Regular Paper

A Share-Correctable Protocol for the Shamir Threshold Scheme and Its Application to Participant Enrollment

Raylin Tso,† Ying Miao,† Takeshi Okamoto† and Eiji Okamoto†

Verifiable secret sharing schemes proposed so far can only allow participants to verify whether their shares are correct or not. In this paper, we propose a new protocol which can allow participants not only to verify the correctness of their shares but also to revise the faulty shares. It is achieved in a cooperative way by participants, but without any assistance from the dealer. This protocol, to the best of our knowledge, is the first one providing such kind of ability. Correcting shares by participants instead of the dealer is important in many situations. In addition, this protocol is also useful for adding new participants without the dealer's assistance.

1. Introduction

Informally, a (k, n)-threshold scheme is a way of distributing partial information called shares to n participants in order to allow any k of them to make an action (e.g., to find a secret value K or to open a vault in a bank), but also to ensure that the action cannot be made by any subset of fewer than k participants.

Threshold schemes based on finite geometries and polynomial interpolations were introduced independently by Blakley [3] and Shamir [14] in 1979. Since then, many constructions have been given for such cryptographic schemes [9),10),13),16),17)]. In the Shamir (k, n)-threshold scheme, a secret $K \in GF(p)$, p a prime greater than n, is distributed by a special participant called *dealer* (D as an abbreviation) to a set $\mathcal{P}$ of participants in the following way.

(1) D chooses n distinct non-zero elements of $GF(p)$, denoted x_l, $1 \le l \le n$. For $1 \le l \le n$, D sends the value x_l to $P_l \in \mathcal{P}$ through a public channel.
(2) D secretly chooses, independently at random, $k-1$ elements of $GF(p)$, $a_1, \ldots, a_{k-1}$.
(3) For $1 \le l \le n$, D computes $y_l = f(x_l)$, where $f(x) = K + \sum_{1 \le j \le k-1} a_j x^j \in GF(p)[x]$ and sends the share y_l to $P_l \in \mathcal{P}$ through a private secure channel.

At a later time, a subset of participants $B \subseteq \mathcal{P}$ will pool their shares in an attempt to compute the secret K. If $|B| \ge k$, then they should be able to compute the value of K; if $|B| < k$, then they should not be able to obtain any information about K.

In many applications, it may be possible that the dealer D does not trust the participants completely, and the participants do not trust D either. For this reason, verifiable secret sharing (VSS) schemes were proposed. A VSS scheme enables each participant to verify whether the share he received from D is consistent with other shares or not, and/or to check whether each pooled share is indeed correct or not. Chor et al. [5] first introduced the notion of a VSS scheme. Since then, VSS schemes have been studied by numerous authors (e.g., Refs. 2), 5), 7), 8) and 10)).

On the other hand, VSS schemes proposed so far do not provide the ability for participants to revise their shares if some of these shares have been verified to be incorrect. What they can do after finding faulty shares is to make complains to the dealer and ask him to re-distribute new shares for them, then to verify the new shares again by the VSS scheme they used previously. We notice that in the following situations, most of the current VSS schemes may become ineffectual.

(1) In the case if, for security purpose, D is not permitted to preserve any information about the shares and the secret after distributing the shares.
(2) In the case if the dealer D, after sending shares to participants, becomes inactive.

Also, in the real world, participants may not do the verification right after receiving their shares from D. They may do that at any time before the secret-recovery phase. There-

† Graduate School of Systems and Information Engineering, University of Tsukuba

fore, when using a common VSS scheme, it is necessary that D preserves all the information concerning the shares and the secret before all the participants do the verification. If D loses such information before the verification phase, the faulty shares will not be able to be corrected successfully. On the other hand, if D pays scant attention to security management, an adversary may steal information about the secret or the shares of the scheme by attacking the dealer's storage where all of the secret information are stored on. Therefore, the adversary can avoid the difficulty of attacking the VSS scheme directly but get the secret information of the scheme. From these viewpoints, it is undesirable that D preserves any secret information about the scheme for a long period of time. Case 2 is also possible if an adversary disturbs the communications between D and the participants after the initial phase.

In this paper, we propose a new protocol for share-verification and share-correction which can overcome the drawbacks described above. Our newly proposed protocol allows participants not only to verify their shares but also to revise the faulty shares without the dealer's assistance. Though this goal is achieved in a cooperative way by participants, this protocol is, to the best of our knowledge, the first VSS protocol providing such kind of ability. We emphasize that error-correcting codes such as Reed-Solomon codes [12] can correct errors only during the phase of pooling shares together which would then reveal the secret.

Our protocol has the following features.

- The dealer D can destroy all of the information about the secret and the shares of the scheme after distributing the shares to the participants.
- Our protocol is an auxiliary for the Shamir threshold scheme. That is, if D is reliable, participants can just use the "original" Shamir threshold scheme and can avoid the use of the bothersome and/or resources-consuming VSS scheme. At any time when they feel doubtful about the sincerity of D, they can apply the proposed method to verify and revise their shares.
- Only $k + 2(t + c) \leq n$ of the n participants are needed to take part in the protocol whereas all the shares can be verified and revised (without revealing any information about the secret and shares of participants, of course.) Those participants not taking part in the protocol can verify and revise (if necessary) their shares according to the public and secret information obtained from other participants. Here k is the threshold value, t is the maximum number of faulty shares participants get from the dealer D and c is the maximum number of cheaters (dishonest participants).
- No secure channel between participants is required.
- One restriction of our protocol is that it can only be applied in the Shamir (k, n)-threshold scheme with $n \geq k + 2(t + c)$, where $t + c \leq min\{k, \lfloor \frac{n-k}{2} \rfloor\}$.

On the other hand, this protocol can also be utilized for other purposes. We show in Section 4 that this protocol is also useful for adding new participants without the dealer's assistance in a threshold scheme with cheaters.

The rest of this paper is organized as follows. In Section 2, we give some preliminaries which will be used in our protocol. Section 3 is the illustration of our protocol. Section 4 explains the application of our protocol to participant enrollment without any assistance of the dealer in a threshold scheme with cheaters. Finally, Section 5 makes the conclusion of this paper.

2. Preliminaries

2.1 Homomorphism Property

In Ref. 1), Benaloh introduced a homomorphism property in secret sharing, which implies that the compositions of shares of several schemes are shares of the composition of these schemes.

Definition 2.1 A function F is said to have $(\oplus, \otimes)$-homomorphism property (or $(\oplus, \otimes)$-homomorphic) if

$$F(x_1, x_2, \ldots, x_n) \oplus F(y_1, y_2, \ldots, y_n) = F(x_1 \otimes y_1, x_2 \otimes y_2, \ldots, x_n \otimes y_n)$$

This property implies that the reconstruction from the combined shares results in a combined secret of several secret sharing schemes. It is easy to see that the Shamir threshold scheme is $(+, +)$-homomorphic.

2.2 TMO Algorithm [17]

In Ref. 13), Rees, et al. considered the problem of determining consistent sets of shares in a (k, n)-threshold scheme with cheaters (i.e., dishonest participants). Their underlying idea is to find a suitable set system $(S, \mathcal{T})$, where S is

the set of all n shares and $\mathcal{T}$ is a collection of k-subsets of S, so that for any t'-subset $S_{t'}$ of shares (and thus the subset of all fake shares, if we assume that at most t' of the n shares are fake), there is at least one $T \in \mathcal{T}$ such that T does not contain any share in this t'-subset S'_t. Then the k-subset $T \in \mathcal{T}$ containing no fake shares can be used to derive the secret correctly. One drawback of Rees, et al.'s algorithms is that they sacrifice the property of *threshold*. That is, no honest participant can be absent if they decide to reconstruct the secret while in a (k, n)-threshold scheme, only k of the n participants are needed to pool their shares. This drawback is improved by Tso, Miao and Okamoto in Ref. 17). In both Refs. 13) and 17), they applied a combinatorial structure called *covering* to their schemes which provides an upper bound on the number of iterations required in their algorithms.

Definition 2.2 [17] Let v, k and t' be positive integers such that $v \geq k \geq t'$. A (v, k, t')-*covering* is a pair $(\mathcal{V}, \mathcal{B})$, where $\mathcal{V}$ is a v-set of elements, called *points*, and $\mathcal{B}$ is a collection of k-subsets of $\mathcal{V}$, called *blocks*, such that every t'-subset of points occurs in at least one block of $\mathcal{B}$. The *covering number* $C(v, k, t')$ is the minimum number of blocks in any (v, k, t')-covering. A (v, k, t')-covering $(\mathcal{V}, \mathcal{B})$ is *optimal* if $|\mathcal{B}| = C(v, k, t')$.

Suppose that the Shamir (k, n)-threshold scheme is implemented in $GF(p)$. Let $S = \{(x_i, y_i) : 1 \leq i \leq n\} \subseteq (GF(p) \setminus \{0\}) \times GF(p)$ be the set of n shares, and assume that at most t' of the n shares are fake. That is, there exists a polynomial $P_0(x) \in GF(p)[x]$ of degree at most $k-1$ such that $y_i = P_0(x_i)$ for at least $n - t'$ of the n shares. The secret, which can be reconstructed from any k genuine shares, is the value $P_0(0)$. In addition, define

- $\mathcal{M}$: a $(k + 2t')$-subset of $\{1, 2, \ldots, n\}$.
- $S_{\mathcal{M}} : \{(x_i, y_i) : i \in \mathcal{M}\} \subseteq S$.
- $\mathcal{T}$: a collection of k-subsets of $\mathcal{M}$ such that its complement $\{\mathcal{M} \setminus T : T \in \mathcal{T}\}$ is the collection of blocks of a $(k+2t', 2t', t')$-covering with minimum number of blocks.
- P_T : the unique polynomial of degree at most $k-1$ reconstructed by the subset $T \in \mathcal{T}$.

Moreover, define $C_T = \{i : P_T(x_i) = y_i, 1 \leq i \leq n\}$ and $NC_T = \{i : P_T(x_i) \neq y_i, 1 \leq i \leq n\}$. Then, Tso, Miao and Okamoto's algorithm [17] (TMO algorithm as an abbreviation) can be outlined as follows. In this algorithm, we denote $\mathcal{M} \setminus T = \{r_{i_1}, \ldots, r_{i_{2t'}}\}$ for each $T \in \mathcal{T}$, where the subscripts are ordered randomly.

TMO Algorithm [17]
Input $\mathcal{M}, \mathcal{T}, S_{\mathcal{M}}, k, t'$.

For each $T \in \mathcal{T}$, perform the following steps:

(1) compute P_T
(2) $|NC_T| = 0$
(3) $|C_T| = k$
(4) **for** $j = 1$ **to** $2t'$ **do**
(5) **if** $y_{r_{i_j}} = P_T(x_{r_{i_j}})$, **then** $|C_T| + +$
(6) **else** $|NC_T| + +$
(7) **if** $|C_T| \geq k + t'$, **then** $P_0 = P_T$ and QUIT
(8) **else if** $|NC_T| \geq t' + 1$ **then** BREAK

The TMO algorithm allows any $k + 2t'$ of the n participants to achieve the end of determining a consistent set of shares in a threshold scheme with at most t' cheaters.

2.3 Publicly Verifiable Secret Sharing Scheme

A publicly verifiable secret sharing (PVSS) scheme is a special type of VSS scheme in which the validity of the shares distributed by the dealer D can be verified by any entity instead of the shareholders only. Here we first review a basic type of VSS scheme in which the security is based on the intractability of the discrete logarithm problem, then we describe the Stadler PVSS scheme [15] based on this basic VSS scheme. We will adopt the Stadler scheme later as a sub-protocol in our scheme.

Basic Type of VSS Scheme

Let

- p be a large prime so that $q = (p-1)/2$ is also a prime.
- g be a generator of $GF(p) \setminus \{0\}$ so that computing discrete logarithms to the base g is difficult.

To share a secret $K \in GF(p)$ in a (k, n)-VSS scheme, the initial setting is the same as that of the Shamir threshold scheme. $x_i \in GF(p) \setminus \{0\}$ is a publicly known element assigned to participant $P_i, 1 \leq i \leq n$. $f(x) = K + \sum_{1 \leq j \leq k-1} a_j x^j \in GF(p)[x]$ is the polynomial D secretly chosen and $y_i = f(x_i) \pmod p$ is the secret share of P_i obtained from D through a private secure channel. Beside these, D publishes the values $A = g^K$ and $F_j = g^{a_j}$, $j = 1, \ldots, k-1$. Any group of at least k participants can compute the secret $K \in GF(p)$ using the

Prover		Verifier
repeat l times:		
$w \in_R GF(q)$		
$t_h = h^w \pmod p$		
$t_g = g^{z_i^w}$	$t_h, t_g \longrightarrow$	
		$c \in_R \{0,1\}$
	$\longleftarrow c$	
$r = w - c \cdot \alpha \pmod q$		
	$r \longrightarrow$	
		$t_h \stackrel{?}{=} h^r \cdot M_1^c \pmod p$
		$t_g \stackrel{?}{=} \begin{cases} g^{z_i^r} & \text{if } c = 0 \\ Y_i^{M_2 \cdot z_i^r} & \text{if } c = 1 \end{cases}$

Lagrange interpolation formula. In addition, any participant P_i can verify his/her share y_i by computing $Y_i = A \cdot \prod_{j=1}^{k-1} F_j^{x_i^j}$ and checking whether $Y_i = g^{y_i}$.

Stadler PVSS Scheme [15]

To make this scheme publicly verifiable, the private secure channels between D and participants are replaced by public key encryption schemes. In the Stadler PVSS scheme, the encryption scheme is identical to the ElGamal public key crypyosystem [6]. First, let $h \in GF(p) \setminus \{0\}$ of order q be a public information selected by D, then each participant P_i randomly chooses a secret key $s_i \in GF(q)$ and publishes his/her public key $z_i = h^{s_i} \pmod p$. To distribute the share $y_i \in GF(p) \setminus \{0\}$ to P_i secretly, D encrypts y_i with P_i's public key z_i. D also randomly chooses $\alpha \in GF(q) \setminus \{0\}$ and then calculates the pair (M_1, M_2) where $M_1 = h^\alpha \pmod p$, and $M_2 = y_i^{-1} z_i^\alpha \pmod p$. If $y_i = 0 \pmod p$ for some i, then D should choose another $x_i \in GF(p) \setminus \{0\}$ for P_i or choose another polynomial so that $y_i \neq 0 \pmod p$ for all i. The ciphertext (M_1, M_2) can only be decrypted by P_i since $y_i = M_1^{s_i}/M_2 \pmod p$.

To verify the shares, the prover (i.e., the dealer D) proves to the verifier (i.e., any entity instead of the shareholders only) that the discrete logarithm of M_1 to the base h is equal to the double discrete logarithm of $Y_i^{M_2}$ to the bases g and z_i. It is based on the fact that if (M_1, M_2) is equal to $(h^\alpha, y_i^{-1} z_i^\alpha) \pmod p$ for some $\alpha \in GF(q) \setminus \{0\}$, then

$$Y_i^{M_2} = g^{y_i M_2} = g^{z_i^\alpha}.$$

Consequently, the probability for the prover to deceive a verifier successfully when repeating the proof-verification protocol l times is $1/2^l$. This verification can also be done noninteractively. The interested readers are referred to Ref. 15) for more details.

3. Proposed Method

This section describes our share-verification protocol which is a combination of the above mentioned methods. The feature of our protocol is that it provides the ability for participants to revise faulty shares in a cooperative way without the dealer's assistant. One restriction of our protocol is that the number of participants n must be greater than or equal to $k + 2(t + c)$, where k is the threshold value, t is the maximum number of faulty shares and c is the maximum number of cheaters (dishonest participants). The parameters p, q, g are the same as those in Section 2.3. In the initial phase, the dealer D shares a secret $K \in GF(p)$ to n participants according to the Shamir (k, n)-threshold scheme. The polynomial D secretly chosen is $f(x) \in GF(p)[x]$, and the share for participant P_i is (x_i, y_i), where x_i is the public information for participant P_i and $y_i = f(x_i)$ for $1 \leq i \leq n$. After this initial phase, D destroys all the secret information about the scheme for security purpose.

If no participant doubts the sincerity of the dealer D, then the procedure for verification is not needed. In this case, the secret sharing scheme is just that of the "original" Shamir threshold scheme, which is believed to be much more efficient than any kind of VSS schemes. Since the dealer D has destroyed all the secret information about the scheme, no "current" VSS protocol is usable if any participant feels doubtful about the correctness of his/her share later. In this situation, with the coopera-

tion of at least $k+2(t+c)-1$ of the other $n-1$ participants, our protocol can be applied and the verification of their shares can be executed.

W.l.o.g., we assume $Q=\{P_1,P_2,\cdots,P_{k+2(t+c)}\}$ be the $(k+2(t+c))$-subset of the n participants which will take part in the share-verification protocol. According to Definition 2.2 and the TMO algorithm, the value of $t+c$ should be less than or equal to k and $\lfloor\frac{n-k}{2}\rfloor$, that is, $t+c \leq min\{k, \lfloor\frac{n-k}{2}\rfloor\}$. We also assume that there is a public information[☆] $h \in GF(p)\setminus\{0\}$ of order q and a bulletin board available for all the participants. Before the verification phase, each participant P_i, $1 \leq i \leq n$, randomly chooses a secret key $s_i \in GF(q)$ and publishes his/her public key $z_i = h^{s_i} \pmod p$ so that other participants can send encrypted message to him/her using the ElGamal public key cryptosystem.

Share-Correctable Protocol

(1) Each participant $P_i \in Q$ in turn plays the role of the dealer of the Stadler PVSS scheme [15]. That is, P_i secretly selects a polynomial $g_i(x) = a_{i,0} + a_{i,1}x + \cdots + a_{i,k-1}x^{k-1} \in GF(p)[x]$ of degree $k-1$ and sends the related share $d_{i_j} = g_i(x_j) \pmod p$ to participant P_j for $1 \leq j \leq n$ using P_j's public key. Here x_j for P_j is the same as that the original dealer D chose for P_j. The public information $A_i(=g^{a_{i,0}}), F_{i,l}(=g^{a_{i,l}}), 1 \leq l \leq k-1$, is also published by P_i. At the end of this stage, each participant P_j, $1 \leq j \leq n$, has shares d_{i_j} from P_i, $1 \leq i \leq k+2(t+c)$, and the original share $y_j = f(x_j)$ from the dealer D. We will call the shares d_{i_j} for all i and j the auxiliary shares for the convenience of notation.

(2) Each participant $P_i \in Q$ verifies the auxiliary shares according to the Stadler PVSS scheme. Participants distributing incorrect auxiliary shares to other participants intentionally will be disclosed in this stage. Also, any cooperation between cheaters (dishonest participants) is invalid because all the auxiliary shares are publicly verifiable. Note that not all the dishonest participants may cheat in this stage.

(3) Participants in Q make complains on a bulletin board against the dishonest participants who distributed incorrect shares. They also abandon the auxiliary shares obtained from those dishonest participants. Other participants not in Q can also know which auxiliary shares should be abandoned from the complains on the bulletin board. (Of course, participants not in Q can also do the verification of the auxiliary shares if they would like.)

(Since there are at most c cheaters, each honest participant abandons at most c auxiliary shares from other participants. In other words, each participant retains at least $k+2t+c(>k)$ auxiliary shares. All of these auxiliary shares will be used in the next steps. W.l.o.g., we may assume that each honest participant P_j retains exactly $k+2t+c$ auxiliary shares d_{i_j} for $1 \leq i \leq k+2t+c$.)

(4) Each P_j sums up his/her share y_j with his/her remaining auxiliary shares d_{i_j}, $1 \leq i \leq k+2t+c$. Consequently, each P_j, $1 \leq j \leq n$, has a summed share $u_j = y_j + d_{1_j} + \cdots + d_{(k+2t+c)_j} \pmod p$.

(5) Participants in Q broadcast their summed shares u_j, $1 \leq j \leq k+2(t+c)$, on the bulletin board (note again that there are at most c participants who may intentionally broadcast incorrect values at this stage).

(6) Any participant P_j who wants to verify and revise his/her share y_j can apply the TMO algorithm. First replace t' described in Section 2.2 with $t+c$, and use the $k+2(t+c)$ broadcasted information on the bulletin board (at most $t+c$ of the information may be incorrect), then he/she will derive a unique polynomial $H(x)$, where

$$H(x) = f(x) + g_1(x) + \cdots + g_{k+2t+c}(x) \in GF(p)[x].$$

(7) Participant P_j in step 6 verifies if $H(x_j) = u_j \pmod p$. If not, then revise his/her share y_j to the value of $H(x_j) - d_{1_j} - d_{2_j} - \cdots - d_{(k+2t+c)_j} \pmod p$.

In step 5, at most c cheaters may broadcast incorrect values of their summed shares. On the other hand, there are at most t faulty shares dis-

[☆] This information can be pre-selected by D or by the cooperation of participants before the share-verification phase.

tributed by the dealer D. Therefore, in order to use the TMO algorithm to construct the unique polynomial $H(x)$, we need at least $k + 2(t + c)$ participants in our protocol.

Security Analysis

No adversary can obtain any secret information from step 1 to step 3, since he/she suffers from the intractability of the discrete logarithm problem. Also note that although the polynomial $H(x)$ can be derived by any participant, the polynomials $f(x)$ and $g_i(x)$, $1 \leq i \leq k + 2t + c$, are still kept secret because of the homomorphism property. A conspiracy of less than k participants still can not get $f(x)$ from $H(x)$ or $g_i(x)$ from their reconstruction because $f(x)$ is still masked by at least $2t + c$ polynomials $g_{i'}(x)$ constructed by other participants. Consequently, no secret information about the secret and the shares belonging to other participants will be leaked out in this protocol.

4. Adding Participants without the Dealer's Assistance

In most of the current secret sharing schemes, only dealers have the ability to enroll new participants in their schemes. Also, in order to achieve this goal of adding new participants, dealers have to preserve some or all of the secret information of their schemes. This results in the same problem as we have claimed at the beginning of this paper. We claimed in Section 1 that for security purpose, it is undesirable that the dealer preserves any secret information about the scheme for a long period of time. On the other hand, the goal of adding new participants into the scheme may be impossible if the dealer becomes inactive after the initial phase. These problems can be solved by using our protocol. We explain it in the following.

Assume the dealer D is honest so no faulty shares have been distributed to participants ☆. Then, with a little modification, the protocol we have described in Section 3 can also be used as a protocol for adding new participants in the scheme without the dealer's assistance. Moreover, according to Definition 2.2 and the TMO algorithm, c cheaters with $c \leq min\{k, \lfloor \frac{n-k}{2} \rfloor\}$ are tolerant in the scheme.

Under the agreements of at least $k + 2c$ participants, by the cooperation of these participants, any group of, say n', new participants can get their shares of the scheme in only one run of the protocol where $n + n' < p$. Assume N is the n-set of all participants in the Shamir (k, n)-threshold scheme, N' is the n'-set of new participants that want to join in the scheme where $N' \cap N = \emptyset$. In the same fashion as that described in Section 3, w.l.o.g., we assume $Q = \{P_1, P_2, \cdots, P_{k+2c}\}$ is the subset of N which will cooperate to construct and distribute new shares to participants in N'. In addition, $h \in GF(p) \setminus \{0\}$ of order q is a public known information and there is a bulletin board available for all participants. With these assumptions, then the modified protocol for new participants enrollment can be described as follows.

Enrollment Protocol

(1) Each participant $P_i \in Q$ randomly chooses an $s_i \in GF(q)$ as his/her secret key and publishes $z_i = h^{s_i} \pmod p$ as his/her public key.

(2) Each participant $P'_j \in N'$ randomly chooses an $s'_j \in GF(q)$ as his/her secret key and publishes $z'_j = h^{s'_j} \pmod p$ as his/her public key. He/She also randomly chooses a random value $x'_j \in GF(p) \setminus \{0\}$ and publishes it as his/her public information of the threshold scheme. Here x'_j must be different from other participants' public information.

(3) Each participant $P_i \in Q$ in turn plays the role of the dealer of the Stadler PVSS scheme. P_i secretly selects a polynomial $g_i(x) = a_{i,0} + a_{i,1}x + \cdots + a_{i,k-1}x^{k-1} \in GF(p)[x]$ of degree $k-1$ and sends the related share (auxiliary share) $d_{i_j} = g_i(x_j) \pmod p$ to participants $P_j \in Q$ and $P'_j \in N'$ using P_j and P'_j's public keys. Here x_j for $P_j \in Q$ is the same as that the original dealer D chose for P_j. x'_j for $P'_j \in N'$ is the value P'_j chose and published in step (2). P_i also publishes $A_i(= g^{a_{i,0}}), F_{i,l}(= g^{a_{i,l}}), 1 \leq l \leq k-1$. At the end of this stage, each participant $P_j \in Q$ and $P'_j \in N'$ has auxiliary shares d_{i_j} from P_i, $1 \leq i \leq k + 2c$. In addition $P_j \in Q$ also has the original share $y_j = f(x_j)$ from the dealer D.

(4) Each participant $P_j \in Q$ and $P'_j \in N'$ verifies the auxiliary shares according to the Stadler PVSS scheme. Participants

☆ If D may be dishonest, participants can first apply our protocol to revise their faulty shares.

distributing incorrect auxiliary shares to other participants intentionally will be disclosed in this stage. Also, any cooperation between dishonest participants is invalid because all the auxiliary shares are publicly verifiable.

(5) Participants in step (4) make complains on the bulletin board against the dishonest participants who distributed incorrect shares. They also abandon the auxiliary shares obtained from those dishonest participants.

(Since there are at most c cheaters, for the convenience of description, we assume that there remain exactly $k + c$ auxiliary shares d_{i_j} for each participant $P_j \in Q$ and $P'_j \in N'$, where $1 \leq i \leq k+c$ and j is the subscription of $P_j \in Q$ and $P'_j \in N'$.)

(6) Each $P_j \in Q$ sums up his/her share y_j with his/her remaining auxiliary shares d_{i_j}, $1 \leq i \leq k + c$. Consequently, each $P_j \in Q$ has a summed share $u_j = y_j + d_{1_j} + \cdots + d_{(k+c)_j} \pmod p$.

(7) Participants $P_j \in Q$ broadcast their summed shares u_j, $1 \leq j \leq k+2c$, on the bulletin board. (Note again that there are at most c participants who may intentionally broadcast incorrect values at this stage.)

(8) Each participant $P'_j \in N'$ applies the TMO algorithm by first replacing t' described in Section 2.2 with c, and then using the $k + 2c$ broadcasted information u_j, $1 \leq j \leq k + 2c$, on the bulletin board (at most c of the information may be incorrect). Then he/she can derive a unique polynomial $H(x)$, where $H(x) = f(x) + g_1(x) + \cdots + g_{k+c}(x) \in GF(p)[x]$.

(9) $P'_j \in N'$ derives his/her share y'_j by computing
$$y'_i = H(x'_i) - d_{1_{i'}} - d_{2_{i'}} - \cdots - d_{(k+c)_{i'}} \pmod p,$$
where $d_{j_{i'}} = g_j(x'_i)$ are the remaining auxiliary shares after step (5) obtained from participants $P_j \in Q$.

Consequently, every new participant P'_i in N' obtains a share y'_i of the scheme and the (k, n)-threshold scheme has modified into a $(k, n+n')$-threshold scheme successfully.

Security Analysis

The security of this protocol can be analyzed in the same way as that in Section 3. In addition, only a cooperation of at least $k+2c$ ($\geq k$) participants can have the ability of adding new participants to the original scheme. Therefore, it is impossible for a conspiracy of less than k malicious participants to execute this protocol and add new participants siding with them for the purpose of reconstructing the secret of the original (k, n)-threshold scheme.

5. Conclusion

Most verifiable secret sharing schemes proposed so far do not provide the ability for participants to correct the faults of their shares. Error-correcting codes such as Reed-Solomon codes can correct errors only during the phase of pooling shares together which would then reveal the secret. In this paper, we proposed a new type of share-verification protocol for the Shamir threshold scheme. Our protocol allows participants not only to verify the correctness of their shares but also to revise any fault of their shares in a cooperative way without any assistance of the dealer. We also showed that our protocol can be utilized to add new participants in a threshold scheme with cheaters without the dealer's assistance.

Acknowledgments The authors wish to thank the two anonymous reviewers for their constructive comments which greatly clarified the description of our protocol.

References

1) Benaloh, J.C.: Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret, *Advances in Cryptology — CRYPTO'86*, Lecture Notes in Comput. Sci., Vol.263, pp.251–260 (1987).
2) Ben-Or, M., Goldwasser, S. and Widgerson, A.: Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation, *Proc. 20th Annual ACM Symp. on Theory of Computing*, pp.1–10 (1988).
3) Blakley, G.: Safeguarding Cryptographic Keys, *Proc. AFIPS National Computer Conference*, pp.313–317 (1979).
4) Chaum, D., Crépeau, C. and Damgard, I.: Multiparty Unconditionally Secure Protocols, *Proc. 20th Annual ACM Symp. on Theory of Computing*, pp.11–19 (1988).
5) Chor, B., Goldwasser, S., Micali, S. and Awerbuch, B.: Verifiable Secret Sharing and Achiving Simultaneity in the Presence of Faults, *Proc. 26th IEEE Ann. Symp. on the Foundations of Comput. Sci.*, pp.383–395

(1985).
6) ElGamal, T.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, *Advances in Cryptology — CRYPTO'84*, Lecture Notes in Comput. Sci., Vol.196, pp.10–18 (1985).
7) Feldman, P.: A Practical Scheme for Non-Interactive Verifiable Secret Sharing, *Proc. 28th IEEE Ann. Symp. on the Foundations of Comput. Sci.*, pp.427–437 (1987).
8) Harn, L.: Efficient Sharing (Broadcasting) of Multiple Secrets, *IEE Proc. -Comput. Digit. Tech.*, Vol.142, pp.237–240 (1995).
9) McEliece, R.J. and Sarwate, D.V.: On Sharing Secrets and Reed-Solomon Codes, *Comm. ACM*, Vol.24, pp.583–584 (1981).
10) Pedersen, T.P.: Non-Interactive and Information-Theoretic Secret Sharing, *Advances in Cryptology — CRYPTO'91*, Lecture Notes in Comput. Sci., Vol.576, pp.129–140 (1992).
11) Rabin, T. and Ben-Or, M.: Verifiable Secret Sharing and Multiparty Protocols with Honest Majority, *Proc. 21st Annual ACM Symp. on Theory of Computing*, pp.73–85 (1989).
12) Reed, I.S. and Solomon, G.: Polynomial Codes Over Certain Finite Fields, *J. Soc. Indust. Appl. Math.*, Vol.8, pp.300–304 (1960).
13) Rees, R.S., Stinson, D.R., Wei, R. and van Rees, G.H.J.: An Application of Covering Designs: Determining the Maximum Consistent Set of Shares in a Threshold Scheme, *Ars Combin.*, Vol.53, pp. 225–237 (1999).
14) Shamir, A.: How To Share a Secret, *Comm. ACM*, Vol.22, pp.612–613 (1979).
15) Stadler, M.: Publicly Verifiable Secret Sharing, *Advances in Cryptology — EUROCRYPT'96*, Lecture Notes in Comput. Sci., Vol.1070, pp.190–199 (1996).
16) Tompa, M. and Woll, H.: How to Share a Secret with Cheaters, *J. Cryptology*, Vol.1, pp.133–138 (1988).
17) Tso, R., Miao, Y. and Okamoto, E.: A New Algorithm for Searching a Consistent Set of Shares in a Threshold Scheme with Cheaters, *Information Security and Cryptology — ICISC 2003*, Lecture Notes in Comput. Sci., Vol.2971, pp.377–385 (2004).

(Received December 1, 2004)
(Accepted June 9, 2005)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.1, pp.313–321.)

Rayiln Tso received his B.S. degree in industrial engineering from National Tsing Hua University, Taiwan in 1995, his M.E. degrees in Business Administration and Public Policy in 2000 and Systems and Information Engineering in 2002 from Univeristy of Tsukuba, Japan. Currently, he is working towards the Ph.D. in Systems and Information Engineering at University of Tsukuba. His research interests include cryptography.

Ying Miao received his B.S. degree from Wuhan University, China, in 1985, his M.S. degree from Suzhou University, China, in 1989, and his D. Sci. degree from Hiroshima University, Japan, in 1997, all in mathematics. From 1989 to 1993, he worked at Suzhou Institute of Silk Textile Technology, China. From 1995 to 1997, he was a research fellow of the Japan Society for the Promotion of Science. During 1997–1998, he was a postdoctoral fellow in the Department of Computer Science, Concordia University, Canada. Since 1998, he has been with the University of Tsukuba, Japan, where he is now an associate professor at the Graduate School of Systems and Information Engineering. His research interests include combinatorial design theory, coding theory, cryptography, and their interactions.

Takeshi Okamoto received B.E. degree from Kyoto Institute of Technology in 1996, and M.I.S. and Dr.I.S. degrees from JAIST (Japan Advanced Institute of Science and Technology) in 1999 and 2002, respectively. From 2002 to 2003, he was an instructor at Department of Information Sciences, Tokyo Denki University. He is an assistant professor at Graduate School of Systems and Information Engineering, University of Tsukuba. His current research interests include cryptography and information security.

Eiji Okamoto received his B.S., M.S. and Ph.D. degrees in electronics engineering from the Tokyo Institute of Technology in 1973, 1975 and 1978, respectively. He worked and studied communication theory and cryptography for NEC central research laboratories since 1978. Then, he became a professor at JAIST (Japan Advanced Institute of Science and Technology) from 1991, and at Toho University from 1999 untill 2002. He is currently a professor at Graduate School of Systems and Information Engineering, University of Tsukuba. His research interests are cryptography and information security.