How to share secrets simultaneously

László Csirmaz[*]

Central European University, Budapest
Rényi Institute, Budapest

Abstract

Each member of a team consisting of n person has a secret. The *k out of n simultaneous threshold secret sharing* requires that any group of k members should be able to recover the secret of the other $n-k$ members, while any group of $k-1$ or less members should have no information on the secret of other team members. We show that when all secrets are independent and have size s then each team member must receive a share of size at least $(n-k)s$, and we present a scheme which achieves this bound. This result shows a significant saving over n independent applications of the k out of $n-1$ threshold schemes which assigns shares of size $(n-1)s$ to each team member independently of k.

Keywords: simultaneous secret sharing; complexity; threshold scheme; secret sharing; interpolation.

MSC numbers: 94A62, 90C25, 05B35.

1 Introduction

A team has n members, and each member of the team has a secret about the same size, say a password. As a safety caution, they want each secret to be distributed among the other members of the group so that it could be recovered in the case any of them forgets it. Also, none of them trusts the others, thus they want their secret to be independent of the information held by any group $n-1$ or less team members. This goal can be achieved by distributing all secrets using Shamir's $n-1$ out of $n-1$ threshold secret sharing method, see [3]. Assuming that all secrets have the same size s bit, the total size of the shares each team member receives will be $(n-1)s$ times the secret size. This holds as in any perfect secret sharing scheme, any share size is at least as large as that of the secret, see [1]. Can we do better if the secrets are distributed not independently but simultaneously? We show that the answer is *yes*: there is a way to distribute the secrets so that

[*]This research was partially supported by the "Lendület Program" of the Hungarian Academy of Sciences and by the grant NKTH OM-00289/2008

1. every team member receives an s bit share;
2. any member's secret can be recovered using the shares *and* the secrets of the other $n-1$ members;
3. even putting together all information $n-2$ team members received during the distribution phase, they will have no information on the secret of the other two team members.

We also show that in all schemes satisfying properties 2. and 3. above the size of every share must be at least s, thus our scheme is *optimal.*

The problem sketched above is a special case of the *k out of n simultaneous secret sharing.* The team has n members as above, and each of them has a secret. In this case we require that any participant's secret should be recoverable by any k out of the remaining $n-1$ participants, while no $k-1$ or less coalition should have any information on the other members' secrets.

Theorem 1 *a) Suppose each secret has the same size s, and the secrets are totally independent (knowing some of them does not help guessing the others beyond guessing them randomly and independently). Then any scheme realizing k out of n simultaneous secret sharing assigns shares of size at least $(n-k)s$.*

b) There is an optimal *k out of n simultaneous secret sharing scheme which assigns $(n-k)s$ size shares.*

In Section 2 we prove part a) of this theorem, while in Section 3 we present an optimal k out of n simultaneous secret sharing scheme proving part b). An interesting property of the presented scheme is that the recovery of a secret can be done in such a way that repeating it not more than $n-k$ times the unaffected secrets are not compromised, i.e., they are (statistically) independent from all published information.

2 Lower bound

To prove the lower bound we use the so-called entropy method, see [1, 2]. First of all, we consider the secrets and shares as random variables. The *size* of the secret ξ_p belonging to participant p is its Shannon entropy $\mathbf{H}(\xi_p)$, which is roughly the number of necessary bits to defined the value of ξ_p uniquely. For any collection $\{\xi_i : i \in I\}$ of random variables we define the real-valued function

$$f(I) = \mathbf{H}(\{\xi_i : i \in I\})$$

where the entropy is taken for the joint distribution of all indicated variables. For example, if p is (the index of) any of the secrets, then $f(\{p\})$ is the *size* of the secret ξ_p, and similarly for shares.

The function f is defined on all subset of some finite set, and satisfies certain linear inequalities which follow from the so-called Shannon inequalities for the entropy function $\mathbf{H}$. The following claim collects those properties which will be used in this paper. As usual, we write $f(XY)$ instead of $f(X \cup Y)$, and $f(x)$ and $f(xX)$ instead of $f(\{x\})$ and $f(\{x\} \cup X)$.

Claim 2 *For any subsets X and Y*

1. $f(X) \geq 0$ *(positivity),*
2. $f(X) \leq f(Y)$ *if* $X \subseteq Y$ *(monotonicity),*
3. $f(X) + f(Y) \geq f(XY)$ *(additivity),*
4. $f(XY) = f(X)$ *if (the variables in) X determines the values of (the variables in) Y;*
5. $f(XY) = f(X) + f(Y)$ *if X and Y are statistically independent.*

The entropy method can be rephrased in a few words as follows. Let A be the (index) of any share. Suppose for *any function* f satisfying properties enlisted in Claim 2 there are (indices) a_1, ..., a_ℓ of secrets such that

$$f(A) \geq f(a_1) + \cdots f(a_\ell).$$

The the size of share A must be at least ℓ times the size of the secrets.

Lemma 3 *Suppose G is a group of participant with $k-1$ members, and a, $\bar{b} = \langle b_1, \ldots, b_{n-k} \rangle$ are the secrets of participants not in G and their share are A and $\bar{B} = \langle B_1, \ldots, B_{n-k} \rangle$, respectively. Then*

$$f(a) + f(A) \geq f(a\bar{b}).$$

Proof Let us denote the total data (secret plus share) held by G by G as well. By assumption, G together with a and A should determine all the secrets b_i, that is $f(aAG) = f(a\bar{b}AG)$. Also, G should have no information on the secrets a and b_i or on their combinations, thus $f(a\bar{b}G) = f(a\bar{b}) + f(G)$. Using these and the additivity and monotonicity property of f, we have

$$f(a) + f(A) + f(G) \geq f(aAG) = f(a\bar{b}AG) \geq f(a\bar{b}G) = f(a\bar{b}) + f(G).$$

Comparing the first and last tag gives the claim of the Lemma. □

From this lemma we can deduct a lower bound on the size of the share each participant receives.

Proof (of part a) of Theorem 1) Use notations from Lemma 3, in particular let a and A respectively be the secret and the share of participant a. All secrets have the same size, thus

$$f(a) = f(b_1) = \cdots = f(b_{n-k}).$$

Secrets are totally independent by assumption, which means

$$f(a\bar{b}) = f(ab_1 \ldots b_{n-k}) = f(a) + f(b_1) + \cdots + f(b_{n-k}) = (n-k+1)f(a).$$

From Lemma 3 we know that $f(A) \geq f(a\bar{b}) - f(a) = (n-k)f(a)$, which proves part a) of Theorem 1. □

3 An optimal construction

In the rest of this paper we give a construction which matches the bound in part a) of Theorem 1. Let us denote the participants by p_i for $1 \leq i \leq n$, and let $\mathbb{F}$ be a finite field with more than $n(n-k+1)$ elements. Choose different field elements $x_{i,j}$ for $1 \leq i \leq n$ and $0 \leq j \leq n-k$, and pick a random polynomial $r(x)$ of degree less than $k(n-k+1)$.

The *secret* of participant p_i will be the value of r at $x_{i,0}$. This can also be achieved by simply choosing r from among those polynomials which satisfy the condition that $r(x_{i,0})$ is the secret of p_i. As for the share, we give participant p_i all field elements $r(x_{i,1})$ up to $r(x_{i,n-k})$. Observe that all secrets are uniform random elements from the field, thus the "size" of all secrets are the same, namely $log_2(|\mathbb{F}|)$. Similarly, all participants received $(n-k)$ field elements as share, therefore the size of the share is exactly $(n-k)$ times that of the secret.

We claim that any k participants can determine the secret value of the remaining $n-k$ participants. This is clear, as the k participants know the value of r at $k(n-k+1)$ different places, while r has smaller degree, thus they can determine r, and its value at $x_{p,0}$ for any participant p.

Next, we claim that the total information of $k-1$ participants is statistically independent of the secrets of the other $n-k+1$ participants. This is true as r is a random polynomial of degree below $k(n-k+1)$, and $k-1$ participants know the value of this polynomial at $(k-1)(n-k+1)$ places, thus the polynomial can take all the possibilities with equal probability at any $n-k+1$ predetermined places – in particular at $x_{p,0}$ where p runs over the missing $n-k+1$ participants.

The method outlined above to recover one of the secrets has the drawback that it not only recovers the secret but it also recovers the polynomial r, consequently all the secrets are also revealed. The participants, however, should only recover the value of r at $x_{p,0}$ and not the whole r. Let $B \subseteq \{1, \ldots, n\}$ be the subset of size k which wants to recover the secret of $p \notin B$. As the values $x_{i,j}$ are public, members of B can publicly compute the constants $\lambda_{i,j} \in \mathbb{F}$ using the Lagrange interpolation formula such that

$$r(x_{p,0} = \sum_{i \in B} \sum_{j=0}^{n-k} \lambda_{i,j} r(x_{i,j})$$

whatever the values $r(x_{i,j})$ are. Consequently to recover p's secret, participant $i \in B$ should only publish the sum

$$\sum_{j=0}^{n-k} \lambda_{i,j} r(x_{i,j}) \tag{1}$$

rather than all the values $r(x_{i,j})$. The sum of published values (1) will give the secret, while even the totality of all revealed values give no information on the secret values of members in B. The same remains true when they repeat the recovery process at most $n-k$ times as the sum in (1) has $n-k+1$ terms.

References

[1] R. M. Capocelli, A. De Santis, L. Gargano, U. Vaccaro: On the size of shares of secret sharing schemes, *J. Cryptology*, vol 6(3) (1993), pp. 157–168

[2] L. Csirmaz: Secret sharing schemes on graphs, *Studia Sci. Math. Hungar.*, vol 44(2007) pp. 297–306 – available as IACR preprint `http://eprint.iacr.org/2005/059`

[3] A. Shamir: How to share a secret, *Commun. of the ACM*, vol 22 (1979) pp. 612–613