Wireless Pers Commun (2014) 77:383–394
DOI 10.1007/s11277-013-1511-3

An Ideal Multi-secret Sharing Scheme Based on Connectivity of Graphs

Ching-Fang Hsu · Lein Harn · Guohua Cui

Published online: 12 November 2013

Abstract Secure communication has become more and more important for many modern communication applications. In a secure communication, every pair of users need to have a secure communication channel (each channel is controlled by a server) In this paper, using monotone span programs we devise an ideal linear multi-secret sharing scheme based on connectivity of graphs. In our proposed scheme, we assume that every pair of users, p and q, use the secret key s_{pq} to communicate with each other and every server has a secret share such that a set of servers can recover s_{pq} if the channels controlled by the servers in this set can connect users, p and q. The multi-secret sharing scheme can provide efficiency for key management. We also prove that the proposed scheme satisfies the definition of a perfect multi-secret sharing scheme. Our proposed scheme is desirable for secure and efficient secure communications.

Keywords Secure communication · Connectivity of graphs · Ideal linear multi-secret sharing schemes · Monotone span programs

This work has been supported by the National Natural Science Foundation of China (No: 61100221, 61272405, 61272451).

C.-F. Hsu (✉)
Computer School, Central China Normal University, Wuhan 430079, China
e-mail: cherryjingfang@gmail.com

L. Harn
Department of Computer Science Electrical Engineering, University of Missouri-Kansas City,
Kansas City, MO 64110, USA

G. Cui
College of Computer Science and Technology,
Huazhong University of Science and Technology, Wuhan 430074, China

1 Introduction

Secret sharing schemes, which were introduced by Shamir [1] and Blakley [2] independently more than 30 years ago, have been widely used in many secure communications. In these schemes, there is a finite set of participants and a collection, Γ, of subsets of the participants (called the access structure). A secret sharing scheme for Γ is a method in which a dealer distributes shares of a secret to the participants such that (1) any subset in Γ can reconstruct the secret from its shares, and (2) any subset not in Γ cannot reveal any partial information about the secret in the information theoretic sense.

A secret sharing scheme is called ideal if the shares of participants are taken from the same domain as the secret. As proved in [3], this is the minimal size of the shares. The access structures which can be realized by ideal secret-sharing schemes are called ideal access structures. This definition of ideal single-secret sharing can be extended to multi-secret sharing, that is, a secret sharing scheme with multiple secrets is said to be ideal if all secrets and the shares of participants have the same size. This is the optimal efficiency for multi-secret sharing schemes.

Many secret sharing applications, in particular those associated with the key management, require the protection of more than one secret. As an example, consider the following situation described by Simmons [4]. There is a missile battery in which each missile has a different launch enable code. The problem is to devise a scheme to protect these codes by using the same pieces of private information. This problem could be trivially solved by realizing different secret sharing schemes, one for each launch enable code; but in this case each participant should remember too much information. In order to reduce the amount of information given to participants, it is interesting to investigate the possibility of constructing multiple secret sharing schemes without necessarily using multiple single secret sharing schemes.

Specific models for the sharing of many secrets have already been considered in the literature. In [3], Karnin et al. considered the problem of sharing m secrets $s_1, \ldots, s_m$ among a set of n participants. In particular, they considered the situation in which, for a fixed value $k \leq n$, any set of k participants can reconstruct the secret s_j, for $1 \leq j \leq m$, whereas, any subset of $k-1$ participants has no information about the secret s_j. These schemes are called (m, k, n) multi-secret sharing threshold schemes, which have been constructed by several papers (see [5–10]). Schemes of this kind have also been considered by Jackson et al. [11,12]. In particular, they considered the situation in which, for a fixed value, $t \in \{1, 2, \ldots, n\}$, there is a secret associated with each subset $\mathcal{P}' \subseteq \mathcal{P}$, such that $|\mathcal{P}'| = t$. For a fixed parameter, $k \leq t$, this secret can be reconstructed by any k participants in $\mathcal{P}'$. They proved bounds on the size of the information that participants must be held in order to ensure that up to w participants ($0 \leq w \leq n - t + k - 1$) cannot obtain any information about a secret which they have no access right. Such schemes are referred to a w-secure (k, t, n) multi-threshold schemes.

In [13], the problem of sharing multiple secrets among a set of shareholders has been generalized to the case where all secrets are shared according to a general secret access structure. In the proposed model, any qualified set of participants can recover all the secrets, whereas, any unqualified set of participants has absolutely no information about each secret but, knowing some secrets, might have some information about the other secrets. Schemes of this kind have also been considered in [14], where some optimal constructions have been proposed. The problem of sharing many secrets according to different access structures has been considered in [15] and further investigated in [16], where a classification of ideal secret sharing schemes with multiple secrets has been proposed.

By using a multi-party computation protocol, [17] solved a secret-leaking problem in multi-secret sharing schemes. They also showed that the non-direct sum linear multi-secret sharing scheme was preferred in reducing share expansion after comparing it with the associated “direct sum” scheme. A corresponding relation between monotone span programs and linear multi-secret sharing schemes has been studied in [18], where the optimal linear multi-secret sharing schemes have also been discussed. These results are fairly interesting as to how to construct ideal multi-secret sharing schemes for general access structures.

So far, very little is known about how to devise ideal multi-secret sharing schemes for general access structures. Due to the difficulty of finding general results, the construction of ideal multi-secret sharing schemes for specific families of access structures may have interesting applications which is worth investigation (see [19,20]). Today, secure communication has become more and more important for many communication applications. In this paper, we consider the following scenario in a group communication. We assume that there are t users and every pair of users have a secure communication channel connecting them (each channel is controlled by a server). Because some communication channels may be shut down due to natural cause or by human interruption, the secure communication service between two users should be maintained properly as long as the remaining channels can establish an alternative connection between two users. To achieve this objective of secure services, we advise to use a secret key to encrypt message between every pair of users. There are all $t(t-1)/2$ secret keys, where $t$is the number of users in the communication. How to efficiently share these $t(t-1)/2$ secret keys among $t(t-1)/2$ servers such that for every pair of users, a set of servers who provide the alternative connection between users can recover the shared secret key of two users is the subject of our proposed scheme.

Using *monotone span programs* (MSP), we devise an ideal *linear multi-secret sharing scheme* (LMSS) based on connectivity of graphs. In our proposed scheme, the dealer is a trusted entity, the servers act the role of shareholders in a secret sharing scheme, the communication channels are denoted by edges and the users are denoted by vertices. We propose an efficient multi-secret sharing scheme. In addition, we prove that the proposed scheme satisfies the definition of a perfect multi-secret sharing scheme. Here, we list the contributions of our paper.

- A novel LMSS based on connectivity of graphs is proposed, which ensures secure and fault-tolerant communications.
- The efficiency of proposed scheme is optimal. Namely, it is an ideal LMSS and all secrets and the shares of participants are the same size.
- The proposed scheme is a perfect LMSS, that is, a set of servers who provide the alternative connection between two users can recover the shared secret key between two users.

The rest of the paper is organized as follows: In Sect. 2, some preliminaries are reviewed. The MSP to permit more than one target vector is introduced in Sect. 3. Then, in Sect. 4, we build an ideal LMSSS based on connectivity of graphs. Section 5 proves the correctness and security of the proposed scheme. Conclusions are given in Sect. 6.

2 Preliminaries

In this section, we review some basic definitions related to secret sharing schemes. For a unified description and detailed proof of results in the subject of secret sharing schemes, the reader can refer to the survey articles by Simmons [4] and Stinson [21].

2.1 Positive Access Structures and Negative Access Structures

Let $\mathcal{P} = \{1, \ldots, n\}$ be the set of players. A positive access structure, denoted by Γ, is a collection of subsets of $\mathcal{P}$ satisfying the monotone ascending property: for any $A' \in \Gamma$ and $A \in 2^{\mathcal{P}}$, $A' \subseteq A$ implies $A \in \Gamma$. A negative structure, denoted by $\mathcal{A}$, is a collection of subsets of $\mathcal{P}$ satisfying the monotone descending property: for any $A' \in \mathcal{A}$ and $A \in 2^{\mathcal{P}}$, $A \subseteq A'$ implies $A \in \mathcal{A}$. Because of the monotone properties, for any positive access structure Γ and any negative access structure $\mathcal{A}$, it is enough to consider the minimum positive access structure $\Gamma_{\min} = \{A \in \Gamma \,|\, \forall B \subset A \Rightarrow B \notin \Gamma\}$ and the maximum negative access structure $\mathcal{A}_{\max} = \{B \in \mathcal{A} \,|\, \forall A \supset B \Rightarrow A \notin \mathcal{A}\}$, respectively. In this paper, we consider the complete situation, i.e., $\mathcal{A} = 2^{\mathcal{P}} - \Gamma$.

We use the following example to illustrate these two structures. We assume that there are 3 users, A, B and C. The positive access structure of a secret is $\Gamma = \{\{AB\}, \{BC\}, \{ABC\}\}$. By taking the logically complement of the access structure, we obtain the negative access structure as A={{A′B′},{A′C′},{A′B′C′},{B′C′}}. We can obtain that the minimum positive access structure is $\Gamma_{\min} = \{\{AB\}, \{BC\}\}$ and the maximum negative access structure is A$_{\max}$={{B′}.{A′B′}}.

2.2 Linear Secret Sharing Schemes and Monotone Span Programs

Suppose that S is the secret-domain and P_i is the share-domain of player i, where $1 \leq i \leq n$. When a dealer D wants to share a secret $s \in S$ among a set of players $\mathcal{P} = \{1, \ldots, n\}$, he will give each player a share $p_i \in P_i$. The shares should be distributed secretly, so no player knows the share given to another player. At a later time, a subset of players will attempt to reconstruct the secret s from the shares they collectively hold. By using Shannon's entropy function, a secret sharing scheme with respect to an access structure Γ is defined such that the following requirements are satisfied.

(i) Correctness requirement: any subset $A \subseteq \mathcal{P}$ of players enabled to recover s can compute s. Formally, for all $A \in \Gamma$, it holds $H(S \,|\, A) = 0$.
(ii) Security requirement: any subset $A \subseteq \mathcal{P}$ of players not enabled to recover s, even pooling all of their shares together, can not reconstruct s. Formally, for all $A \notin \Gamma$, it holds $0 < H(S \,|\, A) \leq H(S)$.

In the security requirement, if for any $A \notin \Gamma$ it holds $H(S \,|\, A) = H(S)$ (that is, players in A pool their shares together obtain no information on s), we call it a perfect secret sharing scheme which we are interested in. If $|\, S\,| = |P_i|$ for $1 \leq i \leq n$, then the secret sharing scheme is called ideal. Furthermore, a perfect secret sharing scheme is linear, if $S = \mathcal{K}$ is a finite field, P_i are linear spaces over $\mathcal{K}$ and the reconstruction operations are linear [22].

Karchmer and Wigderson [23] introduced monotone span programs (MSP) as linear models computing monotone Boolean functions. Usually we denote an MSP by $\mathcal{M}(\mathcal{K}, M, \psi)$, where M is a $d \times l$ matrix over a finite field $\mathcal{K}$ and $\psi : \{1, \ldots, d\} \rightarrow \{1, \ldots, n\}$ is a surjective labeling map which actually distributes to each player some rows of M. We call d the size of the MSP. For any subset $A \subseteq \mathcal{P}$, there is a corresponding characteristic vector $\vec{\delta}_A = (\delta_1, \ldots, \delta_n) \in \{0, 1\}^n$ where for $1 \leq i \leq n$, $\delta_i = 1$ if and only if $i \in A$. Consider a monotone Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ which satisfies that for any $A \subseteq \mathcal{P}$ and $B \subseteq A$, $f(\vec{\delta}_B) = 1$ implies $f(\vec{\delta}_A) = 1$. We say that an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ computes the monotone Boolean function f with respect to a target vector $\vec{v} \in \mathcal{K}^l \backslash \{(0, \ldots 0)\}$, if it holds that $\vec{v} \in span\{M_A\}$ if and only if $f(\vec{\delta}_A) = 1$, where M_A consists of the rows r of M with $\psi(r) \in A$ and $\vec{v} \in span\{M_A\}$ means that there exists a vector $\vec{w}$ such that $\vec{v} = \vec{w} M_A$.

Beimel [22] proved that devising a linear secret sharing scheme (LSSS) for an access structure Γ is equivalent to constructing an MSP computing the monotone Boolean function f_Γ which satisfies $f_\Gamma(\vec{\delta}_A) = 1$ if and only if $A \in \Gamma$. On the other hand, an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ can compute f_Γ if and only if there exists a vector $\vec{v}$ which lies in the space $\bigcap_{A \in \Gamma_{\min}} \sum_{i \in A} V_i - \bigcup_{B \in \mathcal{A}_{\max}} \sum_{i \in B} V_i$, where V_i is the space spanned by the row vectors of M distributed to player i according to ψ and the vector $\vec{v}$ can be seemed as the target vector described above. Hence, finding the linear spaces V_i with the condition $\bigcap_{A \in \Gamma_{\min}} \sum_{i \in A} V_i - \bigcup_{B \in \mathcal{A}_{\max}} \sum_{i \in B} V_i \neq \varnothing$ is the key point of building an LSSS with respect to Γ.

2.3 Multi-secret Sharing Schemes

The problem of sharing many secrets simultaneously among the same set of players has been considered (with some differences in the definitions) by several researchers (see [3,11,13–16,24]). The most straightforward definition of a multi-secret sharing scheme for m secrets $s_1, \ldots, s_m$ is a natural generalization of single secret sharing schemes, where for any $1 \leq j \leq m$, each secret s_j is associated with a (potentially different) access structure Γ_j on $\mathcal{P}$. Let $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ be the m-tuple of access structures and let $S_1 \times \cdots \times S_m$ be the set from which the secrets are chosen (the jth secret to be shared is chosen in S_j), according to some probability distribution on such a set. For any $i \in \mathcal{P}$, denote by P_i the set of all possible shares given to player i.

In the definition of a perfect multi-secret sharing scheme, an m-tuple of secrets $(s_1, \ldots, s_m) \in S_1 \times \cdots \times S_m$ is shared in an m-tuple $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ of access structures on $\mathcal{P}$ in such a way that, for each $j = 1, \ldots, m$, the access structure Γ_j is the set of all subsets of $\mathcal{P}$ that can recover secret $s_j \in S_j$. This means that only the sets in Γ_j can recover the secret s_j, but any set $A \notin \Gamma_j$ has no more information on s_j than that already conveyed by the known secrets. A perfect multi-secret sharing scheme is defined in [15] as follows.

Definition 1 Let $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ be an m-tuple of access structures on the set of players $\mathcal{P}$. A multi-secret sharing scheme for $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ is a sharing of the secrets $(s_1, \ldots, s_m) \in S_1 \times \cdots \times S_m$ in such a way that, for $j = 1, \ldots, m$,

(i) Any subset $A \subseteq \mathcal{P}$ of players enabled to recover s_j can compute s_j. Formally, for all $A \in \Gamma_j$, it holds $H(S_j | A) = 0$.

(ii) Any subset $A \subseteq \mathcal{P}$ of players not enabled to recover s_j, even knowing some of the other secrets, has no more information on s_j than that already conveyed by the known secrets. Formally, for all $A \notin \Gamma_j$ and $T \subseteq \{S_1, \ldots, S_m\} \backslash \{S_j\}$, it holds $H(S_j | AT) = H(S_j | T)$.

2.4 Some Concepts About Graphs

Let $G(V, E)$ be a graph with the vertex set V and the edge set E. Two vertices $u, v \in V$ are *adjacent* if they are, respectively the two endpoints of an edge in G, i.e., $uv \in E$. A *path* in G is of the form $v_0 v_1 \ldots v_l$ such that $v_0, v_1, \ldots, v_l$ are distinct vertices in G and $v_j v_{j+1} \in E$ for $0 \leq j < l$. More precisely, we call it a path from v_0 to v_l. When $l \geq 2$ and v_l is adjacent to v_0, we get a *cycle*. Two vertices $u, v \in V$ are *connected*, if there is a path from u to v. The graph G is called *connected* if each pair of vertices in G are connected. A *tree* is a connected graph with no cycles.

A *subgraph* of a graph $G(V, E)$ is a graph $G'(V', E')$ such that $V' \subseteq V$ and $E' \subseteq E$. A *spanning subgraph* of $G(V, E)$ is a subgraph $G'(V', E')$ with the vertex set $V' = V$. A *spanning tree* is a spanning subgraph that is a tree. When a graph is not connected (i.e., disconnected), its maximal connected subgraphs are called *components*. An *isolated vertex* is a vertex that is not an endpoint of any edge.

3 MSP Computing *m* Monotone Boolean Functions

In this section we introduce the MSP computing m monotone Boolean functions.

Let $\mathcal{M}(\mathcal{K}, M, \psi)$ be an MSP to permit m target vectors $v_1, \dots, v_m$, where M is a $d \times l$ matrix over a finite field $\mathcal{K}$ and $\psi : \{1, \dots, d\} \to \{1, \dots, n\}$ is a surjective labeling map which actually distributes to each player some rows of M. We call d the size of the MSP. For any subset $A \subseteq \mathcal{P}$, there is a corresponding characteristic vector $\vec{\delta}_A = (\delta_1, \dots, \delta_n) \in \{0, 1\}^n$ where for $1 \leq i \leq n$, $\delta_i = 1$ if and only if $i \in A$. Consider a monotone Boolean function $f : \{0, 1\}^n \to \{0, 1\}$ which satisfies that for any $A \subseteq \mathcal{P}$ and $B \subseteq A$, $f(\vec{\delta}_B) = 1$ implies $f(\vec{\delta}_A) = 1$. We say that an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ computes m monotone Boolean functions $f_1, \dots, f_m$ with respect to m target vectors $\vec{v}_1, \dots, \vec{v}_m \in \mathcal{K}^l \backslash \{(0, \dots 0)\}$, if it holds that for each $j = 1, \dots, m$, $\vec{v}_j \in span\{M_A\}$ if and only if $f_j(\vec{\delta}_A) = 1$, where M_A consists of the rows r of M with $\psi(r) \in A$ and $\vec{v}_j \in span\{M_A\}$ means that there exists a vector $\vec{w}$ such that $\vec{v}_j = \vec{w} M_A$.

Likewise, by using the same method of proof in [22], it is easy to obtain that devising a linear multi-secret sharing scheme (LMSSS) for an m-tuple $\vec{\Gamma} = (\Gamma_1, \dots, \Gamma_m)$ of access structures is equivalent to constructing an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ computing m monotone Boolean functions $f_{\Gamma_1}, \dots, f_{\Gamma_m}$ which satisfies for each $j = 1, \dots, m$, $f_{\Gamma_j}(\vec{\delta}_A) = 1$ if and only if $A \in \Gamma_j$. On the other hand, an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ can compute f_{Γ_j} for each $j = 1, \dots, m$ if and only if there exists a vector $\vec{v}_j$ which lies in the space $\bigcap_{A \in (\Gamma_j)_{\min}} \sum_{i \in A} V_i - \bigcup_{B \in (\mathcal{A}_j)_{\max}} \sum_{i \in B} V_i$, where V_i is the space spanned by the row vectors of M distributed to player i according to ψ and the vector $\vec{v}_j$ can be seemed as the target vector described above. Hence, finding the linear spaces V_i with the condition $\bigcap_{A \in (\Gamma_j)_{\min}} \sum_{i \in A} V_i - \bigcup_{B \in (\mathcal{A}_j)_{\max}} \sum_{i \in B} V_i \neq \varnothing$ is the key point of building an LMSSS with respect to $\vec{\Gamma} = (\Gamma_1, \dots, \Gamma_m)$.

4 An Ideal LMSSS Based on Connectivity of Graphs

In this section we firstly define an m-tuple $\vec{\Gamma} = (\Gamma_1, \dots, \Gamma_m)$ of access structures with respect to the problem described in the introduction, i.e., the access structures is defined based on connectivity of graphs. Afterwards, we devise an ideal LMSSS which realizes such an m-tuple $\vec{\Gamma} = (\Gamma_1, \dots, \Gamma_m)$ of access structures.

4.1 Definition of the Access Structures

Let t be a positive integer, $n = \binom{t}{2} = t(t-1)/2$, and $\mathcal{P} = \{1, \dots, n\}$ be the set of players. Let $G(V, E)$ be an undirected complete graph with the vertex set $V = \{v_0, v_1, \dots, v_{t-1}\}$ and edge set $E = \{v_p v_q \mid p \neq q, 0 \leq p, q \leq t-1\}$. Suppose that $f : E \to P$ is a bijection which associates each player with an edge, in other words, associates every two vertices with a number. For any subset $A \subseteq P$, $G(V, E_A)$ is a spanning subgraph of $G(V, E)$ where $E_A = \{v_p v_q \in E \mid f(v_p v_q) \in A\}$. Seeing that every two vertices v_p and v_q use the related secret $s_{f(v_p v_q)}$ to communicate with each other and a set of players can recover $s_{f(v_p v_q)}$ if the edges related to the players in this set can construct a path from v_p to v_q, there are all

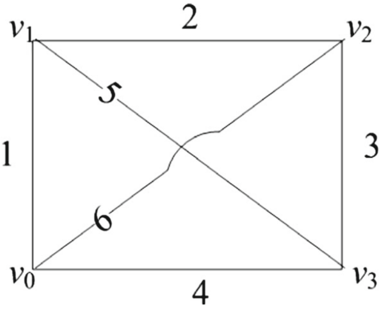

Fig. 1 The access structures for $t = 4$

$m = n = t(t-1)/2$ secrets such that for any $p \neq q$ and $0 \leq p, q \leq t-1$, each secret $s_{f(v_p v_q)}$ is associated with an access structure $\Gamma_{f(v_p v_q)}$ on $\mathcal{P}$ (naturally, $\mathcal{A}_{f(v_p v_q)} = 2^{\mathcal{P}} - \Gamma_{f(v_p v_q)}$). We define such an m-tuple $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ of access structures as follow:

$$\Gamma_{f(v_p v_q)} = \left\{ A \subseteq P \mid v_p \text{ and } v_q \text{ are connected in } G(V, E_A) \right\}, \quad p \neq q \text{ and } 0 \leq p, q \leq t-1. \tag{1}$$

Obviously, $\Gamma_{f(v_p v_q)}$ satisfies the monotone ascending property.

Example 1 Let $t = 4, n = 6$, and $V = \{v_0, v_1, v_2, v_3\}$. Let $\mathcal{P} = \{1, \ldots, 6\}$, $f(v_0 v_1) = 1$, $f(v_1 v_2) = 2$, $f(v_2 v_3) = 3$, $f(v_0 v_3) = 4$, $f(v_1 v_3) = 5$, $f(v_0 v_2) = 6$. See Fig. 1.

It is easy to see that $m = 6$ and there are all six secrets, which are shared in such a 6-tuple $\vec{\Gamma} = (\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5, \Gamma_6)$ of access structures on $\mathcal{P}$ as follows:

$$(\Gamma_1)_{\min} = \{\{1\}, \{2, 3, 4\}, \{2, 6\}, \{4, 5\}, \{3, 5, 6\}\}, \quad (\Gamma_2)_{\min} = \{\{2\}, \{1, 3, 4\}, \{1, 6\}, \{3, 5\}, \{4, 5, 6\}\},$$
$$(\Gamma_3)_{\min} = \{\{3\}, \{1, 2, 4\}, \{4, 6\}, \{2, 5\}, \{1, 5, 6\}\}, \quad (\Gamma_4)_{\min} = \{\{4\}, \{1, 2, 3\}, \{3, 6\}, \{1, 5\}, \{2, 5, 6\}\},$$
$$(\Gamma_5)_{\min} = \{\{5\}, \{1, 6, 3\}, \{1, 4\}, \{2, 3\}, \{2, 6, 4\}\}, \quad (\Gamma_6)_{\min} = \{\{6\}, \{1, 5, 3\}, \{1, 2\}, \{3, 4\}, \{2, 5, 4\}\}.$$

4.2 Construction of the LMSSS

As pointed out in Sect. 3, designing an LMSSS with respect to $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ is equivalent to building an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ by finding linear spaces V_i, $i \in \mathcal{P}$ such that for each $j = 1, \ldots, m$, $\bigcap_{A \in (\Gamma_j)_{\min}} \sum_{i \in A} V_i - \bigcup_{B \in (\mathcal{A}_j)_{\max}} \sum_{i \in B} V_i \neq \varnothing$ and letting any vector in this nonempty space be the target vector $\vec{v}_j$. Based on this fact, we build an LMSSS realizing $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ defined by (1) as follows. This scheme consists of three phases:

(1) The setup phase.

Let $S_1 \times \cdots \times S_m$ be the set from which the secrets are chosen (that is, s_j to be shared is chosen in S_j, $1 \leq j \leq m$). Let $S_1 = \cdots = S_m = \mathcal{K}$ be a finite field with the characteristic $char(\mathcal{K}) = 2$ (that is, $\mathcal{K} = \mathbb{F}_2$), and $\bar{V} = \mathcal{K}^{t-1}$ be the $t-1$ dimensional linear space over $\mathcal{K}$. Here the condition "$char(\mathcal{K}) = 2$" is needed in the proof of the following Proposition 1. For the lager secret space (that is $char(\mathcal{K}) > 2$), we can choose a appropriate positive integer h such that $\mathbb{F}_{2^h} = \mathcal{K}$ is satisfied and then by using our scheme, per bit sharing the secrets can be realized. Obviously, it is still an ideal LMSSS and parallel processing will succeed.

Suppose that $\vec{e}_0 = (0, \ldots, 0) \in \mathcal{K}^{t-1}$ and $\vec{e}_j = (0, \ldots, 0, \overset{j}{1}, 0, \ldots 0) \in \mathcal{K}^{t-1}$for $1 \leq j \leq t-1$. We can associate each vertex v_k with a $(t-1)$-dimensional vector $\vec{e}_k$, where $0 \leq k \leq t-1$. For any $1 \leq i \leq n$, suppose that $f^{-1}(i) = v_p v_q$, where $p \neq q$ and $0 \leq p, q \leq t-1$, and then let $\vec{u}_i = \vec{e}_p + \vec{e}_q$ be the $(t-1)$-dimensional vector associated with the player i and $V_i = span\{\vec{u}_i\}$.

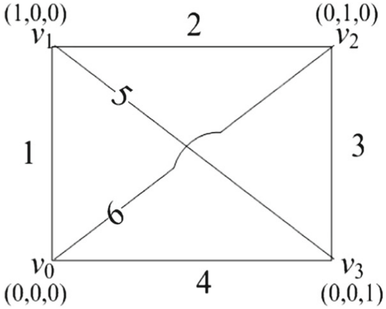

Fig. 2 Associate each vertex with a vector

Let $\vec{v}_j = \vec{u}_j, 1 \leq j \leq m$ be the m target vectors and $\vec{u}_i$ be the row vector distributed to player i for $1 \leq i \leq n$, where $m = n = t(t-1)/2$. We can build an MSP $\mathcal{M}(\mathcal{K}, M, \psi)$ computing $f_{\Gamma_1}, \ldots, f_{\Gamma_m}$, where M is a $n \times (t-1)$ matrix over $\mathcal{K}$ with the ith row vector $\vec{u}_i$, (that is, $\psi(i) = i$ for $1 \leq i \leq n$) and $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ is defined according to (1).

Example 2 (following Example 1) Let $\bar{V} = \mathcal{K}^3$. Select $\vec{e}_0 = (0, 0, 0)$, $\vec{e}_1 = (1, 0, 0)$, $\vec{e}_2 = (0, 1, 0)$, $\vec{e}_3 = (0, 0, 1)$. Associate vertex v_0 with $\vec{e}_0$, vertex v_1 with $\vec{e}_1$, vertex v_2 with $\vec{e}_2$, vertex v_3 with $\vec{e}_3$ (see Fig. 2). We obtain that

$$V_1 = span\{(1, 0, 0)\}, V_2 = span\{(1, 1, 0)\}, V_3 = span\{(0, 1, 1)\},$$
$$V_4 = span\{(0, 0, 1)\}, V_5 = span\{(1, 0, 1)\}, V_6 = span\{(0, 1, 0)\}.$$

(2) The distribution phase.

The dealer first randomly selects a vector $\vec{r} \in \mathcal{K}^{t-1}$ such that the inner product $(\vec{v}_j, \vec{r}) = s_j, 1 \leq j \leq m$. Then he computes $M\vec{r}^{\tau}$ and transmits $M_i\vec{r}^{\tau}$ to player i $(1 \leq i \leq n)$, where "τ" is the transpose and M_i denotes the matrix M restricted to the row i with $\psi(i) = i$. Thus, each player i $(1 \leq i \leq n)$ gets the share $M_i\vec{r}^{\tau}$.

(3) The reconstruction phase.

For any $A \in \Gamma_j$ $(1 \leq j \leq m)$, since $\vec{v}_j \in \sum_{i \in A} V_i$, there exists a vector $\vec{w}$ such that $\vec{v}_j = \vec{w}M_A$. So $s_j = (\vec{v}_j, \vec{r}) = \vec{v}_j \cdot \vec{r}^{\tau} = (\vec{w}M_A)\vec{r}^{\tau} = \vec{w}(M_A\vec{r}^{\tau})$, that is, the players in A can reconstruct the secret s_j by computing a linear combination of their shares.

Obviously, the linear multi-secret sharing scheme that we build is ideal. Namely, the lengths of all secrets and shares are equal to one bit.

Example 3 (following Example 2) Suppose that $\mathcal{M}(\mathcal{K}, M, \psi)$ is the monotone span program constructed as above and the six target vectors are

$$\vec{v}_1 = \vec{u}_1 = (1, 0, 0),\ \vec{v}_2 = \vec{u}_2 = (1, 1, 0),\ \vec{v}_3 = \vec{u}_3 = (0, 1, 1),$$
$$\vec{v}_4 = \vec{u}_4 = (0, 0, 1),\ \vec{v}_5 = \vec{u}_5 = (1, 0, 1),\ \vec{v}_6 = \vec{u}_6 = (0, 1, 0).$$

Note that

$$M = \begin{pmatrix} \vec{u}_1 \\ \vec{u}_2 \\ \vec{u}_3 \\ \vec{u}_4 \\ \vec{u}_5 \\ \vec{u}_6 \end{pmatrix},$$

and $\psi(i) = i$ for $1 \leq i \leq 6$. Then the distribution of six shares and the reconstruction of six secrets are completed as above.

5 Correctness and Security Proof

We now prove that our scheme is a perfect multi-secret sharing scheme. Firstly, we prove the following Proposition.

Proposition 1 *Suppose that* $\vec{\Gamma} = (\Gamma_1, \ldots, \Gamma_m)$ *is defined by* (1) *and* $\mathcal{K}, \vec{u}_i, V_i, 1 \leq i \leq n$ *are given as above. For any* $1 \leq j \leq m$, *let* $\vec{v}_j = \vec{u}_j$, *then it holds that* $\vec{v}_j \in \bigcap_{A\in(\Gamma_j)_{\min}} \sum_{i\in A} V_i - \bigcup_{B\in(\mathcal{A}_j)_{\max}} \sum_{i\in B} V_i$.

Proof Firstly, we prove that $\vec{v}_j \in \bigcap_{A\in(\Gamma_j)_{\min}} \sum_{i\in A} V_i$ for any $1 \leq j \leq m$. For any $A \in (\Gamma_j)_{\min}$, suppose that $f^{-1}(j) = v_p v_q$. According to (1), it implies that there must exist a path $v_{i_0} v_{i_1} \ldots v_{i_k}$ from v_p to v_q in $G(V, E_A)$, where $v_{i_0} = v_p, v_{i_k} = v_q (0 \leq i_0, \ldots, i_k \leq t-1)$. Namely, $\vec{u}_j = \vec{e}_p + \vec{e}_q = \vec{e}_{i_0} + \vec{e}_{i_k}$. We assume that $f(v_{i_l} v_{i_{l+1}}) = h_l (0 \leq l \leq k-1)$, where $h_l \in A$ and $\vec{u}_{h_l} = \vec{e}_{i_l} + \vec{e}_{i_{l+1}}$. We obtain that

$$\vec{u}_{h_0} + \vec{u}_{h_1} + \cdots + \vec{u}_{h_{k-1}} = (\vec{e}_{i_0} + \vec{e}_{i_1}) + (\vec{e}_{i_1} + \vec{e}_{i_2}) + \cdots + (\vec{e}_{i_{k-1}} + \vec{e}_{i_k}) = \vec{e}_{i_0} + \vec{e}_{i_k} = \vec{u}_j.$$

Since $V_i = span\{\vec{u}_i\}(i \in \mathcal{P})$, it implies that for any $A \in (\Gamma_j)_{\min}$, there exists a linear combination of the vectors in $\sum_{i\in A} V_i$ such that it equals to $\vec{v}_j = \vec{u}_j$, where $1 \leq j \leq m$. Namely, $\vec{v}_j = \vec{u}_j \in \sum_{i\in A} V_i$ for every $A \in (\Gamma_j)_{\min}$. Hence, we obtain that $\vec{v}_j \in \bigcap_{A\in(\Gamma_j)_{\min}} \sum_{i\in A} V_i$ for any $1 \leq j \leq m$. □

Then we prove that $\vec{v}_j \notin \bigcup_{B\in(\mathcal{A}_j)_{\max}} \sum_{i\in B} V_i$ for any $1 \leq j \leq m$. For every $B \in (\mathcal{A}_j)_{\max}$, suppose that $f^{-1}(j) = v_p v_q$. According to (1), it implies that there does not exist a path from v_p to v_q in $G(V, E_B)$. We assume that there exists a linear combination of the vectors in $\sum_{i\in B} V_i$ such that it equals to $\vec{v}_j = \vec{u}_j$. Namely, $\vec{u}_{h_0} + \vec{u}_{h_1} + \cdots + \vec{u}_{h_g} = \vec{u}_j = \vec{e}_p + \vec{e}_q$, where $h_l \in B(0 \leq l \leq g)$. Suppose that $f(h_l) = v_{x_l} v_{y_l}$. We obtain that

$$\vec{e}_p + \vec{e}_q = (\vec{e}_{x_0} + \vec{e}_{y_0}) + \cdots + (\vec{e}_{x_g} + \vec{e}_{y_g}), \tag{2}$$

where $\vec{e}_{x_l}, \vec{e}_{y_l} \in \{\vec{e}_0, \vec{e}_1, \ldots, \vec{e}_{t-1}\} (0 \leq l \leq g)$. Due to the fact that $\vec{e}_0, \vec{e}_1, \ldots, \vec{e}_{t-1}$ are linearly independent and $\mathcal{K} = \mathrm{F}_2$, we obtain that the number of times $\vec{e}_p$ and $\vec{e}_q$ appears on the right side of (2) must be an odd number, but the number of times $\vec{e}_r (1 \leq r \leq t-1$ and $r \neq p, q)$ appears on the right side of (2) must be an even number. Thus, the Eq. (2) actually determines a path from v_p to v_q in $G(V, E_B)$. This is a contradiction. We obtain that for every $B \in (\mathcal{A}_j)_{\max}$, there does not exist a linear combination of the vectors in $\sum_{i\in B} V_i$ such that it equals to $\vec{v}_j = \vec{u}_j$, where $1 \leq j \leq m$. Namely, $\vec{v}_j = \vec{u}_j \notin \sum_{i\in B} V_i$ for every $B \in (\mathcal{A}_j)_{\max}$. Hence, $\vec{v}_j \notin \bigcup_{B\in(\mathcal{A}_j)_{\max}} \sum_{i\in B} V_i$ for any $1 \leq j \leq m$.

As a consequence of the above, it holds that

$$\vec{v}_j \in \bigcap\nolimits_{A\in(\Gamma_j)_{\min}} \sum\nolimits_{i\in A} V_i - \bigcup\nolimits_{B\in(\mathcal{A}_j)_{\max}} \sum\nolimits_{i\in B} V_i$$

for any $1 \leq j \leq m$.

Theorem 1 *The scheme presented in Sect.* 4 *is a perfect multi-secret sharing scheme.*

Proof From Proposition 1, for $1 \leq j \leq m$, seeing that $\vec{v}_j \in \bigcap_{A\in(\Gamma_j)_{\min}} \sum_{i\in A} V_i$, it implies that any subset $A \in \Gamma_j$ of players can reconstruct the secret s_j by computing a linear combination of their shares. Hence, it holds that $H(S_j | A) = 0$. □

At the same time, from Proposition 1, for any $1 \leq j \leq m$, seeing that $\vec{v}_j = \vec{u}_j \notin \bigcup_{B \in (\mathcal{A}_j)_{\max}} \sum_{i \in B} V_i$, we obtain that there does not exist a linear combination of their shares such that it equals to s_j. It implies that any subset $B \notin \Gamma_j$ (namely, $B \in \mathcal{A}_j$) of players, even knowing some of other secrets (that is, they can obtain some of other secrets by computing a linear combination of their shares), has no more information on s_j than that already conveyed by the known secrets. Hence, it holds that $H(S_j | BS_B) = H(S_j | S_B)$, where S_B denotes the secrets that B can compute.

Therefore, according to Definition 1, the scheme is a perfect multi-secret sharing scheme.

As a consequence, our scheme is an ideal and perfect linear multi-secret sharing scheme.

Remark 1 In a verifiable secret sharing scheme the validity of the shares can be verified, hence players are not able to cheat. Based on our scheme, we can further construct an ideal verifiable multi-secret sharing scheme by adding the existing verifiability methods where the intractability of discrete logarithm problem is frequently used (see [6,7,9,10]).

6 Conclusions

In this paper we consider an ideal linear multi-secret sharing scheme based on connectivity of graphs. For a set of players $\mathcal{P} = \{1, \ldots, n\}$, every pair of vertices, p and q, use the related secret s_{pq} to communicate with each other such that a set of servers (edges) can recover s_{pq} if these edges can provide an alternative comnnection between p and q. In particular, we put forward a general and simple method of construction such a scheme based on monotone span programs. The correctness and security of the proposed scheme are proved. This scheme is suitable for secure and efficient communications.

References

1. Shamir, A. (1979). How to share a secret. *Communications of the ACM*, *22*(11), 612–613.
2. Blakley, G. R. (June 1979). Safeguarding cryptographic keys. In *Proceedings AFIPS 1979 national computer conference* (pp. 313–317).
3. Karnin, E. D., Greene, J. W., & Hellman, M. E. (1983). On secret sharing systems. *IEEE Transactions on Information Theory*, *29*(1), 35–41.
4. Simmons, G. J. (1991). An introduction to shared secret and/or shared control schemes and their applications. *Contemporary Cryptology*, 441–497; IEEE Press, New York.
5. Chan, C. W., & Chang, C. C. (2005). A scheme for threshold multi-secret sharing. *Applied Mathematics and Computation*, *166*(1), 1–14.
6. Dehkordi, M. H., & Mashhadi, S. (2008). An efficient threshold verifiable multi-secret sharing. *Computer Standards & Interfaces*, *30*(3), 187–190.
7. Dehkordi, M. H., & Mashhadi, S. (2008). New efficient and practical verifiable multi-secret sharing schemes. *Information Sciences*, *178*(9), 2262–2274.
8. Pang, L. J., & Wang, Y. M. (2005). A new (t, n) multi-secret sharing scheme based on Shamir's secret sharing. *Applied Mathematics and Computation*, *167*, 840–848.
9. Shao, J., & Cao, Z. F. (2005). A new efficient (t, n) verifiable multi-secret sharing (VMSS) based on YCH scheme. *Applied Mathematics and Computation*, *168*, 135–140.
10. Zhao, J., Zhang, J., & Zhao, R. (2007). A practical verifiable multi-secret sharing scheme. *Computer Standards & Interfaces*, *29*(1), 138–141.
11. Jackson, W.-A., Martin, K. M., & O'Keefe, C. M. (1994). Multisecret threshold schemes. In *Advances in cryptology—CRYPTO '93, Lecture Notes in Computer Science* (Vol. 773, pp. 126–135). Berlin: Springer.
12. Jackson, W.-A., Martin, K. M., & O'Keefe, C. M. (1996). A construction for multisecret threshold schemes. *Design, Codes and Cryptography*, *9*(3), 287–303.

13. Blundo, C., De Santis, A., & Vaccaro, U. (1993). Efficient sharing of many secrets. In *Proceedings of STACS '93 (10th symposium on theoretical aspects of computer science), Lecture Notes in Computer Science* (Vol. 665, pp. 692–703). Berlin: Springer.
14. Jackson, W.-A., Martin, K. M., & O'Keefe, C. M. (1995). On sharing many secrets. In *Advances in cryptology—ASIACRYPT '94, Lecture Notes in Computer Science* (Vol. 917, pp. 42–54). Berlin: Springer.
15. Blundo, C., De Santis, A., Di Crescenzo, G., Giorgio Gaggia, A., Vaccaro, U. (1994). Multi-secret sharing schemes. In *Advances in cryptology—CRYPTO '94, Lecture Notes in Computer Science* (Vol. 839, pp. 150–163). Berlin: Springer.
16. Jackson, W.-A., Martin, K. M., & O'Keefe, C. M. (1996). Ideal secret sharing schemes with multiple secrets. *Journal of Cryptology*, *9*, 233–250.
17. Liu, M., Xiao, L., & Zhang, Z. (2006). Linear multi-secret sharing schemes based on multi-party computation. *Finite Fields and Their Applications*, *12*, 704–713.
18. Xiao, L., & Liu, M. (2005). Linear multi-secret sharing schemes. *Science in China Series F: Information Sciences*, *48*(1), 125–136.
19. Hsu, C. F., Cui, G. H., Cheng, Q., & Chen, J. (2011). A novel linear multi-secret sharing scheme for group communication in wireless mesh networks. *Journal of Network and Computer Applications*, *34*(2), 464–468.
20. Hsu, C. F., Cheng, Q., Tang, X. M., & Zeng, B. (2011). An ideal linear multi-secret sharing scheme based on MSP. *Information Sciences*, *181*(7), 1403–1409.
21. Stinson, D. R. (1992). An explication of secret sharing schemes. *Design, Codes, and Cryptography*, *2*, 357–390.
22. Beimel, A. (1996). *Secure schemes for secret sharing and key distribution*, Ph.D. dissertation. Technion–Israel Inst. Technol., Haifa, Israel.
23. Karchmer, M., & Wigderson, A. (May 1993). On span programs. In *Proceedings of the 8th annual conference structure in complexity* (pp. 102–111), San Diego, CA.
24. De Santis, A., & Masucci, B. (1999). Multiple ramp schemes. *IEEE Transactions on Information Theory*, *45*(5), 1720–1728.

Author Biographies

Ching-Fang Hsu was born in Hubei, China, on Nov. 22, 1978. She received the M.Eng. and the Ph.D. degrees in information security from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2010, respectively. From Sept. 2010 to Mar. 2013, she was a Research Fellow at the Huazhong University of Science and Technology. She is currently an Assistant Professor at Central China Normal University, Wuhan, China. Her research interests are in cryptography and network security, especially in secret sharing and its applications.

Lein Harn received the B.S. degree in electrical engineering from the National Taiwan University in 1977, the M.S. degree in electrical engineering from the State University of New York-Stony Brook in 1980, and the Ph.D. degree in electrical engineering from the University of Minnesota in 1984. In 1984, he joined the Department of Electrical and Computer Engineering, University of Missouri-Columbia as an assistant professor, and in 1986, he moved to Computer Science and Telecommunication Program (CSTP), University of Missouri, Kansas City (UMKC). While at UMKC, he went on development leave to work in Racal Data Group, Florida for a year. His research interests include cryptography, network security, and wireless communication security. He has published a number of papers on digital signature design and applications and wireless and network security. He has written two books on security. He is currently investigating new ways of using secret sharing in various applications.

Guohua Cui born in 1947, professor, Ph.D. supervisor, His research interests include information and network security.