Improving the Round Complexity of VSS in Point-to-Point Networks

Jonathan Katz[*][†] Chiu-Yuen Koo[‡] Ranjit Kumaresan[*]

Abstract

We revisit the following question: *what is the optimal round complexity of verifiable secret sharing (VSS)*? We focus here on the case of perfectly-secure VSS where the number of corrupted parties t satisfies $t < n/3$, with n being the total number of parties. Work of Gennaro et al. (STOC 2001) and Fitzi et al. (TCC 2006) shows that, assuming a broadcast channel, 3 rounds are necessary and sufficient for efficient VSS. The efficient 3-round protocol of Fitzi et al., however, treats the broadcast channel as being available "for free" and does not attempt to minimize its usage. This approach leads to relatively poor round complexity when protocols are compiled for a point-to-point network.

We show here a VSS protocol that is *simultaneously* optimal in terms of both the number of rounds and the number of invocations of broadcast. Our protocol also has a certain "2-level sharing" property that makes it useful for constructing protocols for general secure computation.

1 Introduction

The round complexity of cryptographic protocols has been the subject of intense study. Besides protocols for general secure computation, protocols for various specific functionalities of interest (e.g., broadcast, zero-knowledge proofs, etc.) have also been explored. Here, we revisit the case of *verifiable secret sharing*, whose definition we now recall informally. (Formal definitions appear in Section 2.) In secret sharing [2, 19], there is a *dealer* who shares a secret among a group of n parties in a *sharing phase*. The requirements are that, for some parameter $t < n$, any set of t colluding parties gets no information about the dealer's secret at the end of the sharing phase, yet any set of $t+1$ parties can recover the dealer's secret in a later *reconstruction phase*. Secret sharing assumes the dealer is honest; *verifiable* secret sharing (VSS) [3] also requires that, no matter what a cheating dealer does (in conjunction with $t-1$ other colluding parties), there is *some* unique secret to which the dealer is "committed" by the end of the sharing phase. VSS serves as a fundamental building block in the design of protocols for general secure multi-party computation as well as other specialized goals (such as Byzantine agreement); thus, it is of interest to understand the inherent round complexity for carrying out this task.

In this work we will always consider perfectly-secure VSS, where the protocol is required to be error-free and security should hold even against an all-powerful adversary. This is known to be possible if and only if $t < n/3$ [1, 6]. Previous research investigating the round complexity of VSS, surveyed further below, has focused on optimizing the round complexity *assuming a broadcast*

[*] Dept. of Computer Science, University of Maryland. Email: {jkatz,ranjit}@cs.umd.edu.

[†] Research supported in part by NSF awards #0310751 and #0447075 (CAREER), and US-Israel Binational Science Foundation grant #2004240.

[‡] Dept. of Computer Science, University of Maryland and Google Labs. Email: cykoo@cs.umd.edu.

channel is available "for free". (We remark that broadcast is essential for VSS, in a way we make precise below.) As argued previously [13], however, if the ultimate goal is to optimize the round complexity of protocols for point-to-point networks (where protocols are likely to be run), then it is preferable to minimize the *number of rounds in which broadcast is used* rather than to minimize the *total number of rounds*. This is due to the high overhead of emulating a broadcast channel over a point-to-point network: deterministic broadcast protocols require $\Omega(t)$ rounds [8]; known randomized protocols [7, 9, 12] require only $O(1)$ rounds in expectation, but the constant is rather high. (The most round-efficient protocol known [12, 13] requires 23 rounds in expectation for $t < n/3$.[1]) Moreover, when using randomized broadcast protocols, if *more than one* invocation of broadcast is used then special care must be taken to deal with sequential composition of protocols without simultaneous termination (see [15, 12, 13]), leading to a substantial increase in the round complexity. As a consequence, a constant-round protocol that only uses a *single* round of broadcast is likely to yield a more round-efficient protocol in a point-to-point setting than any protocol that uses *two* rounds of broadcast (even if that protocol uses no additional rounds).

As a concrete example (taken from [13]) to illustrate the point, consider the VSS protocol of Micali and Rabin [16] and the 'round-optimal' VSS protocol of Fitzi et al. [10]. The former uses 16 rounds but only a single round of broadcast; the latter uses 3 rounds, two of which require broadcast. Compiling these protocols for a point-to-point network using the most round-efficient techniques known (see [13]), the Micali-Rabin protocol runs in an expected 31 rounds while the protocol of Fitzi et al. requires an expected 55 rounds!

In light of the above, when discussing the round complexity of protocols that assume a broadcast channel we keep track of both the number of rounds as well as the number of rounds in which broadcast is used. (In a given round when broadcast is used, each party may use the broadcast channel but a rushing adversary is still assumed. Existing broadcast protocols can be modified so that the round complexity is unchanged even if many parties broadcast in parallel.) We say a protocol has round complexity (r, r') if it uses r rounds in total, and $r' \leq r$ of these rounds invoke broadcast. The round complexity of VSS refers to the sharing phase only, since the reconstruction phase of most known protocols utilizes only a single round, without broadcast. (An exception is the protocol of [10], whose reconstruction phase uses a single round of broadcast.)

Our results and techniques. Gennaro et al. [11] show that three rounds are necessary for VSS, assuming a broadcast channel. We also observe that it is impossible to construct a *strict* constant-round protocol for VSS without using a broadcast channel in at least one round: VSS implies broadcast using one additional round (the message to be broadcast can be treated as the input for VSS), and the results of Fischer and Lynch [8] rule out strict constant-round protocols for broadcast. Prior work [16, 10, 13, 14] shows that optimal round complexity as well as optimal use of the broadcast channel could each be obtained *individually* for VSS, but it was unknown whether they could be obtained *simultaneously*. Here, we resolve this question and show a $(3, 1)$-round VSS protocol that is optimal in both measures. (Our protocol has a 1-round reconstruction phase that does not use broadcast.) As a consequence, we obtain a VSS protocol with the best known round complexity in point-to-point networks. Our work also leads to an improvement in the round complexity of the most round-efficient broadcast protocols known [12].

A nice feature of our VSS protocol is that it also satisfies a certain "2-level sharing" property that is not achieved by the 3-round protocol from [10]. Roughly speaking, this means that the following conditions hold at the end of the sharing phase when the dealer's (effective) input is s:

1. There exists a polynomial $f(x)$ of degree at most t such that $f(0) = s$ and each honest party

[1] Actually, the VSS protocol given here can be used to improve this slightly.

P_i holds the value $f(i)$.

2. For each party P_i, there exists a polynomial $f_i(x)$ of degree at most t such that $f_i(0) = f(i)$ and each honest party P_j holds the value $f_i(j)$.

VSS protocols with this property constitute a useful building block for protocols for general secure multi-party computation (see, e.g., [13, 14]).

Our protocol is efficient, in that the computation and communication are polynomial in n. The communication complexity of our protocol is $\mathcal{O}(n^2t)$ field elements, which matches the communication complexity of [10] but is worse than that of [11].

We now summarize the basic techniques used to prove our main result. As in [10], we begin by constructing a protocol for *weak* verifiable secret sharing (WSS) [18]. (In WSS, informally, if the dealer is dishonest then, in the reconstruction phase, each honest party recovers either the dealer's input or a special failure symbol.) Fitzi et al. show a $(3, 2)$-round WSS protocol that essentially consists of the first three rounds of the 4-round VSS protocol from [11]. On a high level, their protocol works as follows: In the first round, the dealer distributes the shares of the secret using a random bivariate polynomial; in parallel, each pair of parties (P_i, P_j) exchanges a random pad $r_{i,j}$. In the second round, P_i and P_j check for an inconsistency between their shares by broadcasting their common shares masked with the random pad. In the third round, if there is a disagreement between P_i and P_j in round 2 (note that all parties agree whether there is disagreement since broadcast is used in round 2), then the dealer, P_i, and P_j all broadcast the share in question. This allows the rest of the parties to determine whether the dealer "agrees" with P_i or with P_j.

A $(5, 1)$-round WSS protocol is implicitly given in [13].[2] There, rather than using the "random pad" technique, a different method is used to detect disagreement between P_i and P_j. While this saves one round of broadcast, it requires additional rounds of interaction.

To construct a $(3, 1)$-round WSS protocol, we modify the $(3, 2)$-round WSS protocol from [10] by using the random pad idea with the following twist: in the second round of the protocol, P_i and P_j check if there is any inconsistency between their shares by exchanging their common shares over a *point-to-point* link; they also send the random pad $r_{i,j}$ to the dealer. In the third round of the protocol, if there is a disagreement between P_i and P_j, then P_i and P_j each broadcast the shares they hold; otherwise, they broadcast the value of their common share masked with the random pad. The dealer will broadcast the corresponding share masked with the random pad (or the share itself if the random pads it received from P_i and P_j are different). Notice that secrecy of the share is preserved if P_i, P_j, and the dealer are all honest. On the other hand, if the dealer is malicious and there is a disagreement between honest parties P_i and P_j, then the dealer can only "agree" with at most one of P_i and P_j in round 3, but not both of them.

The above is the high-level idea of our WSS protocol. Using the same techniques as in [10], we can then immediately obtain a $(3, 1)$-round VSS protocol. However, the VSS protocol constructed in this manner will not have the "2-level sharing" property; as a consequence, the resulting protocol cannot directly be plugged in to existing protocols for general secure multi-party computation.

To convert the VSS protocol into one with 2-level sharing we note that, by the end of the sharing phase, there is a set of honest parties (that we call a "core set") who already *do* have the required 2-level shares; thus, we only need to provide honest parties outside the core set with their required shares. We achieve this, as in [5], by having the dealer use a *symmetric* bivariate polynomial to share its input, and then modifying the protocol so that honest parties who are not in the core set can still generate appropriate shares by interpolating the shares of the parties in the core set. Of

[2] That work shows a 6-round VSS protocol that uses broadcast in the final two rounds. The first five rounds of that protocol suffice for WSS.

course, this process needs to be carefully designed so that no additional information is leaked to the adversary. We defer the details of this to a later section.

Other related work. Gennaro et al. [11] initiated a study of the exact round complexity of VSS. For $t < n/3$, they show an efficient (i.e., polynomial-time) $(4, 3)$-round protocol, and an inefficient $(3, 2)$-round protocol. (Recall that the round complexity of VSS is defined as the number of rounds in the sharing phase; unless otherwise stated, all protocols mentioned use only one round, without broadcast, in the reconstruction phase.) They also show that three rounds are necessary for VSS when $t < n/3$. For $t < n/4$, they show that two rounds are necessary and sufficient for efficient VSS. Settling the question of the absolute round complexity of efficient VSS for $t < n/3$, Fitzi et al. [10] show an efficient $(3, 2)$-round VSS protocol. The reconstruction phase of their protocol requires one round of broadcast as well.

As discussed extensively already, although the protocol by Fitzi et al. is optimal in terms of the total number of rounds, it is not optimal in terms of its usage of the broadcast channel. VSS protocols for $t < n/3$ using one round of broadcast are known, but these protocols are not optimal in terms of their overall round complexity. Micali and Rabin [16] give a $(16, 1)$-round VSS protocol, and recent work of the authors [13, 14] improves this to give a $(7, 1)$-round protocol.

Our work, as well as all the work referenced above, focuses on VSS protocols with perfect security (i.e., *0-error VSS*). A natural relaxation is to consider *statistical* VSS where the security properties may fail with negligible probability. Surprisingly, recent work subsequent to our own [17] shows that the lower bound of Gennaro et al. [11] no longer holds in this setting, and that 2-round protocols are in fact possible.

Future directions. It would, of course, be nice to characterize the optimal round complexity of VSS in point-to-point networks. Though our work represents progress toward this goal, the question is complicated by the fact that one must consider the *distribution* of running times of any protocol (since strict constant-round protocols are ruled out). It will also be interesting to understand the round complexity of statistical VSS when $t < n/2$; see [17] for work in this direction.

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, with the understanding that it can be emulated in a point-to-point network using a broadcast protocol. 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. We stress that we do not assume *simultaneous* broadcast, but allow rushing here as well.

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 sent by the honest parties before deciding on their own messages.

2.1 VSS and Variants

We now present definitions of WSS, VSS, and VSS with 2-level sharing.

Definition 1 [Weak verifiable secret sharing] 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, is a WSS 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 Each honest party P_i outputs a value s_i at the end of the second phase (the reconstruction phase). If the dealer is honest then $s_i = s$.

Weak commitment At the end of the sharing phase the joint view of the honest parties defines a value s' (which can be computed in polynomial time from this view) such that each honest party will output either s' or a default value $\perp$ at the end of the reconstruction phase. $\diamondsuit$

Definition 2 [Verifiable secret sharing] A two-phase protocol for parties $\mathcal{P}$, where a distinguished dealer $D \in \mathcal{P}$ holds initial input s, is a VSS protocol tolerating t malicious parties if it satisfies the privacy and correctness requirements of WSS as well as the following (stronger) commitment requirement:

Commitment At the end of the sharing phase the joint view of the honest parties defines a value s' (which can be computed in polynomial time from this view) such that all honest parties will output s' at the end of the reconstruction phase. $\diamondsuit$

Definition 3 [Verifiable secret sharing with 2-level sharing] 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, is a VSS protocol with 2-level sharing tolerating t malicious parties if it satisfies the privacy and correctness requirements of VSS as well as the following requirement:

Commitment with 2-level sharing At the end of the sharing phase each honest party P_i outputs s_i and $s_{i,j}$ for $j \in \{1, \ldots, n\}$, satisfying the following requirements:

1. There exists a polynomial $p(x)$ of degree at most t such that $s_i = p(i)$ for every honest party P_i, and furthermore all honest parties will output $s' = p(0)$ at the end of the reconstruction phase.
2. For each $j \in \{1, \ldots, n\}$, there exists a polynomial $p_j(x)$ of degree at most t such that (1) $p_j(0) = p(j)$ and (2) $s_{i,j} = p_j(i)$ for every honest party P_i. $\diamondsuit$

This implies the commitment property of VSS, since the value $s' = p(0)$ that will be output in the reconstruction phase is defined by the view of the honest parties at the end of the sharing phase.

In our protocol descriptions, we implicitly assume all parties send properly-formatted messages at all times; this is without loss of generality, as we may interpret an improper or missing message as some default message. We assume the dealer's input s lies in a finite field $\mathbb{F}$ containing $\{0, 1, \ldots, n\}$ as a subset.

3 Weak Verifiable Secret Sharing

We show a $(3, 1)$-round WSS protocol tolerating $t < n/3$ malicious parties.

3.1 The Protocol

Sharing phase. The sharing phase consists of three rounds, with broadcast used in the last round.

Round 1: The dealer holds s. The following steps are carried out in parallel:

- The dealer chooses a random bivariate polynomial $F(x, y)$ of degree at most t in each variable such that $F(0, 0) = s$. The dealer then sends to each party P_i the polynomials $f_i(x) := F(x, i)$ and $g_i(y) := F(i, y)$.
- Each party P_i picks a random pad $r_{i,j} \in \mathbb{F}$ for $j \in \{1, \ldots, n\}$, and sends $r_{i,j}$ to both P_j and the dealer D.

Round 2: For every ordered pair (i, j), parties P_i and P_j proceed as follows:

- Party P_i sends $a_{i,j} := f_i(j)$ to P_j.
- Party P_j sends $b_{j,i} := g_j(i)$ to P_i.
(Note that, when everyone is honest, then $a_{i,j} = b_{j,i} = F(j, i)$.)
- Let $r'_{i,j}$ be the random pad that P_j received from P_i in the previous round. Then P_j sends $r'_{i,j}$ to D.

Round 3: For every ordered pair (i, j), parties P_i, P_j, and D do:

- (From the viewpoint of P_i:) If $b_{j,i} \neq f_i(j)$, then P_i broadcasts ("disagree", $f_i(j)$, $r_{i,j}$). Otherwise, P_i broadcasts ("agree", $f_i(j) + r_{i,j}$).
- (From the viewpoint of P_j:) If $a_{i,j} \neq g_j(i)$, then P_j broadcasts ("disagree", $g_j(i)$, $r'_{i,j}$). Otherwise, P_j broadcasts ("agree", $g_j(i) + r'_{i,j}$).
- (From the viewpoint of D:) If $r_{i,j} \neq r'_{i,j}$, then D broadcasts ("not equal", $F(j, i)$). Otherwise, D broadcasts ("equal", $F(j, i) + r_{i,j}$).

Local computation. An ordered pair of parties (P_i, P_j) is *conflicting* if, in round 3, party P_i broadcasts ("disagree", $f_i(j)$, $r_{i,j}$); party P_j broadcasts ("disagree", $g_j(i)$, $r'_{i,j}$); and $r_{i,j} = r'_{i,j}$. For a pair of conflicting parties (P_i, P_j), we say that P_i (resp., P_j) is *unhappy* if one of the following conditions hold:

- The dealer broadcasts ("not equal", $d_{i,j}$) and $d_{i,j} \neq f_i(j)$ (resp., $d_{i,j} \neq g_j(i)$).
- The dealer broadcasts ("equal", $d_{i,j}$) and $d_{i,j} \neq f_i(j) + r_{i,j}$ (resp., $d_{i,j} \neq g_j(i) + r'_{i,j}$).

Note that all parties agree on who is unhappy. If there are more than t unhappy parties, the dealer is disqualified and a default value is shared.

Reconstruction phase. The reconstruction phase is similar to the one in [10], except that we do not use broadcast.

1. Every party P_i that is not unhappy sends $f_i(x)$ and $g_i(y)$ to all other parties.
2. Let f^i_j, g^i_j denote the polynomials that P_j sent to P_i in the previous step. P_i then constructs a *consistency graph* G_i whose vertices correspond to the parties who are not unhappy:
    - Initially, there is an edge between P_j and P_k in G_i if and only if $f^i_j(k) = g^i_k(j)$ and $g^i_j(k) = f^i_k(j)$. (Note that we allow also the case $j = k$ here.)

- If there exists a vertex in G_i whose degree is less than $n - t$ (including self-loops), then that vertex is removed from G_i. This is repeated until no more vertices can be removed.

Let Core_i denote the parties whose corresponding vertices remain in G_i.

3. If $|\mathsf{Core}_i| < n - t$, then P_i outputs $\perp$. Otherwise, P_i reconstructs the polynomial $F'(x, y)$ defined by any $t + 1$ parties in Core_i, and outputs $s' := F'(0, 0)$.

We remark that, since we do not use broadcast in the reconstruction phase, it is possible that $\mathsf{Core}_i, \mathsf{Core}_j$ are different for different honest parties P_i, P_j.

3.2 Proofs

Lemma 1 *If the dealer is not corrupted by the end of the sharing phase, then privacy is preserved.*

Proof Let $\mathcal{C}$ denote the set of parties corrupted by the end of the sharing phase. We show that if the dealer remains uncorrupted, then the information the adversary has about the dealer's input at the end of the sharing phase consists of the polynomials $\{f_i(x), g_i(y)\}_{P_i \in \mathcal{C}}$. Since $F(x, y)$ is a random bivariate polynomial of degree at most t and $|\mathcal{C}| \le t$, a standard argument implies that the view of the adversary is independent of the dealer's input s.

It is immediate that the adversary learns nothing additional about s in round 2. As for the values broadcast in round 3, consider any ordered pair (P_i, P_j) of parties who remain honest throughout the sharing phase. Since the dealer is honest, we have $f_i(j) = g_j(i) = F(j, i)$ and, since P_i, P_j are honest, we have $r_{i,j} = r'_{i,j}$. Thus, in round 3, parties P_i, P_j, and the dealer all broadcast the same "blinded" value $F(j, i) + r_{i,j}$. Since $r_{i,j}$ is chosen uniformly at random from the point of view of the malicious parties, they do not learn anything about the value of $F(j, i)$. ■

Lemma 2 *If the dealer is not corrupted by the end of the sharing phase, then correctness holds.*

Proof Observe that if the dealer remains honest then no honest party will be unhappy. It follows that the dealer is not disqualified at the end of sharing phase.

Let P_i be honest. In the reconstruction phase, Core_i contains all the honest parties and so $|\mathsf{Core}_i| \ge n - t$. We claim that for any $P_j \in \mathsf{Core}_i$, it holds that $f^i_j(x) = F(x, j)$ and $g^i_j(y) = F(j, y)$, where F is the dealer's polynomial. When P_j is honest this is immediate. When P_j is malicious, the fact that $P_j \in \mathsf{Core}_i$ means that $f^i_j(k) = g^i_k(j) = F(k, j)$ for at least $n - 2t \ge t + 1$ honest parties P_k. Since $f^i_j(x)$ has degree at most t, it follows that $f^i_j(x) = F(x, j)$. A similar argument shows that $g^i_j(y) = F(j, y)$. Therefore, the polynomial $F'(x, y)$ reconstructed by P_i is equal to $F(x, y)$, and P_i outputs $s = F(0, 0)$. ■

Lemma 3 *Weak commitment holds.*

Proof The case of an honest dealer follows from the proof of correctness, so we consider the case of a malicious dealer. If there are more than t unhappy parties, the dealer is disqualified and weak commitment trivially holds; so, assume there are at most t unhappy parties. Then there are at least $n - 2t \ge t + 1$ honest parties who are not unhappy. Let $\mathcal{H}$ denote the first $t + 1$ such parties. The polynomials f_i sent by the dealer to the parties in $\mathcal{H}$ define a bivariate polynomial $\hat{F}(x, y)$ in the natural way: namely, let $\hat{F}$ be such that $\hat{F}(x, i) = f_i(x)$ for each $P_i \in \mathcal{H}$. Because parties in $\mathcal{H}$ are not unhappy, it holds also that $\hat{F}(i, y) = g_i(y)$ for all $P_i \in \mathcal{H}$. Set $s' := \hat{F}(0, 0)$. We show that every honest party outputs either $\perp$ or s' in the reconstruction phase.

Consider an honest party P_i in the reconstruction phase. If $|\mathsf{Core}_i| < n-t$ then P_i outputs $\perp$ and we are done. Say $|\mathsf{Core}_i| \geq n-t$. We claim that for each $P_j \in \mathsf{Core}_i$, it holds that $f^i_j(x) = \hat{F}(x,j)$ and $g^i_j(y) = \hat{F}(j,y)$. When P_j is honest, the fact that P_j is not unhappy (which is true since $P_j \in \mathsf{Core}_i$) means that $f^i_j(k) = f_j(k) = g_k(j) = \hat{F}(k,j)$ for all $t+1$ parties $P_k \in \mathcal{H}$. Since f^i_j is a polynomial of degree at most t, this implies that $f^i_j(x) = \hat{F}(x,j)$. A similar argument shows that $g^i_j(y) = \hat{F}(j,y)$. When $P_j \in \mathsf{Core}_i$ is malicious, we have that $f^i_j(k) = g^i_k(j) = \hat{F}(k,j)$ for at least $n-2t \geq t+1$ honest parties $P_k \in \mathsf{Core}_i$. Again, since $f^i_j(x)$ has degree at most t it follows that $f^i_j(x) = \hat{F}(x,j)$, and a similar argument shows that $g^i_j(y) = \hat{F}(j,y)$. Therefore, the polynomial reconstructed by P_i is equal to $\hat{F}(x,y)$, and P_i outputs $s' = \hat{F}(0,0)$. ■

As the proof of the above lemma indicates, our WSS protocol also satisfies a weak variant of 2-level sharing that we state for future reference:

Lemma 4 *Say the dealer is not disqualified in an execution of the WSS protocol, and let $\mathcal{H}$ denote the set of all honest parties who are not unhappy. Then there is a bivariate polynomial $\hat{F}$ of degree at most t in each variable such that, at the end of the sharing phase, the polynomials f_i, g_i held by each $P_i \in \mathcal{H}$ satisfy $f_i(x) = \hat{F}(x,i)$ and $g_i(y) = \hat{F}(i,y)$.*

As a consequence, each $P_i \in \mathcal{H}$ can compute s_i and $s_{i,j}$ for $j \in \{1,\ldots,n\}$ such that:

1. *There is a polynomial $p(x)$ of degree at most t with $s_i = p(i)$, and furthermore all honest parties output either $s' = p(0)$ or $\perp$ in the reconstruction phase.*

2. *For each $j \in \{1,\ldots,n\}$, there exists a polynomial $p_j(x)$ of degree at most t such that (1) $p_j(0) = p(j)$ and (2) $s_{i,j} = p_j(i)$.*

Proof When the dealer is honest take $\hat{F}$ to be the dealer's polynomial. When the dealer is dishonest, let $\hat{F}$ be the bivariate polynomial defined in the proof of the preceding lemma. Set $p(x) \stackrel{\text{def}}{=} \hat{F}(0,x)$ and $p_j(x) \stackrel{\text{def}}{=} \hat{F}(x,j)$. In what follows we assume a dishonest dealer, but it is immediate that everything (trivially) holds also if the dealer is honest.

The proof of the preceding lemma shows that, at the end of the sharing phase, each $P_i \in \mathcal{H}$ holds polynomials f_i, g_i with $f_i(x) = \hat{F}(x,i)$ and $g_i(y) = \hat{F}(i,y)$, and such that all honest parties output either $s' = \hat{F}(0,0)$ or $\perp$ in the reconstruction phase. Then each $P_i \in \mathcal{H}$ can compute $s_i := f_i(0) = \hat{F}(0,i) = p(i)$ and $s_{i,j} := g_i(j) = \hat{F}(i,j) = p_j(i)$. Furthermore, $s' = p(0)$. Finally, $p_j(0) = \hat{F}(0,j) = p(j)$ for all $j \in \{1,\ldots,n\}$. Thus, all the stated requirements hold. ■

4 Verifiable Secret Sharing

Before we describe our VSS protocol with 2-level sharing, we review the ideas used in [10] to transform their WSS protocol into a VSS protocol (that does not have 2-level sharing). At a high level, the sharing phase of the VSS protocol is more-or-less the same as the sharing phase of the underlying WSS protocol; the difference is that now, in the reconstruction phase, each party reveals the random pads they used in the sharing phase. A problem that arises is to ensure that a malicious party P_i reveals the "correct" random pads. This is enforced by having each player act as a dealer in its own execution of WSS, and "binding" the random pads of each party to this execution of WSS. In more detail: in parallel with the sharing phase of the larger VSS protocol, each party P_i also acts as a dealer and shares a random secret using the WSS protocol. Let $F^{pad}_i(x,y)$ be the corresponding bivariate polynomial chosen by P_i. Then P_i will use $r_{i,j} := F^{pad}_i(0,j)$ as the appropriate "random pad" in the larger VSS protocol. (The pads $\{r_{i,j}\}$ used by any honest party P_i are thus no longer independent, but secrecy is still preserved since they lie on a random degree-t polynomial.) These random pads are then revealed in the reconstruction phase by using the reconstruction phase of the underlying WSS protocol.

We can use the ideas outlined in the previous paragraph to obtain a $(3, 1)$-round VSS protocol, but the resulting protocol will not have 2-level sharing. Yet all is not lost. As observed already in Lemma 4, by the end of the sharing phase of the resulting VSS protocol the honest parties that are *not* unhappy *do* have the required 2-level shares. To achieve our desired result we must therefore only enable any *unhappy* honest party to construct *its* 2-level shares.

At a high level, we do this as follows: Suppose $\hat{F}(x, y)$ is the dealer's bivariate polynomial, defined by the end of the sharing phase of the VSS protocol, and let P_i be an honest party who is unhappy. We need to show how P_i constructs the polynomials $\hat{F}(x, i)$ and $\hat{F}(i, y)$ (which it will use to generate its 2-level shares exactly as in the proof of Lemma 4). Let P_j be a party such that:

- P_j is not unhappy (in the larger VSS protocol);
- P_j was not disqualified as a dealer it its own execution of WSS; and
- P_i is not unhappy in P_j's execution of WSS.

From the proof of Lemma 4, we know there is a bivariate polynomial $\hat{F}_j^{pad}(x, y)$ for which P_i holds the univariate polynomial $\hat{F}_j^{pad}(x, i)$. Furthermore, P_j has effectively broadcasted the polynomial $B_j(x) \overset{\text{def}}{=} \hat{F}(x, j) + \hat{F}_j^{pad}(0, x)$ in round 3, since it has broadcasted $\hat{F}(k, j) + \hat{F}_j^{pad}(0, k)$ for all k. Thus, party P_i can compute

$$\hat{F}(i, j) := B_j(i) - \hat{F}_j^{pad}(0, i) = \hat{F}(i, j)$$

for any party P_j satisfying the above conditions. If there are $t + 1$ parties satisfying the above conditions, then P_i can reconstruct the polynomial $\hat{F}(i, y)$.

Unfortunately, it is not clear how to extend the above approach to enable P_i to also reconstruct the polynomial $\hat{F}(x, i)$ in the case when $\hat{F}$ is an *arbitrary* bivariate polynomial. For this reason, we have the dealer use a *symmetric*[3] bivariate polynomial. Then $\hat{F}(x, i) = \hat{F}(i, x)$ and we are done.

4.1 The Protocol

We show a $(3, 1)$-round VSS protocol with 2-level sharing that tolerates $t < n/3$ malicious parties. Proofs of security are deferred to the appendix.

Sharing phase. The sharing phase consists of three rounds, with broadcast used in the last round.

Round 1: The dealer holds s. The following steps are carried out in parallel:

1. The dealer chooses a random *symmetric* bivariate polynomial $F(x, y)$ of degree t in each variable such that $F(0, 0) = s$. Then D sends to each party P_i the polynomial $f_i(x) := F(x, i)$. Note that $F(x, i) = F(i, x)$ since F is symmetric.
2. Each party P_i picks a random value $\hat{s}_i$ and executes the first round of the WSS protocol described in the previous section, acting as a dealer to share the "input" $\hat{s}_i$. We refer to this instance of the WSS protocol as WSS_i.

[3] A polynomial F is symmetric if, for all ℓ, m, the coefficient of the term $x^\ell y^m$ is equal to the coefficient of the term $x^m y^\ell$. If F is symmetric then $F(i, j) = F(j, i)$ for all i, j.

3. Let $F_i^{pad}(x,y)$ denote the bivariate polynomial used by P_i in WSS_i (i.e., $F_i^{pad}(0,0) = \hat{s}_i$). Party P_i sends the polynomial $r_i(y) := F_i^{pad}(0,y)$ to the dealer D.

Round 2: Round 2 of WSS_i is run, for all i. Concurrently, each party P_j does the following:

1. For all i, send $a_{j,i} := f_j(i)$ to P_i.
2. Let $f_{i,j}^{pad}(x)$ be the x-polynomial that P_i sent to P_j in round 1 of WSS_i. (If P_i is honest then $f_{i,j}^{pad}(x) = F_i^{pad}(x,j)$.) Party P_j sends $r'_{i,j} := f_{i,j}^{pad}(0)$ to D.

Round 3: Round 3 of WSS_i is run, for all i. Concurrently, for every ordered pair (i,j):

1. (From the viewpoint of P_i:) If $a_{j,i} \neq f_i(j)$, then P_i broadcasts ("disagree", $f_i(j)$, $F_i^{pad}(0,j)$). Otherwise, P_i broadcasts ("agree", $f_i(j) + F_i^{pad}(0,j)$).
2. (From the viewpoint of P_j:) If $a_{i,j} \neq f_j(i)$, then P_j broadcasts ("disagree", $f_j(i)$, $f_{i,j}^{pad}(0)$). Otherwise, P_j broadcasts ("agree", $f_j(i) + f_{i,j}^{pad}(0)$).
3. (From the viewpoint of D:) If $r_i(j) \neq r'_{i,j}$, then D broadcasts ("not equal", $F(j,i)$). Otherwise, D broadcasts ("equal", $F(j,i) + r_i(j)$).

Local computation. Each party locally carries out the following steps:

1. An ordered pair of parties (P_i, P_j) is *conflicting* if, in round 3, party P_i broadcasts ("disagree", $f_i(j)$, $F_i^{pad}(0,j)$); party P_j broadcasts ("disagree", $f_j(i)$, $f_{i,j}^{pad}(0)$); and it holds that $F_i^{pad}(0,j) = f_{i,j}^{pad}(0)$. For a pair of conflicting parties (P_i, P_j), we say that P_i (resp., P_j) is *unhappy* if one of the following conditions hold:

   (a) D broadcasts ("not equal", $d_{i,j}$) and $d_{i,j} \neq f_i(j)$ (resp., $d_{i,j} \neq f_j(i)$).

   (b) D broadcasts ("equal", $d_{i,j}$) and $d_{i,j} \neq f_i(j) + F_i^{pad}(0,j)$ (resp., $d_{i,j} \neq f_j(i) + f_{i,j}^{pad}(0)$).

   Let Core denote the set of parties who are not unhappy with respect to the definition above. For every P_i who was not disqualified as the dealer in WSS_i, let Core_i denote the set of parties who are not unhappy with respect to WSS_i. (If P_i was disqualified in WSS_i, then set $\mathsf{Core}_i := \emptyset$.)

2. For all i,j, remove P_j from Core_i if either of the following hold for the ordered pair (i,j) in round 3:

   - P_i broadcasts ("agree", y) and P_j did not broadcast ("agree", y).
   - P_i broadcasts ("disagree", $\star, w$) and P_j broadcasts anything other than ("disagree", $\star, w$). (Here, $\star$ denotes an arbitrary value.)

3. Remove P_i from Core if $|\mathsf{Core} \cap \mathsf{Core}_i| < n - t$. (Thus, if P_i was disqualified in WSS_i then $P_i \notin \mathsf{Core}$.)

   Note that all parties have the same view regarding Core and the $\{\mathsf{Core}_i\}$.

4. If $|\mathsf{Core}| < n - t$, then the dealer is disqualified and a default value (and appropriate 2-level shares) are shared.

5. Each party P_i computes a polynomial $\hat{f}_i(x)$ of degree at most t:

(a) If $P_i \in \mathsf{Core}$, then $\hat{f}_i(x)$ is the polynomial that P_i received from the dealer in round 1.

(b) If $P_i \notin \mathsf{Core}$, then P_i computes $\hat{f}_i(x)$ in the following way:

i. P_i first defines a set Core'_i as follows: A party P_j is in Core'_i if and only if all the following conditions hold:

- $P_j \in \mathsf{Core}$ and $P_i \in \mathsf{Core}_j$.
- Define $p_{j,k}$, for $k \in \{1, \ldots, n\}$, as follows: if, in step 1 of round 3 for the ordered pair (j, k), party P_j broadcasted ("agree", $y_{j,k}$), then set $p_{j,k} := y_{j,k}$. If P_j broadcasted ("disagree", $w_{j,k}, z_{j,k}$), then set $p_{j,k} := w_{j,k} + z_{j,k}$.

  We require that the $\{p_{j,k}\}$ are consistent with a polynomial $B_j(x)$ of degree at most t; i.e., $B_j(k) = p_{j,k}$ for all k. (If not, then P_j is not included in Core'_i.)

Our proofs show that $|\mathsf{Core}'_i| \geq t + 1$ if the dealer is not disqualified.

ii. For each $P_j \in \mathsf{Core}'_i$, set $p_j := p_{j,i} - f^{pad}_{j,i}(0)$. Let $\hat{f}_i$ be the polynomial of degree at most t such that $\hat{f}_i(j) = p_j$ for every $P_j \in \mathsf{Core}'_i$. (It will follow from our proof that such an $\hat{f}_i$ exists.)

6. Finally, P_i outputs $s_i := \hat{f}_i(0)$ and $s_{i,j} := \hat{f}_i(j)$ for all $j \in \{1, \ldots, n\}$.

Reconstruction phase. Each party P_i sends s_i to all other parties. Let $s'_{j,i}$ be the value that P_j sends to P_i. Using Reed-Solomon decoding, P_i computes a polynomial $f(x)$ of degree at most t such that $f(j) = s'_{j,i}$ for at least $2t + 1$ values of j. The final output of P_i is $f(0)$.

4.2 Proofs

We prove that the protocol given in the previous section is a VSS protocol with 2-level sharing that tolerates $t < n/3$ malicious parties.

Lemma 5 *If the dealer is not corrupted by the end of the sharing phase, privacy is preserved.*

Proof Let $\mathcal{C}$ denote the set of parties corrupted by the end of the sharing phase. We show that if the dealer remains uncorrupted, then the view of the adversary can be simulated given the polynomials $\{f_i(x)\}_{P_i \in \mathcal{C}}$. Since $F(x, y)$ is a random symmetric bivariate polynomial of degree at most t and $|\mathcal{C}| \leq t$, a standard argument (see, e.g., [4]) implies that the view of the adversary is independent of the dealer's input s.

It is immediate that the adversary learns nothing additional about s in round 2. As for the values broadcast in round 3, consider an ordered pair (P_i, P_j) of parties who remain honest throughout the sharing phase. Since the dealer is honest, we have $f_i(j) = F(j, i) = F(i, j) = f_j(i)$ and, since P_i, P_j are honest, $r_i(j) = r'_{i,j}$. Thus, in round 3, parties P_i, P_j, and the dealer all broadcast the same "blinded" value $f_i(j) + F^{pad}_i(0, j)$. Since $F^{pad}_i(0, y)$ is a random polynomial of degree at most t this does not leak any information about the $\{f_i(x)\}_{P_i \notin \mathcal{C}}$ that the adversary does not already know. ■

Lemma 6 *If the dealer is not corrupted by the end of the sharing phase, then correctness and commitment with 2-level sharing hold.*

Proof If the dealer is honest, then no honest party is unhappy. Also, all honest parties are in Core_i for any honest player P_i. Since there are at least $n - t$ honest parties, no honest party is removed from Core. It follows that the dealer is not disqualified.

Since all honest parties are in Core, each honest party P_i sets $\hat{f}_i(x) := f_i(x) = F(x,i)$. Defining $p(x) \overset{\text{def}}{=} F(0,x)$ and $p_j(x) \overset{\text{def}}{=} F(j,x)$, it is straightforward to verify that the properties of commitment with 2-level sharing hold:

- Each honest party P_i outputs $s_i := \hat{f}_i(0) = F(0,i) = p(i)$.
- For all j, it holds that $p_j(0) = F(j,0) = F(0,j) = p(j)$.
- For each honest party P_i and all $j \in \{1,\ldots,n\}$, we have
$$s_{i,j} = \hat{f}_i(j) = F(j,i) = p_j(i).$$

In the reconstruction phase, $s'_{j,i} = s_j = p(j)$ for any honest party P_j. Thus, each honest party P_i receives at most t values $s'_{j,i}$ that do not lie on the polynomial $p(x)$. It follows that P_i outputs $s = p(0) = F(0,0)$, the dealer's input. This completes the proof. ■

We now move on to show that commitment with 2-level sharing holds even when the dealer is malicious. The case of a disqualified dealer is obvious, so we focus on the case of a malicious dealer who is not disqualified. We begin by proving three claims:

Claim 7 *If the dealer is not disqualified, then for any honest P_i it holds that $|\mathsf{Core}'_i| \geq t+1$.*

Proof If the dealer was not disqualified, then Core contains at least $n-2t \geq t+1$ honest parties. We show that any honest $P_j \in \mathsf{Core}$ is also in Core'_i, proving the claim.

Since P_i and P_j are both honest, $P_i \in \mathsf{Core}_j$. Set $B(x) \overset{\text{def}}{=} f_j(x) + F_j^{pad}(0,x)$. This is a polynomial of degree at most t, and the $p_{j,k}$ computed by P_i all lie on $B_j(x)$. We conclude that $P_j \in \mathsf{Core}'_i$. ■

Claim 8 *If the dealer is not disqualified in the sharing phase, there is a bivariate symmetric polynomial $\hat{F}(x,y)$ of degree at most t in each variable that is consistent with the polynomials $\hat{f}_i$ computed by every honest party in Core; i.e., for every honest $P_i \in \mathsf{Core}$ it holds that $\hat{f}_i(x) = \hat{F}(x,i)$.*

Proof If the dealer is not disqualified, then there are at least $n-t$ parties in Core and at least $n-2t \geq t+1$ of them are honest. Let $\mathcal{H}$ denote the first $t+1$ such parties. The polynomials f_i sent by the dealer to the parties in $\mathcal{H}$ define a bivariate polynomial $\hat{F}(x,y)$ in the natural way: namely, let $\hat{F}$ be such that $\hat{F}(x,i) = f_i(x)$ for each $P_i \in \mathcal{H}$. We show that $\hat{F}$ satisfies the requirements of the claim.

By definition of $\hat{F}$, we have $\hat{f}_i(x) = f_i(x) = \hat{F}(x,i)$ for any $P_i \in \mathcal{H}$. Next, observe that for every honest $P_i, P_j \in \mathsf{Core}$ it holds that $\hat{f}_i(j) = \hat{f}_j(i)$. Indeed, it must be the case that $f_i(j) = f_j(i)$ (or else one of P_i or P_j would be unhappy), and since $P_i, P_j \in \mathsf{Core}$ we have $\hat{f}_i(x) = f_i(x)$ and $\hat{f}_j(x) = f_j(x)$. Since $\mathcal{H} \subset \mathsf{Core}$, this implies that $\hat{F}$ is symmetric. It also implies that for *every* honest $P_i \in \mathsf{Core}$ (i.e., not just the $P_i \in \mathcal{H}$) we have $\hat{f}_i(x) = \hat{F}(i,x) = \hat{F}(x,i)$, proving the claim. ■

Claim 9 *Assume the dealer is not disqualified in the sharing phase, and let $\hat{F}$ be the polynomial guaranteed to exist by Claim 8. Then for any honest $P_i \notin \mathsf{Core}$, it holds that $\hat{f}_i(x) = \hat{F}(x,i)$.*

Proof Fix an honest $P_i \notin \mathsf{Core}$, and $P_j \in \mathsf{Core}'_i$. (Claim 7 shows that Core'_i is non-empty.) By definition, this means $P_j \in \mathsf{Core}$ and $P_i \in \mathsf{Core}_j$. So P_j was not disqualified as a dealer in WSS_j and, by Lemma 4, there exists a bivariate polynomial $\hat{F}_j^{pad}$ of degree at most t in each variable such that $f^{pad}_{j,k}(x) = \hat{F}^{pad}_j(x,k)$ for all $P_k \in \mathsf{Core}_j$. (Recall that $f^{pad}_{j,k}$ denotes the polynomial that P_j sent to P_k in round 1 of WSS_j.)

Let $p_{j,k}$ be the values computed by P_i, and let $B_j(x)$ be a polynomial of degree at most t such that $B_j(k) = p_{j,k}$ for all k. Such a polynomial is guaranteed to exist because otherwise $P_j \notin \mathsf{Core}'_i$.

Since P_j remains in Core, we have $|\mathsf{Core} \cap \mathsf{Core}_j| \geq n - t$. This means that there are at least $n-2t \geq t+1$ honest parties that are in both Core and Core_j. Letting $\hat{F}$ be the symmetric polynomial guaranteed by the previous claim, we now show that for any honest $P_k \in \mathsf{Core} \cap \mathsf{Core}_j$ we have $B_j(k) = \hat{F}(k,j) + \hat{F}^{pad}_j(0,k)$. There are two cases to consider:

- If, in step 1 of round 3 for the ordered pair (j,k), party P_j broadcasted ("agree", $y_{j,k}$), then $p_{j,k} := y_{j,k}$. Since $P_k \in \mathsf{Core}_j$, this means that P_k must have broadcasted ("agree", $y_{k,j}$) with $y_{k,j} = y_{j,k}$ in step 2 of that round (cf. step 2 of the local computation phase). Since P_k is honest,

$$\begin{aligned}B_j(k) = p_{j,k} = y_{j,k} = y_{k,j} & \\ &= f_k(j) + f^{pad}_{j,k}(0) \\ &= \hat{F}(j,k) + f^{pad}_{j,k}(0) \qquad \text{(using Claim 8 and } P_k \in \mathsf{Core}) \\ &= \hat{F}(j,k) + \hat{F}^{pad}_j(0,k) \qquad \text{(since } P_k \in \mathsf{Core}_j) \\ &= \hat{F}(k,j) + \hat{F}^{pad}_j(0,k),\end{aligned}$$

  using the fact that $\hat{F}$ is symmetric.

- If, in step 1 of round 3 for the ordered pair (j,k), party P_j broadcasted ("disagree", $w_{j,k}, z_{j,k}$) then, since $P_k \in \mathsf{Core}_j$, this means that P_k must have broadcasted ("disagree", $w_{k,j}, z_{k,j}$) with $z_{k,j} = z_{j,k}$. It must also be the case that $w_{k,j} = w_{j,k}$ or else one of P_j or P_k would be unhappy. It follows that

$$B_j(k) = p_{j,k} = w_{j,k} + z_{j,k} = w_{k,j} + z_{k,j},$$

  and then an argument as before shows that $B_j(k) = \hat{F}(k,j) + \hat{F}^{pad}_j(0,k)$.

Summarizing, we have $B_j(k) = \hat{F}(k,j) + \hat{F}^{pad}_j(0,k)$ for at least $t+1$ values of k. Since $B_j(x)$ has degree at most t, this means $B_j(x) = \hat{F}(x,j) + \hat{F}^{pad}_j(0,x)$.

Party P_i next computes

$$\begin{aligned}p_j := p_{j,i} - f^{pad}_{j,i}(0) &= B_j(i) - \hat{F}^{pad}_j(0,i) \\ &= \hat{F}(i,j) + \hat{F}^{pad}_j(0,i) - \hat{F}^{pad}_j(0,i) \;=\; \hat{F}(i,j),\end{aligned}$$

using the fact that $P_i \in \mathsf{Core}_j$ in the first line. Since this is true for arbitrary $P_j \in \mathsf{Core}'_i$, we see that the polynomial $\hat{f}_i$ computed by P_i satisfies $\hat{f}_i(x) = \hat{F}(i,x) = \hat{F}(x,i)$. This completes the proof. ■

Lemma 10 *Even when the dealer is malicious, commitment with 2-level sharing holds.*

Proof By the preceding two claims, there exists a symmetric bivariate polynomial $\hat{F}(x,y)$ with degree at most t in each variable such that $\hat{f}_i(x) = \hat{F}(x,i)$ for any honest party P_i. Set $p(x) := \hat{F}(x,0)$ and $p_j(x) := \hat{F}(x,j)$. One can then verify that the properties of commitment with 2-level sharing hold:

- Each honest party P_i outputs $s_i \overset{\text{def}}{=} \hat{f}_i(0) = \hat{F}(0, i) = \hat{F}(i, 0) = p(i)$.
- At the end of the reconstruction phase, each honest party P_i will output $s' = p(0)$.
- For all j, it holds that $p_j(0) = \hat{F}(0, j) = p(j)$.
- For each honest party P_i and all $j \in \{1, \ldots, n\}$, we have

$$s_{i,j} \overset{\text{def}}{=} \hat{f}_i(j) = \hat{F}(j, i) = \hat{F}(i, j) = p_j(i).$$

This completes the proof. ■

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, 1988.

[2] G. R. Blakley. Safeguarding cryptographic keys. In *National Computer Conference*, volume 48, pages 313–317. AFIPS Press, 1979.

[3] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *26th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 383–395, 1985.

[4] R. Cramer and I. Damgard. Multiparty computation, an introduction. Lecture notes available at http://www.daimi.au.dk/˜ivan/mpc_2004.pdf.

[5] R. Cramer, I. Damgård, and U. Maurer. General secure multi-party computation from any linear secret sharing scheme. In *Adv. in Cryptology — Eurocrypt 2000*, pages 316–334. Springer-Verlag, 2000.

[6] D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly secure message transmission. *J. ACM*, 40(1):17–47, 1993.

[7] P. Feldman and S. Micali. An optimal probabilistic protocol for synchronous Byzantine agreement. *SIAM J. Computing*, 26(4):873–933, 1997.

[8] M. J. Fischer and N. A. Lynch. A lower bound for the time to assure interactive consistency. *Information Processing Letters*, 14(4):183–186, 1982.

[9] M. Fitzi and J. Garay. Efficient player-optimal protocols for strong and differential consensus. In *22nd Annual ACM Symp. on Principles of Distributed Computing*, pages 211–220, 2003.

[10] M. Fitzi, J. A. Garay, S. Gollakota, C. P. Rangan, and K. Srinathan. Round-optimal and efficient verifiable secret sharing. In *3rd Theory of Cryptography Conference (TCC)*, pages 329–342, 2006.

[11] R. Gennaro, Y. Ishai, E. Kushilevitz, and T. Rabin. The round complexity of verifiable secret sharing and secure multicast. In *33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 580–589, 2001.

[12] J. Katz and C.-Y. Koo. On expected constant-round protocols for Byzantine agreement. In *Advances in Cryptology — Crypto 2006*, pages 445–462. Springer-Verlag, 2006.

[13] J. Katz and C.-Y. Koo. Round-efficient secure computation in point-to-point networks. In *Advances in Cryptology — Eurocrypt 2007*, pages 311–328. Springer-Verlag, 2007.

[14] C. Koo. *Studies on Fault-Tolerant Broadcast and Secure Computation.* PhD thesis, University of Maryland, 2007.

[15] Y. Lindell, A. Lysyanskaya, and T. Rabin. Sequential composition of protocols without simultaneous termination. In *21st Annual ACM Symposium on Principles of Distributed Computing*, pages 203–212, 2002.

[16] S. Micali and T. Rabin. Collective coin tossing without assumptions nor broadcasting. In *Adv. in Cryptology — Crypto '90*, pages 253–266. Springer-Verlag, 1990.

[17] A. Patra, A. Choudhary, B. Ashwinkumar, and C. Rangan. Probabilistic verifiable secret sharing tolerating an adaptive adversary. Available at http://eprint.iacr.org/2008/101.

[18] T. Rabin and M. Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority. In *21st Annual ACM Symposium on Theory of Computing*, pages 73–85, 1989.

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