Regular Paper

How to Verify the Threshold t of Shamir's (t, n)-Threshold Scheme

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

In the Shamir (t, n)-threshold scheme, the dealer constructs a random polynomial $f(x) \in GF(p)[x]$ of degree *at most* $t-1$ in which the constant term is the secret $K \in GF(p)$. However, if the chosen polynomial $f(x)$ is of degree less than $t-1$, then a conspiracy of any $t-1$ participants can reconstruct the secret K; on the other hand, if the degree of $f(x)$ is greater than $t-1$, then even t participants can not reconstruct the secret K properly. To prevent these from happening, the degree of the polynomial $f(x)$ should be exactly equal to $t-1$ if the dealer claimed that the threshold of this scheme is t. There also should be some ways for participants to verify whether the threshold is exactly t or not. A few known verifiable threshold schemes provide such ability but the securities of these schemes are based on some cryptographic assumptions. The purpose of this paper is to propose some threshold-verification protocols for the Shamir (t, n)-threshold scheme from the viewpoint of unconditional security.

1. Introduction

Consider the problem of n trustees where any t of them are needed to be in agreement to make an action (e.g., to open a vault in a bank), and in addition, if less than t trustees are in agreement, they should not be able to make such an action. Solutions to this type of problems are called (t, n)-threshold schemes.

Threshold schemes based on finite geometries and polynomial interpolations were introduced independently by Blakley [2] and Shamir [11] in 1979. They are the first well-known examples of secret sharing schemes. A *secret sharing scheme* is a way of sharing a secret K by distributing partial information called *shares* to a set of participants $\mathcal{P}$ in such a way that authorized subsets of the participants can reconstruct the secret K, whereas any non-authorized subsets of $\mathcal{P}$ can determine nothing about K. The value of K is chosen and distributed by a *trustworthy* participant $D \notin \mathcal{P}$, often called the *dealer*.

In the Shamir (t, n)-threshold scheme, a secret $K \in GF(p)$, p being a prime greater than n, is distributed by a *trustworthy* dealer D to a set of participants $\mathcal{P}$ 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, $t-1$ elements of $GF(p)$, $a_1, a_2, \ldots, a_{t-1}$.
(3) For $1 \le l \le n$, D computes $y_l = f(x_l)$, where $f(x) = K + \sum_{1 \le j \le t-1} a_j x^j \in GF(p)[x]$.
(4) For $1 \le l \le n$, D 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 reconstruct the secret K. If $|B| \ge t$, then they should be able to reconstruct the value of K by Lagrange interpolation; if $|B| < t$, then they should not be able to obtain any information about K. The parameter t is called the *threshold* of this scheme.

The Shamir (t, n)-threshold scheme is very simple and efficient when sharing a secret. Unfortunately, such a scheme is not secure against cheaters. Dishonest participants could submit fake shares in the process of reconstructing the secret so that the honest participants cannot obtain the proper secret. McEliece and Sarwate [6], Tompa and Woll [12], Tso, Miao and Okamoto [13] and others investigated this problem and proposed some methods to defense dishonest participants.

Another weakness of the Shamir (t, n)-threshold scheme is that it is not secure against any dishonest dealer either. In such a scheme, D should distribute $y_l = f(x_l)$ to P_l, $1 \le l \le n$, respectively, according to $f(x)$ of degree at most $t-1$. However, if the dishonest dealer D chose a polynomial of degree greater than

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

$t-1$, then different t-subsets of participants would compute different secrets from the shares they collectively hold. On the other hand, although the Shamir (t, n)-threshold scheme only requires that the degree of $f(x)$ to be *at most* $t-1$, it is usually necessary that the degree of $f(x)$ to be *exactly* $t-1$, so that no conspiracies of less than t participants can determine the secret. The recognition of such a requirement came when NIST tried to introduce the controversial Clipper Chip [7] with key escrowing to achieve legal wiretapping. The proposed escrowed encryption algorithm used two parties called Key Escrow Agencies to deposit the valid cryptographic key. Only if the two parties pooled their partial keys together, could ciphertext be decrypted. The case described at the beginning of this section is also a good example to explain the importance of an unmistaken threshold.

Verifiable secret sharing schemes were first proposed in Ref. 3) to overcome the problem of dishonest dealers. In a verifiable secret sharing scheme, the shareholders can verify the validity of their shares, and thus they can reconstruct the secret properly. Although some of these works (e.g., Refs. 4) and 9)) also provide as a byproduction the ability to verify the threshold t, the security of these methods are based on some mathematical problems such as the intractability of the discrete logarithm problem. As an undesirable consequence, their methods enlarge the size of shares as well as the secret comparing to those of the original Shamir (t, n)-threshold scheme. We observe that it is not difficult to establish threshold-verification methods based on some cryptographic assumptions but such methods require more computational resources than those based on unconditional security, and the difference between these two is becoming more and more striking because of the rapid progress of computer industry, improvement of computing speed and the appearances of faster and faster algorithms. Therefore, in this paper, we only investigate methods of verifying the threshold t of the Shamir (t, n)-threshold scheme from the viewpoint of unconditional security.

Benaloh [1] first described an unconditionally secure method to verify the degree of the chosen polynomial based on which the dealer D claimed to distribute the shares in the Shamir (t, n)-threshold scheme. However, Benaloh's method can only guarantee, with an overwhelming probability, that the polynomial is of degree *at most* $t-1$. Laih, Harn and Chang asked, in Chapter 13 of their book [5], whether there is an effective way to verify if $t-1$ is *exactly* the degree of $f(x)$ as being claimed by the dealer D.

In this paper, we assume that the dealer D is asked to construct a polynomial $f(x) \in GF(p)[x]$ of degree exactly $t-1$ in the Shamir (t, n)-threshold scheme. This may increase the probability of successfully guessing the secret $K \in GF(p)$ from $1/p$ to $1/(p-1)$ when $t-1$ participants form a conspiracy, since they know that the coefficient of the x^{t-1}-th term of $f(x) \in GF(p)[x]$ is not zero. They may construct a polynomial $f'(x) \in GF(p)[x]$ of degree at most $t-2$ using the shares they collectively hold by Lagrange interpolation and then they would know that $f'(0) \in GF(p)$ is not the possible value of the secret $K \in GF(p)$. The purpose of this assumption is that we want the scheme to be *t-consistent* but not $(t-1)$-*consistent*. In addition, similar to many other interactive methods, in our methods, we assume that every participant is honest; they may be curious but they do not cheat, which means that they may form a conspiracy of less than t participants and try to find the secret or other participants' shares but they follow the threshold-verification protocol and do not lie about their shares. The only one who may be dishonest is supposed to be the dealer. Based on these assumptions, we propose two simple but effective methods to verify the threshold t of the Shamir (t, n)-thresold scheme. The security of these methods does not depend on any unproven assumptions about the complexity of computing certain number-theoretic functions and also these methods solve the abovementioned Laih, Harn and Chang's problem [5].

The rest of this paper is organized as follows. In Section 2, we revisit Benaloh's method for verifying the threshold in the Shamir (t, n)-threshold scheme. Section 3 describes our first method modified from Benaloh's method. Section 4 describes our second method and show how it can overcome the shortcoming of the first method. Section 5 is the conclusion of the paper.

2. Benaloh's Method Revisited

We start this section with a brief description of some terminologies.

A set of shares in a (t, n)-threshold scheme is called *t-consistent* if every t-subset of the shares derives the same secret. The purpose of *verifiable secret sharing* is to convince shareholders that their shares (collectively) are t-consistent. It is easy to see that in the Shamir (t, n)-threshold scheme, the shares $y_1, y_2, \ldots, y_n$ are t-consistent if and only if the interpolation of the points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ yields a polynomial in $GF(p)[x]$ of degree at most $t-1$.

The following two lemmas are obvious but useful.

Lemma 2.1 Let p be a prime, and $f(x), g(x) \in GF(p)[x]$. Then $\deg(f+g) \leq \max\{\deg(f), \deg(g)\}$.

Lemma 2.2 Let p be a prime. If the sum of two polynomials in $GF(p)[x]$ is of degree at most d, then either both polynomials are of degree at most d or both polynomials are of degree greater than d.

Let $t-1$ be the degree of $f(x) \in GF(p)[x]$ which was used by the dealer D to distribute shares in the Shamir (t, n)-threshold scheme. Benaloh's proof that $f(x)$ is of degree at most $t-1$ can essentially be outlined as follows.

Benaloh's method

(1') D computes the shares $y_l = f(x_l)$, $1 \leq l \leq n$, and then distributes y_l to participant P_l for $1 \leq l \leq n$ through a private secure channel.

(2') D selects many, say N, additional random polynomials $f_s(x) \in GF(p)[x]$, $1 \leq s \leq N$, of degree at most $t-1$, and then distributes the corresponding shares $f_s(x_l)$ to P_l for $1 \leq l \leq n$ through a private secure channel.

(3') All participants designate a random $\lceil N/2 \rceil$-subset R of $f_s(x)$, $1 \leq s \leq N$, say, $R = \{f_{i_1}(x), f_{i_2}(x), \ldots, f_{i_{\lceil N/2 \rceil}}(x) : i_j \in \{1, 2, \ldots, N\}, 1 \leq j \leq \lceil N/2 \rceil\}$.

(4') All participants pool their shares $f_{i_j}(x_l)$, $1 \leq j \leq \lceil N/2 \rceil$, to recover $f_{i_j}(x) \in GF(p)[x]$ by Largrange interpolation. These polynomials $f_{i_j}(x)$ must all be of degree at most $t-1$.

(5') These participants pool $f_s(x_l) + f(x_l) \in GF(p)$ for all the remaining random polynomials $f_s(x)$ to reconstruct the polynomials $f_s(x)+f(x)$ by Largrange interpolation. These polynomials $f_s(x)+f(x)$ must all be of degree at most $t-1$.

For a large positive integer N, if $f_{i_j}(x)$, $1 \leq j \leq \lceil N/2 \rceil$, are all of degree at most $t-1$, and if furthermore $f_s(x) + f(x)$ are all of degree at most $t-1$ for all the remaining $N - \lceil N/2 \rceil$ random polynomials $f_s(x)$, then we are almost sure that, by Lemma 2.2, the degree of $f(x)$ is upper bounded by $t-1$. Also note that each participant has N additional shares of the same size, and that although $f_s(x_l) + f(x_l)$ were pooled, the shares $f(x_l)$ are still kept secret.

According to our assumption on the Shamir (t, n)-threshold scheme, $f(x)$ must be exactly of degree $t-1$. Unfortunately, Benaloh's protocol can only convince the participants that $f(x)$ is of degree *at most* $t-1$. In the next section, we will describe a simple method to modify Benaloh's protocol to convince the participants that $f(x)$ is of degree *at least* $t-1$.

3. Proposed Method 1

In the Shamir (t, n)-threshold scheme, Benaloh's protocol, as we complained in the last section, only allows participants to verify whether the degree of $f(x)$ is at most $t-1$ or not. In this section, we describe a simple modification of Benaloh's protocol which can be used to convince the participants, to some extent, that the degree of $f(x)$ used in the Shamir (t, n)-threshold scheme is exactly of degree $t-1$.

In step (5') of Benaloh's protocol, if the polynomials $f_s(x)+f(x)$ are of degree at most $t-1$, then they can only ensure to participants that the degree of $f(x)$ is upper bounded by $t-1$ since the degrees of the random polynomials $f_s(x)$ are at most $t-1$. If all these random polynomials $f_s(x)$, $1 \leq s \leq N$, were asked to be exactly of degree $t-1$ in step (2') of Benaloh's protocol, and if the coefficients of the x^{t-1}-th term of the random polynomials $f_s(x)$ could be verified to be different from those of $f_s(x)+f(x)$ (without revealing any other information about $f_s(x)$ and $f(x)$, of course,) then participants can be convinced that $f(x)$ is exactly of degree $t-1$, as the dealer D previously claimed. The essentials of our method are first to require the N additional random polynomials $f_s(x)$, $1 \leq s \leq N$, to be of degree exactly $t-1$ in step (2') of Benaloh's protocol, and then, in addition, to require the coefficients of the x^{t-1}-th term of these polynomials to be all primes so that participants can verify if the coefficients of the x^{t-1}-th term have been changed to composites or not after the computation of $f_s(x) + f(x)$.

In step (3') of Benaloh's protocol, participants designate a random $(N-1)$-subset of

the random polynomials $f_s(x)$, and in step (4'), participants check these $(N-1)$ random polynomials $f_s(x)$ to see if these polynomials are all of degree exactly $t-1$ and the coefficients of the x^{t-1}-th term are all primes. In step (5'), participants pool the sums of their shares $f_{s'}(x_l)+f(x_l)$ to recover $g_{s'}(x)=f_{s'}(x)+f(x)$ where $f_{s'}(x)$ is the remaining polynomial not been checked by participants. With the additional property that the coefficient of the x^{t-1}-th term of $f_{s'}(x)$ is a prime, if $g_{s'}(x)$ is of degree less than $t-1$, or of degree $t-1$ but its coefficients of the x^{t-1}-th term not a prime, then participants can assert, with a probability $(N-1)/N$, that $f(x)$ is of degree exactly $t-1$. This probability is computed from the fact that the dealer is able to cheat the verification by guessing the random polynomial $f_{s'}(x)$ that is not tested in step (4') with a probability $1/N$. Problems arise when the degree of $g_{s'}(x)$ is $t-1$ but the coefficient of its x^{t-1}-th term is still a prime. We prove that this problem will only arise with a probability approximately equal to $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$.

Lemma 3.1 Let p be a prime, $\mathcal{A}=GF(p)\backslash\{0\}$, and $\mathcal{B}=\{b:\ 1\le b\le p-1,\ b$ is a prime$\}$. If $a\in\mathcal{A}$ and $b\in\mathcal{B}$ are chosen randomly, then the probability that $a+b \bmod p$ is a prime is approximately equal to $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$.

Proof According to Prime Number Theorem, the number of primes not exceeding a positive integer m is approximately $m/\ln m$, so $|\mathcal{B}|\approx(p-1)/\ln(p-1)$. We consider the set $H=\{(a,b)\in\mathcal{A}\times\mathcal{B}:\ a+b \bmod p$ is a prime$\}$. For any prime g, $1\le g\le p-1$, let $H_g=\{(a,b)\in\mathcal{A}\times\mathcal{B}:\ g=a+b \bmod p\}$. Then clearly H can be partitioned into H_g, where g, $1\le g\le p-1$, are primes. For any prime g, $1\le g\le p-1$, $|H_g|=|\mathcal{B}|-1\approx\frac{p-1}{\ln(p-1)}-1$ since b is not allowed to be g (because a is not allowed to be 0). Therefore $|H|\approx\frac{p-1}{\ln(p-1)}\times(\frac{p-1}{\ln(p-1)}-1)$. Meanwhile, $|\mathcal{A}\times\mathcal{B}|\approx(p-1)\times\frac{p-1}{\ln(p-1)}$. Hence the probability for $a+b \bmod p$, where a and b are randomly chosen from $\mathcal{A}$ and $\mathcal{B}$, to be a prime is approximately equal to

$$\frac{\frac{p-1}{\ln(p-1)}\times(\frac{p-1}{\ln(p-1)}-1)}{(p-1)\times\frac{(p-1)}{\ln(p-1)}}=\frac{1}{\ln(p-1)}-\frac{1}{p-1}.$$

∎

Therefore, there is an average probability $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$ that the coefficient of the x^{t-1}-th term of $g_{s'}(x)=f_{s'}(x)+f(x)$ is still a prime when $f(x)$ used in the Shamir (t,n)-threshold scheme is of degree $t-1$. From the fact that the average probability $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$ tends to 0 as p becomes very large, the participants can ignore this average probability and conclude that the degree of $f(x)$ is less than $t-1$ if the degree of $g_{s'}(x)$ is $t-1$ but its leading coefficient is still a prime. Also note that though the dealer D can, but he will not, choose a_{t-1}, the leading coefficient of $f(x)$, intentionally to make the coefficient of the x^{t-1}-th term of $g_{s'}(x)$ to be a prime, because his purpose is to convince the participants that he is honest about the threshold t of the Shamir (t,n)-threshold scheme.

With the pre-knowledge that the threshold is less than $t+1$, the threshold-verification problem can be viewed as the following decision problem: "Is the threshold of this scheme exactly t?" The answer *Yes* will mean that the threshold of this scheme is exactly t and the answer *No* will mean that the threshold of this scheme is less than t. As a consequence, if we lay the dealer's cheating aside, then our method can be regarded as a *yes-biased* probabilistic method with an error probability nearly equal to $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$, where *Yes* is always correct but *No* may be incorrect with an average probability $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$.

If the answer is *No*, the participants should reject this scheme although it may be incorrect with a probability nearly equal to $\frac{1}{\ln(p-1)}-\frac{1}{p-1}$. The participants should ask the dealer D to recreate a new Shamir (t,n)-threshold scheme and to distribute new shares to participants through a private secure channel. The participants should verify the threshold of this new Shamir (t,n)-threshold scheme again according to this slight modification of Benaloh's protocol.

The approach to consider the security of this slightly modified method is similar to that of Benaloh's one. Although the sums of shares $f_{s'}(x_l)+f(x_l)$ are pooled and $g_{s'}(x)=f_{s'}(x)+f(x)$ is reconstructed, the shares $f(x_l)$ and the secret K are still kept secret. Besides that, this modified method still preserves perfect secrecy against any conspiracy of less than $t-1$ participants, even if they had participated in the threshold-verification process. In this case, they can successfully guess a_{t-1} as $g_{s',t-1}-p_i\in GF(p)$ with an average probability $\frac{\ln(p-1)}{p-1}$, where a_{t-1}, $g_{s',t-1}\in GF(p)$ are

the coefficients of the x^{t-1}-th term of $f(x)$, $g_{s'}(x) \in GF(p)[x]$ respectively and p_i is any prime not exceeding p. However, according to the property of the Shamir (t, n)-threshold scheme, no information concerning the secret $K \in GF(p)$ and shares belonging to other participants will be leaked out in the case of less than $t-1$ shares and the value a_{t-1} having been revealed. That is, a conspiracy of less than $t-1$ participants can not obtain any information about the secret K and other participants' shares from a_{t-1} and $g_{s'}(x)$. But this method does not provide perfect secrecy against a conspiracy of exactly $t-1$ participants, because the approximate $\frac{p-1}{\ln(p-1)}$ possible values of $a_{t-1} \in GF(p)$ will give them enough information to solve the system of $t-1$ linear equations in the $t-1$ unknowns $K, a_1, \ldots, a_{t-2}$, $y_{i_j} = K + a_1 x_{i_j} + \cdots + a_{t-2} x_{i_j}^{t-2} + a_{t-1} x_{i_j}^{t-1}$, $1 \le j \le t-1$, where all arithmetic is done in $GF(p)$, and thus reduce the number of possible values of the secret K from $p-1$ (c.f. Section 1) to approximately $\frac{p-1}{\ln(p-1)}$ accordingly. As an immediate consequence, the probability for any conspiracy of exactly $t-1$ participants to find the secret K will be increased approximately from $\frac{1}{p-1}$ to $\frac{\ln(p-1)}{p-1}$.

This modified method is simple, and any Shamir (t, n)-threshold scheme with an incorrect threshold t will be rejected with a high probability by the participants. Unfortunately, this method is not very efficient. Each participant receives N additional shares of the same size in each verification round. We also mentioned that the dealer can cheat the verification by guessing the random polynomial $f_{s'}(x)$ that is not tested in step (4') with a probability $1/N$. In order to make this probability negligible, we should choose a very large N. Consequently, we have to generate a very large number of additional random polynomials each of degree exactly $t-1$ and each with a prime leading coefficient, which in turn requires a primality test in generating such a polynomial.

We know that with an error probability approximately equal to $\frac{1}{\ln(p-1)} - \frac{1}{p-1}$, the Shamir (t, n)-threshold scheme will be rejected by the participants even though the dealer D is honest about threshold t. Also we know that the number of possible values of the secret $K \in GF(p)$ will be reduced from $p-1$ to approximately $\frac{p-1}{\ln(p-1)}$ for any conspiracy of exactly $t-1$ participants. As a consequence, for security purpose, the prime p should also be very large. **Table 1** shows the values of $\frac{1}{\ln(p-1)} - \frac{1}{p-1}$ and $\frac{p-1}{\ln(p-1)}$ for several known Mersenne primes p, where $\frac{1}{\ln(p-1)} - \frac{1}{p-1}$ is the approximate error probability and $\frac{p-1}{\ln(p-1)}$ is the number of possible values of the secret $K \in GF(p)$ against any conspiracy of exactly $t-1$ participants.

Table 1 Error probability and possible values of K.

p	$\frac{1}{\ln(p-1)} - \frac{1}{p-1}$	$\frac{p-1}{\ln(p-1)}$
$2^{13}-1$	1.107×10^{-1}	909
$2^{61}-1$	2.365×10^{-2}	2^{56}
$2^{107}-1$	1.348×10^{-2}	2^{101}
$2^{521}-1$	2.769×10^{-3}	2^{512}
$2^{607}-1$	2.376×10^{-3}	2^{598}
$2^{1297}-1$	1.112×10^{-3}	2^{1287}
$2^{2203}-1$	6.551×10^{-4}	2^{2193}

We emphasize here that although the error probability illustrated in Table 1 seems to be non-negligible, this causes no serious problems to participants since they should reject the scheme in this case. Also we can see from Table 1 that for a large prime p, the number of possible values of the secret $K \in GF(p)$ against any conspiracy of exactly $t-1$ participants is very close to the number of all possible values of K, so this will cause no serious problems too for any conspiracy of exactly $t-1$ participants.

4. Proposed Method 2

In this section, we describe our second method which can overcome the shortcomings of our first method. This method requires the threshold t to be greater than 2.

Since Benaloh's method described in Section 2 provides the ability to convince the participants that the polynomial $f(x)$ used in the Shamir (t, n)-threshold scheme is of degree *at most* $t-1$, in this section, we pay our attention only to the problem of convincing the participants that the degree of $f(x)$ is of degree *at least* $t-1$. Combining this method to Benaloh's one, participants can succeed in verifying the exact threshold of the scheme.

Note that if $f(x)$ used in the Shamir (t, n)-threshold scheme is of degree exactly $t-1$, then the shares are necessarily not $(t-1)$-consistent. Therefore, if two groups of $t-1$ participants each pool their shares to reconstruct the polynomials $f'(x)$ and $f''(x)$ respectively, then with a high probability, $f'(x)$ and $f''(x)$ should not be the same. This is the main fact we will use in this section. The point is that this verification

should be done without revealing any information about the secret K as well as the shares of participants. We illustrate our method first from the following example.

If $2t-2$ participants, say $P_1, \ldots, P_{2t-2}$, agree to verify whether the threshold of the scheme is greater than $t-2$, then they first randomly divide them into two groups of $t-1$ participants each, say $\mathcal{T}_1 = \{P_1, \ldots, P_{t-1}\}$ and $\mathcal{T}_2 = \{P_t, \ldots, P_{2t-2}\}$. If each P_i possesses a secret random number $r_i \in GF(p)$, $1 \le i \le 2t-2$, such that $\sum_{i=1}^{2t-2} r_i = 0 \bmod p$, then each $P_i \in \mathcal{T}_1$ computes $y_i \cdot \prod_{\substack{1 \le l \le t-1 \\ l \ne i}} (x_i - x_l)^{-1} + r_i \bmod p$ and each $P_i \in \mathcal{T}_2$ computes $(-y_i) \cdot \prod_{\substack{t \le l \le 2t-2 \\ l \ne i}} (x_i - x_l)^{-1} + r_i \bmod p$ respectively. These participants pool these values and sum them up.

$$
\begin{aligned}
&\sum_{i=1}^{t-1} \Big(y_i \cdot \prod_{\substack{1 \le l \le t-1 \\ l \ne i}} (x_i - x_l)^{-1} + r_i \bmod p\Big) + \\
&\sum_{i=t}^{2t-2} \Big((-y_i) \cdot \prod_{\substack{t \le l \le 2t-2 \\ l \ne i}} (x_i - x_l)^{-1} + r_i \bmod p\Big) \\
&\quad = \sum_{i=1}^{t-1} \Big(y_i \cdot \prod_{\substack{1 \le l \le t-1 \\ l \ne i}} (x_i - x_l)^{-1}\Big) - \sum_{i=t}^{2t-2} \Big(y_i \cdot \\
&\quad \prod_{\substack{t \le l \le 2t-2 \\ l \ne i}} (x_i - x_l)^{-1}\Big) + \sum_{i=1}^{2t-2} r_i \bmod p \\
&\quad = \sum_{i=1}^{t-1} \Big(y_i \cdot \prod_{\substack{1 \le l \le t-1 \\ l \ne i}} (x_i - x_l)^{-1}\Big) - \sum_{i=t}^{2t-2} \Big(y_i \cdot \\
&\quad \prod_{\substack{t \le l \le 2t-2 \\ l \ne i}} (x_i - x_l)^{-1}\Big) \bmod p \\
&\quad = a'_{t-2} - a''_{t-2} \bmod p, \qquad (1)
\end{aligned}
$$

where $a'_{t-2} \in GF(p)$ is the coefficient of the x^{t-2}-th term of $f'(x)$, which can be uniquely computed by $\mathcal{T}_1$ by Lagrange interpolation, and $a''_{t-2} \in GF(p)$ is the coefficient of the x^{t-2}-th term of $f''(x)$, which can be uniquely computed by $\mathcal{T}_2$ by Lagrange interpolation. If $a'_{t-2} - a''_{t-2} \ne 0 \bmod p$, then $f'(x) \ne f''(x)$, which means that $f(x)$ used in the scheme cannot be uniquely determined by these two groups $\mathcal{T}_1$ and $\mathcal{T}_2$ of $t-1$ participants each. Therefore, the degree $f(x)$ should be greater than $t-2$. On the other hand, if $f'(x) \ne f''(x)$, then $a'_{t-2} - a''_{t-2} = 0 \bmod p$ can only happen with a probability $\frac{p^{t-2}-1}{p^{t-1}-1}$, which is computed as follows.

There are totally p^{t-1} numbers of polynomials of degree less than or equal to $t-2$. Since $f'(x) \ne f''(x)$, the total number of possibilities of the pair $(f'(x), f''(x))$ is $\binom{p^{t-1}}{2}$. If, in addition, the coefficients of the x^{t-2}-th term, a'_{t-2} of $f'(x)$ and a''_{t-2} of $f''(x)$, are the same, then the total number of possibilities of the pair $(f'(x), f''(x))$ becomes $p \times \binom{p^{t-2}}{2}$. Therefore, the probability that $a'_{t-2} - a''_{t-2} = 0 \bmod p$ under the condition of $f'(x) \ne f''(x)$ is $p \times \binom{p^{t-2}}{2} / \binom{p^{t-1}}{2} = \frac{p^{t-2}-1}{p^{t-1}-1}$.

When p is large enough, participants can ignore this probability. In other words, if p is large enough, then $f'(x) \ne f''(x)$ will almost imply $a'_{t-2} - a''_{t-2} \ne 0 \bmod p$. Or conversely, when p is large enough, $a'_{t-2} - a''_{t-2} = 0 \bmod p$ will almost imply $f'(x) = f''(x)$, which means that the dealer D is dishonest about the threshold and the degree of $f(x)$ is less than $t-1$.

A simple explanation to the security of this method is that because every secret share y_i is concealed by the random number r_i and $\sum_{i=1}^{2t-2} r_i = 0 \bmod p$, any conspiracy of less than $2t-3$ participants can obtain no information about other participants' shares y_i not in the conspiracy from the Eq. (1). Also note that the threshold to be verified is t, so the number of participants engaged in threshold-verification can be reduced to any number q with $t+1 \le q \le 2t-2$, where in this case, $|\mathcal{T}_1| = |\mathcal{T}_2| = t-1$ and $|\mathcal{T}_1 \cap \mathcal{T}_2| = 2t-2-q$. The case $q = t$ is not allowed, since otherwise the conspiracy of the $t-2$ participants in $\mathcal{T}_1 \cap \mathcal{T}_2$ would be able to compute the secret by using the two shares pooled by the two participants outside of $\mathcal{T}_1 \cap \mathcal{T}_2$.

Now we describe our method in the following protocol. In this protocol, the number of participants engaged in threshold-verification is q, where $t+1 \le q \le 2t-2$. Without loss of generality, we assume these participants to be $P_1, \ldots, P_q$, $\mathcal{T}_1 = \{P_l : 1 \le l \le t-1\}$, $\mathcal{T}_2 = \{P_l : q-t+2 \le l \le q\}$. Consequently, $\mathcal{T}_1 \cap \mathcal{T}_2 = \{P_l : q-t+2 \le l \le t-1\}$. In addition, similar to Benaloh's original method, the dealer D should choose many, say N', sets $R_1, \ldots, R_{N'}$ of random numbers in $GF(p)$, $R_i = \{r_{i,1}, \ldots, r_{i,q}\}$, $1 \le i \le N'$, such that $\sum_{j=1}^{q} r_{i,j} = 0 \bmod p$.

Protocol

(1"')D computes the shares $y_l = f(x_l)$, $1 \le l \le n$, and then distributes y_l to participant P_l for $1 \le l \le n$ through a private secure channel.

(2"')When participants $P_1, \ldots, P_q$ agree to verify the threshold of the scheme, D selects many, say N', sets $R_1, \ldots, R_{N'}$ of non-zero random numbers in $GF(p)$, $R_i = \{r_{i,1}, \ldots, r_{i,q}\}$, $1 \le i \le N'$, such that $\sum_{j=1}^{q} r_{i,j} = 0 \bmod p$. D distributes $r_{i,l}$ for $1 \le i \le N'$ to participant P_l for $1 \le l \le q$ through a private secure channel.

(3"')Participants P_l, $1 \le l \le q$, designate a random $(N'-1)$-subset of $\mathcal{R} = \{R_1, \ldots, R_{N'}\}$, say, R_{i_j}, $1 \le j \le N'-1$, pool their random numbers $r_{i_j,l} \in R_{i_j}$, $1 \le j \le N'-1$, and verify if $\sum_{l=1}^{q} r_{i_j,l} = 0 \bmod p$. If these congruences are verified to be true for all j, $1 \le j \le N'-1$, then $\sum_{l=1}^{q} r_{i_{N'},l} = 0 \bmod p$ can also be considered true.

(4"')Participants P_l, $P_l \in \mathcal{T}_1 \setminus (\mathcal{T}_1 \cap \mathcal{T}_2)$, compute $y_l \cdot \prod_{\substack{1 \le i \le t-1 \\ i \ne l}} (x_l - x_i)^{-1} + r_{i_{N'},l} \bmod p$, participants P_l, $P_l \in \mathcal{T}_2 \setminus (\mathcal{T}_1 \cap \mathcal{T}_2)$, compute $(-y_l) \cdot \prod_{\substack{q-t+2 \le i \le q \\ i \ne l}} (x_l - x_i)^{-1} + r_{i_{N'},l} \bmod p$, and participants P_l, $P_l \in \mathcal{T}_1 \cap \mathcal{T}_2$, compute $y_l \cdot (\prod_{\substack{1 \le i \le t-1 \\ i \ne l}} (x_l - x_i)^{-1} - \prod_{\substack{q-t+2 \le i \le q \\ i \ne l}} (x_l - x_i)^{-1}) + r_{i_{N'},l} \bmod p$, secretly and individually. They pool all these results and compute S_j as their sum.

$$S = \sum_{l=1}^{t-1} y_l \cdot \prod_{\substack{1 \le i \le t-1 \\ i \ne l}} (x_l - x_i)^{-1} + \sum_{l=q-t+2}^{q} (-y_l) \cdot \prod_{\substack{q-t+2 \le i \le q \\ i \ne l}} (x_l - x_i)^{-1} + \sum_{l=1}^{q} r_{i_{N'},l} \bmod p.$$

(5"')If $S \ne 0$, then the degree of $f(x)$ must be greater than $t-2$.

(6"')If $S = 0$, then the degree of $f(x)$ must be less than or equal to $t-2$ with an error probability $\frac{p^{t-2}-1}{p^{t-1}-1}$.

In step (3"'), a random $(N'-1)$-subset of $\mathcal{R}$ is tested. This allows the dealer D to cheat the verification by guessing the random mask that is not tested in step (3"') with a probability $1/N'$. If N' becomes very large, then this probability tends to 0. There is a trick that makes the dealer's chance of cheating even more smaller. In step (3"'), participants randomly choose an $\lceil N'/2 \rceil$-subset of $\mathcal{R}$ to challenge. In step (4"'), participants compute the values of S for all the unchallenged cases. The test is successful only if all the values of S are not 0. To cheat the dealer D has to be able to guess exactly which subset participants will challenge; the odds are much smaller than the odds of guessing which one set R_i of random numbers participants will not challenge. In fact the probability is $1/\binom{N'}{\lceil N'/2 \rceil}$ in this case, which is obviously much smaller than $1/N'$. The idea described above is exactly the same as that used in the blind signatures depicted in Page 114 of[10] to prevent the holder of the message to be signed blindly from cheating. However, in this paper, we do not describe this better method explicitly for the convenience of explanation, although everything works correctly with this method.

We look at this protocol in more details. If $S = 0$, then participants are unable to correctly determine the threshold, since even if D is honest, $S = 0$ will happen with a probability $\frac{p^{t-2}-1}{p^{t-1}-1}$. However, if the prime p is large enough, then this probability can be reduced to zero and thus be ignored. Another way to reduce this probability is to replace at least one participant in $\mathcal{T}_1 \setminus \mathcal{T}_1 \cap \mathcal{T}_2$ and at least one in $\mathcal{T}_2 \setminus \mathcal{T}_1 \cap \mathcal{T}_2$ by new participants respectively, or just to replace at least one participant in $\mathcal{T}_1 \cap \mathcal{T}_2$ by new participants to carry out this protocol from step (2"') again. If the degree of $f(x)$ is less then or equal to $t-2$, then S_j in step (4"') must still be zero. Conversely, if the degree of $f(x)$ is great than $t-2$, then $S = 0$ will only occur again with a probability $(\frac{p^{t-2}-1}{p^{t-1}-1})^2$. Repeating this protocol m times, the error probability will be reduced to $(\frac{p^{t-2}-1}{p^{t-1}-1})^m$. As a consequence, participants are able to verify, with an overwhelming probability, whether the degree of $f(x)$ is exactly $t-1$ as being claimed by the dealer D.

Now we consider the security of this protocol in detail. It is clear that no one can obtain any information about the secret $K \in GF(p)$ and other participants' shares from the messages pooled in step (4"') because each of the shares $y_l \in GF(p)$ is concealed by the private random number $r_{i_{N'},l}$. We notice that any individual participant can obtain a linear equation with $q-1$ or q unknowns from S in step (4"') according to whether he takes part in the threshold-verification or not, where the unknowns are the

shares belonging to other participants engaged in the threshold-verification. But this equation is not helpful to him for increasing the probability of finding the secret $K \in GF(p)$ and the shares of other participants, because $q-1$ or q unknowns with $t+1 \leq q \leq 2t-2$ can not be uniquely determined from one linear equation. When a set $\mathcal{C}$ of $t-1$ participants, where $\mathcal{C} = \mathcal{T}_1$ or $\mathcal{T}_2$, form a conspiracy, they are able to obtain the values $\sum_{l=1}^{t-1} y_l \cdot \prod_{\substack{1 \leq i \leq t-1 \\ i \neq l}} (x_l - x_i)^{-1}$ and $\sum_{l=q-t+2}^{q} (-y_l) \cdot \prod_{\substack{q-t+2 \leq i \leq q \\ i \neq l}} (x_l - x_i)^{-1}$ from S in step (4"'), where the first term is in fact the value of a'_{t-2}, the coefficient of the x^{t-2}-th term of $f'(x)$ computed by participants $P_1, \ldots, P_{t-1}$, and the second term is in fact the value of a''_{t-2}, the coefficient of the x^{t-2}-th term of the polynomial $f''(x)$ computed by participants $P_{q-t+2}, \ldots, P_q$. But these values still give them no help to increase the probability of finding the secret $K \in GF(p)$ and the shares belonging to other participants not in their conspiracy. What they can obtain from a''_{t-2} (if $\mathcal{C} = \mathcal{T}_1$) or a'_{t-1} (if $\mathcal{C} = \mathcal{T}_2$) or from S directly is a linear equation with $q-t+1$ unknowns where these unknowns can not be uniquely determined because $q \geq t+1$. Therefore, the probability for the $t-1$ participants in a conspiracy to find the secret $K \in GF(p)$ in this case remains $1/(p-1)$ (c.f. Section 1). The security in the case $\mathcal{C} \subset \mathcal{T}_1 \cup \mathcal{T}_2$ with $\mathcal{C} \cap \mathcal{T}_1 \neq \emptyset$ and $\mathcal{C} \cap \mathcal{T}_2 \neq \emptyset$, or the case $\mathcal{C} \not\subset \mathcal{T}_1 \cup \mathcal{T}_2$ with $|\mathcal{C}| \leq t-1$ can be also analyzed in the same way. Therefore, no conspiracy formed by less than t participants can obtain any information about the secret $K \in GF(p)$ and the shares of other participants not in their conspiracy. If the process of the threshold-verification is repeated m times, since at least one participant in each of the two groups needs to be replaced by a new participant each time, the number of participants took part in the threshold-verification will be *at least* $q+m-1$, while any conspiracy of less than t participants can obtain *at most* m linear equations, one from each round, with totally z unknowns, where $z \geq q+m-1-(t-1) \geq m+l$. Since at least $m+l$ unknowns can not be uniquely determined by at most m equations, any conspiracy less than t participants still cannot rule out any possible values of the secret $K \in GF(p)$ and shares y_i of participants P_i not in their conspiracy.

Similar to the last section, with the preknowledge that the threshold is less than $t+1$, threshold-verification problem can be viewed as the following decision problem: "Is the threshold of this scheme exactly t?" The answer *Yes* means again that the threshold of this scheme is exactly t, and *No* means the threshold of this scheme is less than t. This method is again a *yes-biased* probabilistic method with an error probability $\frac{p^{t-2}-1}{p^{t-1}-1}$, if we lay the dealer's cheating aside. An advantage of this method over our first method is that if in both $\mathcal{T}_1$ and $\mathcal{T}_2$, at least one participant is replaced by a new participant not belonging to $\mathcal{T}_1 \cup \mathcal{T}_2$, or just replace at least one participant from $\mathcal{T}_1 \cap \mathcal{T}_2$, then the participants can go through the same verification process once again. In the case when D is honest about the threshold, the probability that the verification process will return *No* m times in succession will be reduced to less than or equal to $(\frac{p^{t-2}-1}{p^{t-1}-1})^m$.

We emphasize again that this method is only applicable to the Shamir (t, n)-threshold scheme with $t > 2$ because it is necessary that $2t-2 \geq t+1$.

5. Concluding Remarks

In this paper, we proposed two unconditionally secure methods to verify the threshold t of the Shamir (t, n)-threshold scheme. Our first method is a simple modification of Benaloh's method [1]. This method is a *yes-biased* probabilistic method, and returns *No* answer with an error probability approximately equal to $\frac{1}{\ln(p-1)} - \frac{1}{p-1}$, where *Yes* answer means the threshold is exactly t and *No* answer means the threshold is *not* greater than $t-1$, with the preknowledge that the threshold is less than $t+1$. Among others, a shortcoming of this method is that the number of possible values of the secret $K \in GF(p)$ will be reduced from $p-1$ to approximately $\frac{p-1}{\ln(p-1)}$ against a conspiracy of $t-1$ participants. Our second method, also a yes-biased probabilistic method returning *No* answer with an error probability less than or equal to $\frac{p^{t-2}-1}{p^{t-1}-1}$, can overcome the shortcomings of the first method. The second method allows any m-subset of participants with $m > t$ to verify if the threshold is at least t or not, but requires the threshold t to be greater than 2. Its error probability can be reduced to $(\frac{p^{t-2}-1}{p^{t-1}-1})^m$ by repeating the verification procedure m times under some mild requirements. Combining this method with Benaloh's one, participants can verify if

the threshold is exactly t or not. These new proposed methods solve Laih, Harn and Chang's problem in their book [5], although they are not very efficient for the reasons that they require many additional shares to be distributed and many verification processes to be executed.

We also note that there are some close relations between threshold-verification and shares-verification. If the dealer gives a faulty share to one participant, then it will cause the scheme to be *non* t-consistent and consequently cause the polynomial participants computed to be of degree greater than $t-1$. On the other hand, if the polynomial $f(x)$ the dealer used in the Shamir (t,n)-threshold scheme is verified to be of degree greater than $t-1$, then participants can conclude that either the threshold is greater than t, in which all shares can be viewed as faulty shares, or that at least one faulty share has been distributed by the dealer. Another fact we would like to emphasize is that even if no faulty shares are detected by share-verification protocols, it can only guarantee that the Shamir (t,n)-threshold scheme is t-consistent but can not rule out the possibility of being $(t-1)$-consistent. In this sense, threshold-verification surpasses share-verification.

Acknowledgments This research is supported by Grant-in-Aid for Scientific Researches (C) under Contract Number 14540100 and (B) under Contract Number 16360184.

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) Blakley, G.: Safeguarding Cryptographic Keys, *Proc. AFIPS National Computer Conference*, pp.313–317 (1979).
3) 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 Foundations of Comput. Sci.*, pp.383–395 (1985).
4) Harn, L.: Efficient Sharing (Broadcasting) of Multiple Secrets, *IEE Proc. -Comput. Digit. Tech.*, Vol.142, pp.237–240 (1995).
5) Laih, C.S., Harn, L. and Chang, C.C.: Computer Cryptography and Applications, Taiwan (1995).
6) McEliece, R.J. and Sarwate, D.V.: On Sharing Secrets and Reed-Solomon Codes, *Comm. ACM*, Vol.24, pp.583–584 (1981).
7) National Institute of Standards and Technology (NIST): NIST FRIPS PUB, Vol.185, Escrowed Encryption Standard (1994).
8) National Institute of Standards and Technology (NIST): NIST FRIPS PUB, Vol.196, Digital Signature Standard (1994).
9) Pedersen, T.P.: Non-Interactive and Informatin Theoretic Secure Verifiable Secret Sharing, *Advances in Cryptology — CRYPTO'91*, Lecture Notes in Comput Sci., Vol.576, pp.129–140 (1991).
10) Schneier, B.: *Applied Cryptography Second Edition: Protocols, Algorithms, and Source Code in C*, John Wiley & Sons, New York (1996).
11) Shamir, A.: How to Share a Secret, *Comm. ACM*, Vol.22, pp.612–613 (1979).
12) Tompa, M. and Woll, H.: How to Share a Secret with Cheaters, *J. Cryptology*, Vol.1, pp.133–138 (1988).
13) 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 November 25, 2004)
(Accepted April 1, 2005)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.1, pp.294–303.)

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.