this game, with probability at least $1-\varepsilon$, we must reconstruct s'.

D.3 Game $\mathcal{H}_3$

In this game, the dealer $\mathcal{D}$, as well as the players in B, are corrupt. The sharing phase of this game proceeds exactly as in $\mathcal{H}_2$. In particular, the secret initially held by $\mathcal{D}$ is s. For convenience, let $\mathcal{T}_A$ denote the set of transcripts held by all the players in A, and similarly for the sets B and C.

In the reconstruction phase, the players in B randomly choose a set of transcripts $\mathcal{T}^*_B$ and $\mathcal{T}^*_{\mathcal{D}}$ as if the secret is actually s', in such a way that the communication between $B \cup \{\mathcal{D}\}$ and C according to the fake transcripts $\mathcal{T}^*_B, \mathcal{T}^*_{\mathcal{D}}$ matches that according to $\mathcal{T}_B, \mathcal{T}_D$. To achieve this, the adversary exhaustively tries

executing the protocol with every choice of randomness for the dealer $\mathcal{D}$, as well the players in A and B, and picks at random one choice which ensures that the communication between $B \cup \{\mathcal{D}\}$ and C is exactly as in the sharing phase. Note that there might be multiple choices for the randomness, and the adversary could pick any of them at random.

Now, by a commitment argument, with a probability of at least $1-\varepsilon-\delta_s$, at the end of the sharing phase, the secret s is "committed". Therefore, the probability that we reconstruct s is at least $(1-\varepsilon-\delta_s)\cdot(1-\delta_r)$.

D.4 Game $\mathcal{H}'_3$

In this game, the dealer $\mathcal{D}$ and the players in A are corrupt. The sharing phase of this game proceeds identically as in game $\mathcal{H}'_2$. (The secret held by $\mathcal{D}$ is s'.)

Our goal is now to produce a set of transcripts that are identical (i.e., identically distributed as that in the game $\mathcal{H}_3$), and yet reconstruct s' with high probability. Towards this, in the reconstruction phase, the adversary now randomly picks a transcript $\mathcal{T}^*_A$, as if the secret is actually s, such that the communication between A and C are identical under the real transcript $\mathcal{T}_A$ and the fake one $\mathcal{T}^*_A$.

In this game, by a similar commitment argument, we reconstruct the secret s' with probability $(1-\varepsilon-\delta_s)\cdot(1-\delta_r)$.

It can be verified that the transcripts produced by all the players in the games $\mathcal{H}_3$ and $\mathcal{H}'_3$ are identically distributed. However, in the game $\mathcal{H}_3$, we reconstruct s with overwhelming probability $(1-\varepsilon-\delta_s)\cdot(1-\delta_r)$, while in the game $\mathcal{H}'_3$, we reconstruct s' with the same overwhelming probability, which cannot happen simultaneously.

E Upper Bounds for Connectivity

Lemma 4. *Assuming a broadcast channel, and a one-way link between a pair of players P^1 and P^2, one can securely and efficiently simulate message transmissions in both directions. In particular, we can achieve this with just one private message.*

Proof. Suppose P^1 and P^2 are two players with a link that allows messages from P^1 to P^2, but not in the opposite direction. Now, we are required to simulate a message transmission from P^2 to P^1. This is inspired by the idea of **random secret pads**. When P^2 must send a value m to P^1, first P^1 generates a uniformly random field element r, and transmits it to P^2, who broadcasts the "encrypted" value $c = m+r$. P^1 obtains the value m by "decrypting" c, using $m = c - r$.

The correctness of the above scheme is obvious. (When either player is dishonest, the adversary's strategy may also be simulated directly.) Further, the (perfect) privacy of the transmitted value m follows from the perfect security of the folklore *random padding technique.*

F Proof of Lemma 2

We will assume without any loss in generality that only the dealer has access to a random input r. We restate the lemma here.

Lemma 5. *Suppose Π is a VSS protocol, with dealer $\mathcal{D}$, on $n = 3t+1$ players, tolerating up to t corrupt players. Let E be a corruption-free execution of the protocol, with secret s, and randomness r given as input to $\mathcal{D}$. Suppose G is the private communication graph of E. Then, $G\backslash\{\mathcal{D}\}$ must be $2t$-vertex connected.*

Proof.

Suppose not. Then, there are $(2t-1)$ vertices in $G\backslash\{\mathcal{D}\}$, which when removed disconnects the graph. Partition these $(2t-1)$ vertices arbitrarily into two sets A and B, of sizes t and $(t-1)$ respectively. Let $D = B \cup \{\mathcal{D}\}$. The graph $G\backslash(A \cup D)$ is disconnected; let us assume it has two components C and C'. It is straightforward to see that $|A| + |D| = 2t$, so that $|C| + |C'| = n - 2t = t + 1$. In particular, we must have that $1 \leq |C| \leq t$, and $1 \leq |C'| \leq t$.

By the secrecy requirement, for any $s' \neq s$, there exists an execution E', with secret s' and randomness r', such that the views of A in E and E' are identical. Clearly, the communication graph of E' is also G, and the set of broadcasts done by the players in the executions E and E' are identical.

Now, consider the following executions of the protocol.

- **Execution** $\mathcal{H}_1$: In this execution, the dealer is honest, and the adversary corrupts C'. (Note that $|C'| \leq t$.) The dealer holds the secret s, and honestly deals it to all players, according to execution E.
  In the reconstruction phase, the players in C' provide a view corresponding to execution E'. However, since the dealer is honest, the secret s is **correctly** reconstructed.
- **Execution** $\mathcal{H}_2$: In this execution, the adversary corrupts the players in D. Intuitively, the dealer deals out the secret s to players in C, and secret s' to players in C'. Formally, the dealer gives out shares according to execution E to players in C, and according to E' to players in C'. The players in C and C' behave exactly as in the respective executions–E or E'. Since, the players in A have identical views in the executions E and E', they behave in an identical view in this *hybrid* execution as well. On the other hand, the corrupt players pretend to behave according to execution E, when communicating with C, and according to E' when communicating with C'. (When a corrupt player is required communicates privately with an A-player, the message to be sent is identical in the executions E and E', so the common message is sent.) Finally, we note that the collective broadcasts in the execution E is identical to that in E' (both in the origin of the broadcast and the message broadcast); therefore the behavior of any honest player does not deviate from the respective execution(s) (E or/and E').
  In the reconstruction phase, however, the dishonest parties broadcast their views according to E. It is easy to see that the views broadcast by all the players in this execution are the same as in the previous execution $\mathcal{H}_1$. Therefore, the reconstruction function reconstructs the same value, s, as before.
- **Execution** $\mathcal{H}'_1$: In this execution, the dealer is honest, and the adversary corrupts C. (Note that $|C| \leq t$.) The dealer holds the secret s', and honestly deals it to all players.
  In the reconstruction phase, the players in C provide a view that they were given shares corresponding to secret s. However, since the dealer is honest, the secret s' is correctly reconstructed.
- **Execution** $\mathcal{H}'_2$: The sharing phase of this execution is identical to the sharing phase of the execution $\mathcal{H}_2$. In the reconstruction phase, however, the dishonest players, including the dealer, publish their views according to E'. Again, the views broadcast by all the players in executions $\mathcal{H}'_1$ and $\mathcal{H}'_2$ match, so that the secret s' is reconstructed.

Now, we have a pair of executions, $\mathcal{H}_2$ and $\mathcal{H}'_2$, where the sharing phases are identical. However, depending on what views are provided by the dishonest players (D) in the reconstruction phase, the adversary may force the protocol to reconstruct either s or s'. This directly violates the commitment property of the VSS scheme. This implies that, contrary to our starting assumption, the execution graph (with the dealer removed) $G\backslash\{\mathcal{D}\}$ is indeed $(2t)$-connected.

The above lemma establishes a strong bound on connectivity of the communication graph of any execution of the protocol. This can be translated to give a (tight) bound on the number of messages that must be communicated privately between the players.

F.1 A discussion of the lower bound on private messages

We provide an intuition behind the lower bound argument provided in this section. The lower bound established in this section is derived by creating a scenario, where the views of all the honest guys are *locally consistent*, and yet, there is no *global consistency* over all players. In order to ensure that the local consistency for each player leads to commitment, we see that each player must perform consistency checks with sufficiently many players. Further, the secrecy requirement forces sufficiently many checks to be performed *privately*.

All natural protocols for VSS are designed such that every player gets a share of the secret, with sufficient redundancy. The redundancy is useful in making *consistency checks* among players; when such a check fails, leading to what is known as a *conflict*, the dealer is requested to broadcast some information, helpful in *resolving* the conflict. In developing a scenario with no global consistency, we must however take care that no local conflicts occur (between the honest players). This was crucial, for instance, in the last proof: C and C' cannot communicate with each other, and executions corresponding to the secrets s and s' are carefully chosen, so that they produce identical views in the middle set A. Further, in case we do create certain local conflicts in a protocol execution, we must carefully analyze the subsequent behavior of the protocol. As long as we are interested in communication complexity, we can assume that the dealer $\mathcal{D}$ is required to take stern steps in the event of a conflict: for instance, the dealer may detect the player who disagrees with him, and *expose his complete view* to all the other players by broadcast. Such conflict resolution mechanisms may, however, come at a cost. For instance, broadcasting the unhappy player's view might consume an extra round in the protocol. Care must be taken to broadcast as little information as possible while still retaining commitment.

Finally, when comparing the views of executions, it is advisable to choose only *corruption-free executions*. This is because an adversary may potentially misbehave in an arbitrary way, causing the protocol also to behave differently than before, in order to catch or cope up with the central adversary.

F.2 Proof for lower bound on private messages for $n < 4t$

In this section, we show a lower bound on the number of private messages communicated by a (n, t)-PVSS protocol, when $3t + 1 \leq n \leq 4t$.

Theorem 10. *Any (n, t)-VSS protocol, with $n < 4t$, must communicate at least $n \cdot t/(n - 3t)$ messages privately among the players.*

Proof. This proceeds along similar lines as the optimal resilience case (Theorem 6). Assume that there is a (n, t)-PVSS protocol, with $n < 4t$, that communicates less than $n \cdot t/(n - 3t)$ messages privately. Let G be the communication graph of some protocol execution E. Let C be a set of $n - 3t < t$ vertices in G such that the sum of degrees is minimum. The sum of degrees of all vertices is $< 2nt/(n - 3t)$, so the sum of degrees of vertices in C is less than $2nt/(n - 3t) \cdot (n - 3t)/n = 2t$. Therefore, the number of neighbors of C is also at most $2t$. Therefore, there exist $2t$ vertices that when removed will disconnect the graph into C and C' (say).

Let us partition the $2t$ vertices arbitrarily into A and B, each of size t. Now, by an argument identical to the optimal resilience case, this will lead to a break in the commitment property of PVSS. Thus, we get a $\Omega(nt/(n - 3t))$ bound on the number of private messages communicated by the protocol.

G Tightness of Lemma 2

In this appendix, we show that the Lemma 2 is tight. We *do not use* the primitive of Secure Message Transmission (SMT) to achieve it and instead show an alternative result. This provides better insight and understanding into *why* we require the connectivity $2t$ for VSS.

G.1 A connectivity-optimal protocol for *any* connectivity graph

Our solution is based on an inefficient 3-round $(3t+1, t)$-VSS protocol proposed by [12]. We first describe the original scheme, and analyze the security of the protocol.

The scheme is based on *replication based secret sharing* scheme given by [17]. In this protocol, we let $\binom{\mathcal{P}}{\ell}$ denote the set of all ℓ-subsets of $\mathcal{P}$. The dealer $\mathcal{D}$ divides the share additively into $\binom{n}{t}$ shares. That is, $\mathcal{D}$ chooses random s_A for every $A \in \binom{\mathcal{P}}{n-t}$, subject to the condition that $s = \sum_{A \in \binom{\mathcal{P}}{n-t}} s_A$. Then $\mathcal{D}$ hands down s_A to every player in A. Subsequently, every pair of players P_i and P_j exchange their common shares to check if they agree. In case of a disagreement, the pair of players broadcast a "complaint". The dealer resolves the complaint by broadcasting the controversial share. In the reconstruction phase, every share s_A is reconstructed by a majority vote.

As described above, the protocol would take 4 rounds, but one of the round can be collapsed by every pair of players exchanging independent random pads for every common share, and subsequently, broadcasting the common share, padded with the randomness.

Let us analyze the security of the protocol. The following observation is critical. At the end of the sharing phase, for every set A, all the honest players agree on s_A. Further, if the dealer is honest, this agreed value is the share s_A originally computed by the $\mathcal{D}$. This makes sure that a majority vote on the share s_A will reveal the correct output. Finally, we consider the secrecy property of the protocol. Suppose the dealer is honest. Since the adversary corrupts at most t players, there exists at least *one* t-subset A of players such that all the players in t are honest. Clearly, the adversary has no information regarding s_A, and hence, regarding s.

VSS on a given underlying network We now describe our modification to obtain a VSS protocol which uses only the private channels allowed by G. We assume that the network satisfies the following conditions.

1. All players have access to a broadcast channel.
2. $\mathcal{D}$ can communicate with all players.
3. $G' = G \backslash \{\mathcal{D}\}$ is $2t$-connected. That is, the subgraph of G' induced by any $t+1$-subset of $\mathcal{P} \backslash \mathcal{D}$ is connected.

The protocol is similar to the one just described. However, since there is no private channel between every pair of players, we restrict ourselves to just the checks that are directly allowed by the underlying network. Formally, two players P_i and P_j exchange their common shares, only if the edge (P_i, P_j) is in the graph G.

It remains to be shown that this is indeed a VSS protocol, when the graph G, with the dealer removed, still has sufficient connectivity. The correctness and secrecy properties are satisfied by the protocol in a straightforward way, akin to the original protocol. Let us now show that commitment is also satisfied. Assume that the dealer is corrupt. We only need to show that at the end of the sharing phase, all the honest players in A agree on s_A. Clearly, if there was any conflict regarding the share s_A between any pair of players, then the dealer broadcasts the value, thus the all the honest players agree on the broadcasted value trivially. Let us now consider the case that there was no conflict regarding s_A. In other words, all pairs of players agreed with each other about s_A; in particular, so did the honest players.

Consider the set A_{hon} of all the honest players in A. Clearly, $\mathcal{D}$ is not in A_{hon}. Further, $|A_{hon}| \geq |A| - t \geq n - 2t \geq t + 1$. Therefore, the subgraph of G induced by A_{hon} is connected. So, between any two players, there exists a path in G, consisting of players in A_{hon}, and there is agreement between every pair of adjacent players. This implies that any two players in A_{hon} are in agreement with each other.

Thus, we have shown that the modified protocol remains a VSS , as long as the underlying network has sufficiently high connectivity.