Determining the maximum consistent set of shares in a threshold scheme

D. R. Stinson* and R. Wei
Department of Combinatorics and Optimization
University of Waterloo
Waterloo Ontario, N2L 3G1
Canada

March 29, 1999

Abstract

The shares in a (k, n) Shamir threshold scheme consist of n points on some polynomial of degree at most $k-1$. If one or more of the shares are faulty, then the secret may not be reconstructed correctly. Supposing that at most t of the n shares are faulty, we show how a suitably chosen covering design can be used to compute the correct secret. We review known results on coverings of the desired type, and give two new constructions. We also consider a randomized algorithm for the same problem, and compare it to the deterministic algorithm obtained by using a particular class of coverings.

1 Introduction

Suppose we have a (k, n) threshold scheme, say a Shamir scheme (see, e.g., [4]) implemented in $\mathbb{F}_q$. Let

$$S = \{(x_i, y_i) : 1 \leq i \leq n\} \subseteq \mathbb{F}_q \times \mathbb{F}_q$$

be the set of n shares, and assume that at most t of the shares are faulty. In other words, there exists a polynomial $p_0(x) \in \mathbb{F}_q[x]$ 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 non-faulty shares, is the value $p_0(0)$. The problem we consider in this note is to find an efficient algorithm to compute p_0, given that some unspecified subset of t of the n shares are faulty.

Denote $\mathsf{G} = \{i : y_i = p_0(x_i)\}$ (the *good shares*) and $\mathsf{B} = \{1, \ldots, n\} \backslash \mathsf{G}$ (the *bad shares*). Then $|\mathsf{G}| = n - t$ and $|\mathsf{B}| = t$.

For any $T \subseteq \{1, \ldots, n\}$ such that $|T| = k$, there is a unique polynomial p_T of degree at most $k-1$ such that $p_T(x_i) = y_i$ for all $i \in T$. The polynomial p_T can easily be computed by Lagrange interpolation. The following two facts are obvious:

1. If $T \subseteq \mathsf{G}$, then $p_T = p_0$.

*research supported by NSERC grants IRC # 216431-96 and RGPIN # 203114-98

2. If $T \cap \mathsf{B} \neq \emptyset$, then $p_T \neq p_0$.

Now, for $T \subseteq \{1, \ldots, n\}$, $|T| = k$, define $C_T = \{i : p_T(x_i) = y_i\}$. Then it is clear that $|C_T| \geq n - t$ if $T \subseteq \mathsf{G}$. On the other hand, if $T \cap \mathsf{B} \neq \emptyset$, then $|C_T| \leq k + t - 1$, since $|C_T \cap \mathsf{B}| \leq |\mathsf{B}| \leq t$ and $|C_T \cap \mathsf{G}| \leq k - 1$.

If $n - t \leq k + t - 1$, then there could exist a polynomial $p_T \neq p_0$ of degree at most $k - 1$ such that at least $n - t$ shares lie on p_T. Therefore, in order to guarantee that our problem can be solved, it must be the case that $n - t > k + t - 1$, or $n \geq 2t + k$. We will assume that this inequality holds for the rest of this note.

In the remaining sections of this paper, we show how a suitably chosen covering design can be used to compute the correct secret. We review known results on coverings of the desired type, and give two new constructions. We also consider a randomized algorithm for the same problem, and compare it to the deterministic algorithm obtained by using a particular class of coverings.

2 An algorithm to compute the polynomial p_0

Let $\mathcal{T}$ be a set of k-subsets of $\{1, \ldots, n\}$, called *blocks*. The following algorithm will compute the polynomial p_0 if the set system $\mathcal{T}$ is chosen appropriately.

Algorithm 1

Input $\mathcal{T}$, S, n, k and t.

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

1. compute p_T
2. compute C_T
3. if $|C_T| \geq n - t$, then set $p_0 = p_T$ and QUIT

In order for Algorithm 1 to succeed, we require the following property (*) to be satisfied by the set system $\mathcal{T}$:

For any $B \subseteq \{1, \ldots, n\}$, $|B| \leq t$, there exists a block $T \in \mathcal{T}$ such that $B \cap T = \emptyset$.

A collection $\mathcal{T}$ of k-subsets of $\{1, \ldots, n\}$ (called *blocks*) is an (n, k, t)-*covering* if every t-subset of $\{1, \ldots, n\}$ is contained in at least one block. The following lemma is obvious.

Lemma 1 *A set system $\mathcal{T}$ satsifies (*) if and only if the set system*

$$\{\{1, \ldots, n\} \backslash T : T \in \mathcal{T}\}$$

is an $(n, n - k, t)$-covering.

3 Coverings

Let $C(v,k,t)$ denote the minimum number of blocks in a (v,k,t)-covering. Then, $C(n,n-k,t)$ provides an upper bound on the number of iterations required by Algorithm 1. Some results on covering numbers of this form can be found in Mills [2] and Todorov [5]. We briefly summarize some known results now.

Theorem 2 [2, eq. (2.16), p. 221] *If $n \geq k(t+1)$, then $C(n,n-k,t) = t+1$.*

For $n \geq k(t+1)$, the set $\mathcal{T}$ can be taken to be $t+1$ disjoint k-subsets of $\{1,\ldots,n\}$. It is clear that this collection of subsets satisfies property (*), since any element of a t-subset is contained in at most one of the given blocks.

Theorem 3 [2, eq. (2.17), p. 221] *If $k(t+1) > n \geq k(t+1/2)$, then $C(n,n-k,t) = t+2$.*

For $k(t+1) > n \geq k(t+1/2)$ and k even, we can constuct the set $\mathcal{T}$ as follows. First, take $t-1$ disjoint k-subsets of $\{1,\ldots,n\}$. Then, construct three further k-subsets on a disjoint set of $3k/2$ points, such that each of the $3k/2$ points occurs in two of the three blocks. (This can be done since $n \geq k(t+1/2) = k(t-1)+3k/2$.) We show that this collection of blocks satisfies property (*). First, $t-1$ points are required to hit the $t-1$ disjoint blocks. Then, since any additional point hits only two of the remaining three blocks, (*) is satisfied.

In general, from [2, Theorem 2.4] we have the following result.

Theorem 4 *Suppose s and t are integers such that $3 \leq s \leq (t+3)/2$, and suppose that*

$$k\left(t-\frac{s-3}{2}\right) \leq n < k\left(t-\frac{s-4}{2}\right). \tag{1}$$

If k is odd and

$$k\left(t-\frac{s-3}{2}\right) \leq n < k\left(t-\frac{s-3}{2}\right)+\frac{s-3}{2},$$

then $C(n,n-k,t) \geq t+s+1$. Otherwise, $C(n,n-k,t) = t+s$.

If (1) is satisfied and k is even, then we can constuct the set $\mathcal{T}$ by generalizing the construction given after Theorem 3, as follows. First, take $t-2s+3$ disjoint k-subsets, say $A_1,\ldots,A_{t-2s+3}$. Next, for $1 \leq i \leq s-1$, let B_i, C_i, D_i be disjoint $(k/2)$-subsets, and construct blocks $B_i \cup C_i$, $C_i \cup D_i$, and $D_i \cup B_i$. We have a collection $\mathcal{T}$ of $t-2s+3+3(s-1) = t+s$ blocks of size k. Further, the cardinality of the union of these blocks is

$$k(t-2s+3)+\frac{3k(s-1)}{2} = k\left(t-\frac{s-3}{2}\right) \leq n.$$

We show that $\mathcal{T}$ satisfies property (*). First, $t-2s+3$ points are required to hit the blocks $A_1,\ldots,A_{t-2s+3}$. Then, for $1 \leq i \leq s-1$, we require two points to hit the three blocks $B_i \cup C_i$, $C_i \cup D_i$, and $D_i \cup B_i$. Hence, any set of $t-2s+3+2(s-1)-1 = t$ points is disjoint from at least one block in $\mathcal{T}$.

Remarks:

1. Theorems 2, 3 and 4 were proved independently by Sidorenko [3], using the terminology of Turán systems.

2. There are more known results about the covering numbers $C(n, n-k, 2)$ in [2]. For example, it is shown there that $C(n, n-k, 2) = 5$ for $\frac{5}{2}k > n \geq \frac{9}{4}k$; and $C(n, n-k, 2) = 6$ for $\frac{9}{4}k > n \geq \frac{7}{4}k$.

Observe that Theorems 2, 3 and 4 leave a finite interval of covering numbers undetermined for any fixed values of k and t. For "small" values of k and t, the missing numbers can be found in the tables presented in [1]. As an example, consider the case $t = k = 3$. We have that $C(n, n-3, 3) = 4$ for $n \geq 12$, by Theorem 2. $C(11, 8, 3) = 5$ by Theorem 3. $C(10, 7, 3) = 6$ by Theorem 4. The remaining values of $C(n, n-3, 3)$ are found in [1]: $C(9, 6, 3) = 7$, $C(8, 5, 3) = 8$, $C(7, 4, 3) = 12$ and $C(6, 3, 3) = 20$.

For the case $k = 4$ and $t = 3$, we have $C(n, n-4, 3) = 4$ for $n \geq 16$ from Theorem 2; $C(n, n-4, 3) = 5$ for $n = 14, 15$ from Theorem 3; and $C(n, n-3, 3) = 6$ for $n = 12, 13$ from Theorem 4. The remaining covering numbers can be found in [1]: $C(11, 7, 3) = 8, C(10, 6, 3) = 10$.

In the rest of this section, we present two simple new constructions for coverings. Our first construction is described in the following theorem.

Theorem 5 *Suppose $n \geq k + st$ and $k \equiv 0 \bmod s$. Then*

$$C(n, n-k, t) \leq \binom{t + \frac{k}{s}}{\frac{k}{s}}.$$

Proof. Let $\mathcal{A} = \{A_1, A_2, \ldots, A_{t+\frac{k}{s}}\}$ be a set of disjoint s-subsets of $\{1, \ldots, n\}$. Let $\mathcal{T}$ be be formed by taking all unions of k/s subsets in $\mathcal{A}$. We show that $\mathcal{T}$ satisfies property (*). In fact, a set B of t points hits at most t subsets in $\mathcal{A}$. Hence, at least k/s subsets in $\mathcal{A}$ are disjoint from B, so there is a block $T \in \mathcal{T}$ which is disjoint from B. $\square$

In general, Theorem 5 does not produce optimal coverings. However, one case in which Theorem 5 does yield optimal coverings is $s = k$, when the covering resulting from Theorem 5 is the same as that of Theorem 2. Another case of interest is when k is even and $s = k/2$, in which the following corollary of Theorem 5 is obtained.

Corollary 6 *Suppose k is even. Then*

$$C(k(t/2+1), kt/2, t) \leq \binom{t+2}{2}.$$

We note that the value $n = k(t/2+1)$ is not covered by any of Theorems 2, 3 or 4. In some cases, Corollary 6 gives optimal coverings. For example, letting $k = 4$ and $t = 3$ in Corollary 6, we have $C(10, 6, 3) \leq 10$. Actually, we mentioned above that $C(10, 6, 3) = 10$. Two other small examples are $k = 6$, $t = 2$; and $k = 4$, $t = 2$. Corollary 6 yields $C(12, 6, 2) \leq 6$ and $C(8, 4, 2) \leq 6$. In fact, $C(12, 6, 2) = 6$ and $C(8, 4, 2) = 6$ (see, e.g., [1]).

When $k \equiv l \bmod s$ for $l > 0$ and $n \geq k + (s+l)t$, we can obtain a covering as follows. Let $\mathcal{A}$ consist of $2t + \lfloor \frac{k}{s} \rfloor + 1$ disjoint subsets of $\{1, \ldots, n\}$, where $t + \lfloor \frac{k}{s} \rfloor$ of them are of size s and the other $t+1$ subsets are of size l. Since $n \geq k + (s+l)t$, such a system exists. Then the blocks of $\mathcal{T}$ are the unions of $\lfloor \frac{k}{s} \rfloor$ s-subsets and one l-subset of $\mathcal{A}$. A similar argument shows that $\mathcal{T}$ satisfies the property (*). Thus we have the following.

Theorem 7 *Suppose $k \equiv l \bmod s$, $l > 0$, and $n \geq k + (s+l)t$. Then*

$$C(n, n-k, t) \leq (t+1)\binom{t + \lfloor \frac{k}{s} \rfloor}{\lfloor \frac{k}{s} \rfloor}.$$

4 A randomized algorithm

Next, we provide a randomized algorithm to compute p_0.

Algorithm 2

Input S, n, k and t.

REPEAT the following steps:

1. Let T be a random k-subset of $\{1, 2, \ldots, n\}$.
2. compute p_T
3. compute C_T
4. if $|C_T| \geq n - t$, then set $p_0 = p_T$ and QUIT; otherwise, proceed to the next iteration of the REPEAT loop.

Note that Algorithm 2 is a Las Vegas type algorithm, since it terminates if and only if the correct polynomial p_0 has been found. In any iteration, the algorithm is successful if T contains no bad shares. If there are exactly t bad shares, this happens with probability

$$p = \frac{\binom{n-t}{k}}{\binom{n}{k}}.$$

Hence, the expected number of iterations of Algorithm 2 is

$$\beta_r = \frac{1}{p} = \frac{\binom{n}{k}}{\binom{n-t}{k}} = \frac{n(n-1)\cdots(n-k+1)}{(n-t)(n-t-1)\cdots(n-t-k+1)}.$$

5 Comparison of the two algorithms

The value β_r derived above is an average computed over all possible random choices made in Algorithm 2. In order to compare this to Algorithm 1, we should compute the average-case complexity of Algorithm 1. This requires specifying a particular set system $\mathcal{T}$, and we will use the set system from Theorem 2. Since Algorithm 1 is a deterministic algorithm, we compute the average number of iterations over all possible t-subsets B.

Suppose $n \geq k(t+1)$, and Let $\mathcal{T} = \{T_1, T_2, \ldots, T_{t+1}\}$ be the set system containing $t+1$ disjoint k-subsets of $\{1, \ldots, n\}$, in which $T_i = \{(i-1)k+1, \ldots, ik\}$ for $1 \leq i \leq t+1$. For a t-subset B, define

$$i_{\mathsf{B}} = \min\{i : \mathsf{B} \cap T_i = \emptyset\}.$$

Let $\psi(j)$ denote the number of t-subsets B such that $i_{\mathsf{B}} = j$. Then the average number of iterations required for Algorithm 1 is

$$\frac{\sum_{j=1}^{t+1} j\,\psi(j)}{\binom{n}{t}}.$$

By a simple application of the inclusion-exclusion principle, we have

$$\psi(j) = \sum_{i=1}^{j} (-1)^{i+1} \binom{n-ik}{t} \binom{j-1}{i-1}$$

for $1 \leq j \leq t+1$. Therefore, it follows that

$$\begin{aligned}
\sum_{j=1}^{t+1} j\,\psi(j) &= \sum_{j=1}^{t+1} j \sum_{i=1}^{j} (-1)^{i+1} \binom{n-ik}{t} \binom{j-1}{i-1} \\
&= \sum_{i=1}^{t+1} (-1)^{i+1} \binom{n-ik}{t} \sum_{j=i}^{t+1} j \binom{j-1}{i-1} \\
&= \sum_{i=1}^{t+1} (-1)^{i+1} \binom{n-ik}{t} i \sum_{j=i}^{t+1} \binom{j}{i} \\
&= \sum_{i=1}^{t+1} (-1)^{i+1} \binom{n-ik}{t} i \binom{t+2}{i+1}.
\end{aligned}$$

Hence, the average number of iterations required for Algorithm 1 is

$$\beta_d = \sum_{i=1}^{t+1} (-1)^{i+1} i \binom{n-ik}{t} \binom{t+2}{i+1} \Big/ \binom{n}{t}.$$

Table 1 lists some of the values of β_r and β_d. These values are very close, especially for large n. Also, observe that $\beta_d < \beta_r$ for all values computed. Thus the average-case complexities of the two algorithms are similar, and indeed, there is no real advantage in using the randomized algorithm, at least when $n \geq k(t+1)$. For $n < k(t+1)$, the randomized algorithm might be prefered since the required coverings become more difficult to construct and there is no uniform description of "good" coverings for parameters in this range.

Table 1: Comparison of the deterministic and randomized algorithms

n	t	k	β_d	β_r	$\frac{\beta_d}{\beta_r}$
9	2	3	1.833	2.400	.764
10	2	3	1.733	2.143	.809
11	2	3	1.655	1.964	.842
12	2	3	1.591	1.833	.868
13	2	3	1.538	1.733	.888
59	2	3	1.105	1.111	.995
109	2	3	1.056	1.058	.998
159	2	3	1.038	1.039	.999
209	2	3	1.029	1.029	1.000
12	2	4	1.818	2.357	.771
13	2	4	1.744	2.167	.805
14	2	4	1.681	2.022	.831
15	2	4	1.629	1.909	.853
16	2	4	1.583	1.818	.871
62	2	4	1.134	1.144	.992
112	2	4	1.073	1.076	.997
162	2	4	1.050	1.051	.999
262	2	4	1.031	1.031	1.000
16	3	4	2.036	2.545	.800
17	3	4	1.956	2.378	.823
18	3	4	1.887	2.242	.842
19	3	4	1.828	2.130	.858
20	3	4	1.775	2.036	.872
66	3	4	1.196	1.210	.989
116	3	4	1.108	1.112	.996
216	3	4	1.057	1.058	.999
316	3	4	1.039	1.039	1.000

References

[1] D. M. Gordon, G. Kuperberg and O. Patashnik. New constructions for covering designs, *J. Combin. Designs* **3** (1995), 269–284.

[2] W. H. Mills. Covering designs I: coverings by a small number of subsets, *Ars Combin.* **8** (1979), 199–315.

[3] A. F. Sidorenko. *Extremal constraints and inequalities for distributions of sums of random vectors* (in Russian), PhD Thesis, Moscow State University, 1982.

[4] D. R. Stinson. *Cryptography Theory and Practice*, CRC Presss, Inc., 1995.

[5] D. T. Todorov. On some covering designs, *J. Combin. Theory A* **39** (1985), 83–101.