A New Metric and the Construction for Evolving 2-Threshold Secret Sharing Schemes Based on Prefix Coding of Integers

Wei Yan, *Member, IEEE*, Sian-Jheng Lin, *Member, IEEE* and Yunghsiang S. Han, *Fellow, IEEE*

***Abstract*—Evolving secret sharing schemes do not require prior knowledge of the number of parties n, which may be infinitely countable. It is known that the evolving 2-threshold secret sharing scheme and prefix coding of integers have a one-to-one correspondence. However, it is unknown what prefix coding of integers should be used to construct a better secret sharing scheme. In this paper, we introduce a metric K_Σ to evaluate evolving 2-threshold secret sharing schemes Σ such that a smaller K_Σ of a scheme is better. The metric K_Σ is related to the ratio of the sum of the share sizes for the first n parties in scheme Σ and the sum of the share sizes for the optimal $(2, n)$-threshold secret sharing scheme. Then we prove that the metric $K_\Sigma \geq 1.5$ and construct a new prefix coding of integers, termed λ code, to achieve the metric $K_\Lambda = 1.59375$. Thus, this shows that the range of the metric K_Σ for the optimal $(2, \infty)$-threshold secret sharing scheme is $1.5 \leq K_\Sigma \leq 1.59375$. In addition, an achievable lower bound on the sum of share sizes for $(2, n)$-threshold secret sharing schemes is also provided.**

***Index Terms*—evolving secret sharing, universal coding of integers, prefix coding of integers, global metric.**

I. Introduction

The secret sharing scheme was first proposed independently by Shamir [1] and Blakley [2] in 1979. To store a sensitive secret safely, the secret sharing scheme encodes the secret into n shares, and each share is assigned to a party. Some specific subsets of n parties are set as qualified subsets, and others as unqualified subsets. When m parties form a qualified subset, they can use their own shares to recover the sensitive secret. Conversely, when m parties form an unqualified subset, they cannot recover the secret or even obtain any information about the secret. Secret sharing has been applied in widespread applications, such as verifiable signature sharing [3], threshold digital signatures [4], [5], and electronic voting [6].

Shamir [1] and Blakley [2] proposed (t, n)-threshold secret sharing, which means that any t parties among n parties form a qualified subset. General secret sharing schemes were introduced by Ito *et al.* [7]. Traditional secret sharing schemes assume that the number of parties n is known in advance, or the upper bound of n can be estimated. However, this assumption carries the following potential costs. When the estimated n is too small, secret sharing shall be re-made; when the estimated n is too large, it may cause waste.

Recently, an evolving secret sharing scheme was introduced by Komargodski *et al.* [8], [9]. The evolving secret sharing does not require prior knowledge of the upper bound of n, and n may be infinitely countable. Komargodski *et al.* found a one-to-one correspondence between the evolving 2-threshold secret sharing scheme and prefix coding of integers, where 2-threshold means that any $m \geq 2$ parties form a qualified subset and any single party forms an unqualified subset. In 2018, D'Arco *et al.* [10] reinterpreted the equivalence between the evolving 2-threshold secret sharing scheme and prefix coding of integers with a new perspective. In 2020, Okamura and Koga [11] extended the shared secret from 1-bit to any ℓ-bit based on the work of D'Arco *et al.*, and by combining Shamir's secret sharing scheme, proposed an evolving 2-threshold secret sharing scheme using D-ary prefix codes. In addition, there are studies on evolving secret sharing schemes for dynamic thresholds and robustness [12], probabilistic evolving secret sharing [13], and evolving ramp secret sharing [14], [15].

Although the evolving 2-threshold secret sharing scheme can be completely characterized by the prefix coding of integers, we do not know which prefix code should be chosen to improve the secret sharing scheme's performance. Precisely, we need a metric to determine which evolving 2-threshold secret sharing scheme constructed by the prefix coding of integers performs better. Although the previous work did not propose a clear metric for the evolving 2-threshold secret sharing, some works [9], [11] focused on the asymptotic performance of the evolving 2-threshold secret sharing, where asymptotic performance focuses on the situation of the secret sharing scheme when the number of parties n tends to infinity and requires that the share distributed at a sufficiently large moment be as small as possible. However, in the problem setting and actual scenario, the number of parties n is unknown and does not necessarily tend to infinity. Therefore, it is relatively one-sided to focus only on cases where n tends to infinity.

Universal coding of integers [16], [17], [18] is a subclass of prefix coding of integers. The most famous universal coding of integers, the Elias codes [16], is applied to the evolving 2-

This work was supported in part by the National Natural Science Foundation of China under Grant 62071446 and in part by the National Key Research and Development Program of China under Grant 2022YFA1004902.(Corresponding author: Yunghsiang S. Han.)

Wei Yan was with the CAS Key Laboratory of Electromagnetic Space Information, School of Cyber Science and Technology, University of Science and Technology of China, Hefei, China. He is now with the Theory Laboratory, Huawei Technologies Company Ltd., Hefei, China (email:yan1993@mail.ustc.edu.cn).

Sian-Jheng Lin is with the Theory Laboratory, Huawei Technologies Company Ltd., Hong Kong, China (email:lin.sian.jheng1@huawei.com).

Yunghsiang S. Han is with Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China (email: yunghsiangh@gmail.com).

threshold secret sharing scheme [8], [9], [11]. In recent years, the metric, called minimum expansion factor K_σ^* [19], [20], for universal coding of integers has been introduced. The idea of building the minimum expansion factor K_σ^* will be applied in this paper.

In this paper, based on prefix coding of integers, we introduce a metric K_Σ for evolving 2-threshold secret sharing and construct good evolving 2-threshold secret sharing schemes under this metric. Unlike previous works [9], [11] that focused on the asymptotic performance of evolving 2-threshold secret sharing, we consider the global performance, which refers to all cases where the number of parties n is considered. That is, the metric K_Σ is not only suitable for n tending to infinity but also for small n, even $n = 2$ or $n = 3$. K_Σ is the first metric for evolving 2-threshold secret sharing schemes in all cases with the number of parties n. The contributions of this paper are enumerated as follows.

1) An achievable lower bound on the sum of share sizes for $(2, n)$-threshold secret sharing schemes is proved (see Theorem 4).
2) A new metric for evolving 2-threshold secret sharing schemes is introduced (see Definition 8).
3) Under the new metric, evolving 2-threshold secret sharing schemes whose performance is close to the optimal scheme are constructed (see Section V).

The paper is structured as follows. Section II introduces some background knowledge. Section III proves an achievable lower bound on the sum of share sizes for $(2, n)$-threshold secret sharing schemes. Section IV proposes a new metric for evolving 2-threshold secret sharing schemes. Section V constructs evolving 2-threshold secret sharing schemes whose performance is close to the optimal scheme under the new metric. The comparisons are placed in Section VI. Section VII concludes this work.

II. Preliminaries

We first introduce some necessary notations. Let $\mathbb{N}$ be the set of positive integers. Let $\mathbb{B} := \{0, 1\}$, and let $\mathbb{B}^*$ be a set consisting of all finite-length binary strings. $\#S$ denotes the cardinality of the set S. $|\alpha|$ denotes the length of string α. For a positive integer n, let $[n] := \{1, 2, \ldots, n\}$.

A. Secret Sharing Scheme

Let $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ denote the set of participants, and let $2^{\mathcal{P}}$ be the power set of the set $\mathcal{P}$. $\mathcal{M} \subseteq 2^{\mathcal{P}}$ is called *monotone* if for any $S_1 \in \mathcal{M}$ and $S_1 \subseteq S_2$, then $S_2 \in \mathcal{M}$. Before defining secret sharing scheme, we first define the access structure as follows.

Definition 1 *$\mathcal{M} \subseteq 2^{\mathcal{P}}$ is said to be an access structure if $\mathcal{M}$ is non-empty and monotone. Elements in $\mathcal{M}$ are called qualified, and elements not in $\mathcal{M}$ are called unqualified.*

Definition 2 *Let t and n both be positive integers, and $1 \leq t \leq n$. The (t, n)-threshold access structure $\mathcal{M}$ refers to the set containing only all elements in $2^{\mathcal{P}}$ of size at least t, i.e.*

$$\mathcal{M} = \{A \in 2^{\mathcal{P}} \mid \#A \geq t\}.$$

The definition of secret sharing scheme is given based on the access structure.

Definition 3 *A secret sharing scheme Σ for an access structure $\mathcal{M}$ consists of a pair of probabilistic algorithms $(\mathcal{S}, \mathcal{R})$. The sharing algorithm $\mathcal{S}$ generates n shares $sh_{P_1}^{(s)}, sh_{P_2}^{(s)}, \ldots, sh_{P_n}^{(s)}$ according to the secret $s \in S$ and the number of participants n. The recovery algorithm $\mathcal{R}$ outputs a string according to the shares of the subset $A \in 2^{\mathcal{P}}$. The algorithm is required to satisfy:*

- ***Correctness:*** *For every qualified set $A \in \mathcal{M}$ and any secret $s \in S$, the recovery algorithm $\mathcal{R}$ can recover the secret s with probability 1, i.e.*

$$Pr[\mathcal{R}(A, \{sh_j^{(s)}\}_{j \in A}) = s] = 1.$$

- ***Secrecy:*** *Each unqualified set $B \notin \mathcal{M}$ does not get any information about the secret s, that is, for any two different secrets $s_1, s_2 \in S$, each unqualified set $B \notin \mathcal{M}$ and each type of shares SH assigned to B,*

$$\begin{aligned} &Pr(\{sh_j\}_{j \in B} = SH | s = s_1) \\ &= Pr(\{sh_j\}_{j \in B} = SH | s = s_2). \end{aligned}$$

For designing a secret sharing scheme, the goal is to generate the sum of share sizes $\sum_{i=1}^{n} |sh_{P_i}^{(s)}|$ is as small as possible, which can make the amount of communication as small as possible.

The following introduces two important conclusions that will be used later.

Theorem 1 *[21], [22] Suppose that Σ is (t, n)-thresholded secret sharing scheme for 1-bit secret, and the j-th share size is m_j bits, where $2 \leq t \leq n$, $j \in [n]$ and $m_j \in \mathbb{N}$. Then, the sum of share sizes*

$$\sum_{j=1}^{n} m_j \geq n \log_2(n - t + 2).$$

In particular, when $t = 2$, the sum of share sizes for $(2, n)$-threshold secret sharing schemes

$$\sum_{j=1}^{n} m_j \geq n \log_2 n.$$

Lemma 1 *[21], [22], [9] Suppose that Σ is $(2, n)$-thresholded secret sharing scheme for 1-bit secret, and the j-th share size is m_j bits, where $j \in [n]$ and $m_j \in \mathbb{N}$. Then,*

$$\sum_{j=1}^{n} \frac{1}{2^{m_j}} \leq 1.$$

B. Evolving Secret Sharing Scheme

Because the number of participants n is uncertain and the upper bound of n cannot be estimated in real scenarios, a class of secret sharing schemes needs to be defined so that n can be infinitely countable. Naturally, in this scenario, the parties participating in secret sharing will not be present at the same time. We assume that at the t-th moment, the t-th person P_t arrives at the scene and asks for the distribution of subsequent shares. The previously distributed shares should

not be changed, which can effectively reduce the amount of communication. Let $\mathcal{PN} = \{P_1, P_2, \ldots, P_n, \ldots\}$ denote the set of participants. Definitions 1 and 2 are naturally extended to the following definitions.

Definition 4 *[9] Suppose that $\mathcal{M} \subseteq 2^{\mathcal{PN}}$ is monotone, and for each time $t \in \mathbb{N}$, $\mathcal{M}_t := \mathcal{M} \cap 2^{\{P_1, P_2, \ldots, P_t\}}$ is an access structure. Then $\mathcal{M}$ is said to be an evolving access structure.*

Definition 5 *[9] Let m be a positive integer. The evolving m-threshold access structure $\mathcal{M}$ refers to the set consisting only of all elements in $2^{\mathcal{PN}}$ of size at least m, i.e.*

$$\mathcal{M} = \{A \in 2^{\mathcal{PN}} \,|\, \#A \geq m\}.$$

For simplicity, we use (m, ∞)-threshold to represent evolving m-threshold. Now, the formal definition of the evolving secret sharing scheme is as follows.

Definition 6 *[8], [9] Let S denote a domain of secrets, where $\#S \geq 2$. Let $\mathcal{M}$ denote an evolving access structure. An evolving secret sharing scheme Σ for S and $\mathcal{M}$ consists of a pair of probabilistic algorithms $(\mathcal{S}, \mathcal{R})$. The sharing algorithm $\mathcal{S}$ and the recovery algorithm $\mathcal{R}$ are required to satisfy:*

1) *At time $t \in \mathbb{N}$, the sharing algorithm $\mathcal{S}$ generates share $sh_{P_t}^{(s)}$ according to the secret $s \in S$ and shares $sh_{P_1}^{(s)}, sh_{P_2}^{(s)}, \ldots, sh_{P_{t-1}}^{(s)}$, i.e.*

$$\mathcal{S}(s, \{sh_{P_i}^{(s)}\}_{i \in [t-1]}) = sh_{P_t}^{(s)}.$$

2) ***Correctness:*** *For each time $t \in \mathbb{N}$, every qualified set $A \in \mathcal{M}_t$ and any secret $s \in S$, the recovery algorithm $\mathcal{R}$ can recover the secret s with probability 1, i.e.*

$$Pr[\mathcal{R}(A, \{sh_i^{(s)}\}_{i \in A}) = s] = 1.$$

3) ***Secrecy:*** *For each time $t \in \mathbb{N}$, each unqualified set $B \notin \mathcal{M}_t$ does not get any information about the secret s, that is, for each time t, any two different secrets $s_1, s_2 \in S$, each unqualified set $B \notin \mathcal{M}_t$ and each type of shares SH assigned to B,*

$$\begin{aligned} &Pr(\{sh_j\}_{j \in B} = SH | s = s_1) \\ =\ &Pr(\{sh_j\}_{j \in B} = SH | s = s_2). \end{aligned}$$

The most important theorem for $(2, \infty)$-thresholded secret sharing scheme is shown below.

Theorem 2 *[8], [9] Let $\sigma : \mathbb{N} \to \mathbb{B}^*$ be a prefix coding of integers. The length of its t-th codeword is $L_\sigma(t)$, where $t \in \mathbb{N}$. Such an integer code exists if and only if there exists a $(2, \infty)$-threshold secret sharing scheme Σ for 1-bit secret, and the t-th share size is $L_\sigma(t)$ bits.*

Theorem 2 shows the equivalence between the $(2, \infty)$-thresholded secret sharing scheme and prefix coding of integers. An interesting and essential understanding of Theorem 2 can be found in [10].

C. The Minimum Expansion Factor $K_{\mathcal{C}}^*$ for Universal Coding of Integers

Universal coding of integers is a class of binary prefix code, such that the ratio of the expected codeword length to $\max\{1, H(P)\}$ is within a constant for any decreasing probability distribution P of $\mathbb{N}$ (i.e., $\sum_{n=1}^{\infty} P(n) = 1$, and $P(m) \geq P(m+1) \geq 0$ for all $m \in \mathbb{N}$), where $H(P) := -\sum_{n=1}^{\infty} P(n) \log_2 P(n)$ is the entropy of P. The formal definition of universal coding of integers is as follows.

Definition 7 *[16], [19] Let $\sigma : \mathbb{N} \to \mathbb{B}^*$ be a binary prefix coding of integers. Let $L_\sigma(\cdot)$ denote the length function of σ so that $L_\sigma(m) = |\sigma(m)|$ for all $m \in \mathbb{N}$. σ is called universal if there exists a constant K_σ independent of P, such that*

$$\frac{E_P(L_\sigma)}{\max\{1, H(P)\}} \leq K_\sigma, \tag{1}$$

for any decreasing probability distribution P with finite entropy, where

$$E_P(L_\sigma) := \sum_{n=1}^{\infty} P(n) L_\sigma(n)$$

denotes the expected codeword length for σ. Then K_σ is called the expansion factor. Let $K_\sigma^* := \inf\{K_\sigma \,|\, \forall P \text{ and } H(P) < \infty\}$ be the infimum of the set of expansion factors and K_σ^* is called the minimum expansion factor.

The minimum expansion factor K_σ^* is the smallest of the expansion factors. For any universal coding of integers, its minimum expansion factor K_σ^* is unique. The minimum expansion factor K_σ^*, as a metric, evaluate the compression performance of universal coding of integers. Therefore, universal coding of integers σ is called *optimal* if σ achieves the smallest K_σ^*.

Finally, we introduce two classes of universal coding of integers, ι code [20] and η code [19], which can achieve small expansion factors. In particular, ι code is currently the only universal coding of integers that can achieve $K_\iota = 2.5$. The two codes are briefly introduced as follows.

The unary code α of the non-negative integer m is constructed as m bits of 0 followed by a single 1. Let $\beta(m)$ denote the standard binary representation of $m \in \mathbb{N}$, and let $\overline{\beta(m)}$ denote the removal of the most significant bit 1 of $\beta(m)$. For example, $\alpha(2) = 001$, $\beta(10) = 1010$ and $\overline{\beta(10)} = 010$. Note that $\overline{\beta(1)}$ is a null string. We define $\beta(0)$ as a null string, and the length of the null string is 0.

1) The code $\iota : \mathbb{N} \to \mathbb{B}^*$ can be expressed as

$$\iota(m) = \begin{cases} 1, & \text{if } m = 1, \\ \alpha(\frac{|\beta(m)|}{2}) 0 \overline{\beta(m)}, & \text{if } |\beta(m)| \text{ is even}, \\ \alpha(\frac{|\beta(m)|-1}{2}) \beta(m), & \text{otherwise}, \end{cases}$$

for all $m \in \mathbb{N}$. The codeword length is given by

$$\begin{aligned} L_\iota(m) &= 1 + \lfloor \frac{|\beta(m)|}{2} \rfloor + |\beta(m)| \\ &= 2 + \lfloor \frac{1 + \lfloor \log_2 m \rfloor}{2} \rfloor + \lfloor \log_2 m \rfloor \\ &\leq \frac{5}{2} + \frac{3}{2} \lfloor \log_2 m \rfloor, \end{aligned}$$

for $2 \leq m \in \mathbb{N}$ and $L_\iota(1) = 1$.

2) The code $\eta : \mathbb{N} \to \mathbb{B}^*$ can be expressed as

$$\eta(m)=\begin{cases}\alpha(\frac{|\beta(m-1)|}{2})\beta(m-1), \\ \qquad\qquad \text{if } |\beta(m-1)| \text{ is even}, \\ \alpha(\frac{1+|\beta(m-1)|}{2})0\overline{\beta(m-1)}, \text{ otherwise},\end{cases}$$

for all $m \in \mathbb{N}$. The codeword length is given by

$$\begin{aligned}L_\eta(m) &= 1+\lfloor\frac{1+|\beta(m-1)|}{2}\rfloor+|\beta(m-1)| \\ &= 3+\lfloor\frac{\lfloor\log_2(m-1)\rfloor}{2}\rfloor+\lfloor\log_2(m-1)\rfloor \\ &\le 3+\frac{3}{2}\lfloor\log_2(m-1)\rfloor,\end{aligned}$$

for $2 \le m \in \mathbb{N}$ and $L_\eta(1) = 1$.

III. An Achievable Lower Bound on the Sum of Share Sizes for $(2, n)$-Threshold Secret Sharing Schemes

In this section, an achievable lower bound on the sum of share sizes for $(2, n)$-threshold secret sharing schemes is proved. For simplicity, an achievable lower bound on the sum of share sizes for (t, n)-threshold secret sharing schemes is called the capacity of (t, n)-threshold secret sharing schemes. Let $\min[sum(2, n)]$ denote the minimum sum of share sizes in all $(2, n)$-threshold secret sharing schemes. Thus, $\min[sum(2, n)]$ is the capacity of $(2, n)$-threshold secret sharing schemes. The *optimal* $(2, n)$*-threshold secret sharing scheme* is defined as the secret sharing scheme that achieves capacity.

In the early 1990s, Kilian and Nisan first proposed and proved Theorem 1 in an email, but it was not officially published. This unpublished result has been mentioned in numerous papers and the proof by Kilian and Nisan was first published in [22]. Theorem 1 shows that the sum of share sizes for $(2, n)$-threshold secret sharing schemes is greater than or equal to $n \log_2 n$, i.e.

$$\min[sum(2, n)] \ge n \log_2 n.$$

This section gives a tighter bound than the lower bound $n \log_2 n$ and this bound is achievable, that is, this section gives the exact expression for $\min[sum(2, n)]$. First, a result similar to Theorem 2 is given.

Theorem 3 *Let $\sigma : [n] \to \mathbb{B}^*$ be a prefix code. The length of its t-th codeword is $L_\sigma(t)$, where $t \in [n]$. Such a prefix code exists if and only if there exists a $(2, n)$-threshold secret sharing scheme Σ for 1-bit secret, and the t-th share size is $L_\sigma(t)$ bits.*

Proof ($\Longleftarrow$) *Suppose that there exists a $(2, n)$-threshold secret sharing scheme Σ for 1-bit secret, and the t-th share size is $L_\sigma(t)$ bits. From Lemma 1, we obtain*

$$\sum_{j=1}^{n} \frac{1}{2^{L_\sigma(j)}} \le 1.$$

Due to Kraft's inequality [23], we know that there exists a prefix code with codeword lengths $L_\sigma(1), L_\sigma(2), \ldots, L_\sigma(n)$.

($\Longrightarrow$) *Suppose that there is a prefix code with codeword lengths $L_\sigma(1), L_\sigma(2), \ldots, L_\sigma(n)$. Next, we construct a $(2, n)$-threshold secret sharing scheme Σ for 1-bit secret. Let $s \in \mathbb{B}$ be the secret and M denote the maximum value among $L_\sigma(1), L_\sigma(2), \ldots, L_\sigma(n)$. Initially, the dealer randomly generates a binary string Q of length M. Let $Q|_t$ denote the first $L_\sigma(t)$ bits of the string Q. If $s = 0$, then the t-th share is $sh(t) = \sigma(t) \oplus Q|_t$; If $s = 1$, then the t-th share is $sh(t) = Q|_t$. The t-th share size is $L_\sigma(t)$ bits.*

In the recovery secret stage, let the two different shares be $sh(t_1)$ and $sh(t_2)$. Without loss of generality, we assume that $|sh(t_1)| \le |sh(t_2)|$. If $sh(t_1)$ is a prefix of $sh(t_2)$, the output is 1; otherwise, the output is 0.

(2.1) ***Correctness:*** *If $s = 0$, then $sh(t_1) = \sigma(t_1) \oplus Q|_{t_1}$ and $sh(t_2) = \sigma(t_2) \oplus Q|_{t_2}$. Since $Q|_{t_1}$ is a prefix of $Q|_{t_2}$ and $\sigma(t_1)$ is not a prefix of $\sigma(t_2)$, then $sh(t_1)$ is not a prefix of $sh(t_2)$. Therefore, 0 is correctly output. If $s = 1$, then $sh(t_1) = Q|_{t_1}$ and $sh(t_2) = Q|_{t_2}$. Since $Q|_{t_1}$ is a prefix of $Q|_{t_2}$, 1 is correctly output.*

(2.2) ***Secrecy:*** *Because $Q|_t$ is uniformly distributed on $\mathbb{B}^{L_\sigma(t)}$, whether $s = 0$ or $s = 1$, there is a share $sh(t)$ uniformly distributed on $\mathbb{B}^{L_\sigma(t)}$. Therefore, for any single party A and each string $SH \in \mathbb{B}^{L_\sigma(t)}$,*

$$\begin{aligned}&Pr(sh(t) = SH|s = 0) = \frac{1}{2^{L_\sigma(t)}} \\ &= Pr(sh(t) = SH|s = 1).\end{aligned}$$

From Theorem 2, we know that there is a one-to-one correspondence between the $(2, \infty)$-thresholded secret sharing scheme and the prefix coding of integers. Theorem 3 shows a one-to-one correspondence between the $(2, n)$-thresholded secret sharing scheme and the prefix code with n codewords. Theorem 2 can be viewed as the case where the number of codewords n tends to infinity in Theorem 3. Next, the main theorem of this section is given.

Theorem 4 *Let n be an integer greater than 1, then*

$$\min[sum(2, n)] = nm + 2l,$$

where $m := \lfloor \log_2 n \rfloor$ and $l := n - 2^m$.

Proof *Suppose that Σ is a $(2, n)$-threshold secret sharing scheme for 1-bit secret, and the j-th share size is m_j bits, where $j \in [n]$. We need to find a scheme Σ that minimizes the sum $\sum_{j=1}^{n} m_j$, that is, find the minimum value $\min[sum(2, n)]$ of $\sum_{j=1}^{n} m_j$.*

Due to Theorem 3, Σ corresponds to a prefix code $\sigma : [n] \to \mathbb{B}^*$, and the length of its j-th codeword is m_j, where $j \in [n]$. To make $\sum_{j=1}^{n} m_j$ minimum is equivalent to minimizing the expected codeword length $L = \sum_{j=1}^{n} P(j)m_j = \frac{1}{n}\sum_{j=1}^{n} m_j$ of the prefix code σ with probability distribution

$$P = \left(P(1) = P(2) = \cdots = P(n) = \frac{1}{n}\right).$$

Because given a probability distribution, Huffman code is the prefix code with the smallest expected codeword length [24]. Therefore, it is only necessary to calculate the expected codeword length when encoding with Huffman code in the case of the probability distribution P. From P is a uniform

distribution and the encoding rule of Huffman code [24], it can be observed that the code tree obtained by encoding is a full binary tree and the layers of leaf nodes differ by at most 1*, so the lengths of all codewords differ by at most* 1*. Then, there are* $2^m - l$ *codewords with codeword length* m *and* $2l$ *codewords with codeword length of* $m+1$*. Therefore, the minimum value of* $\sum_{j=1}^{n} m_j$ *is*

$$\begin{aligned}\min[sum(2,n)] &= m(2^m - l) + 2l(m+1)\\ &= m2^m + ml + 2l\\ &= nm + 2l.\end{aligned}$$

From Theorem 4, $nm+2l$ is the capacity of $(2,n)$-threshold secret sharing scheme. Since $n\log_2 n$ is the lower bound, there must be $nm + 2l \geq n\log_2 n$. To show that our results are novel, we compare $nm + 2l$ and $n\log_2 n$ only from a purely mathematical point of view.

Lemma 2 *For any* $n = 2^m + l \in \mathbb{N}$*, where* $m = \lfloor \log_2 n \rfloor$*, we have*

$$nm + 2l \geq n\log_2 n. \tag{2}$$

Proof *Let* $x := \log_2 n$ *and* $y := x - \lfloor x \rfloor$*, then* $0 \leq y < 1$*. We obtain*

$$\begin{aligned}&nm + 2l \geq n\log_2 n\\ \Longleftrightarrow\ &n\lfloor \log_2 n \rfloor + 2n - 2\cdot 2^{\lfloor \log_2 n \rfloor} \geq n\log_2 n\\ \Longleftrightarrow\ &n(2 + \lfloor \log_2 n \rfloor - \log_2 n) \geq 2\cdot 2^{\lfloor \log_2 n \rfloor}\\ \Longleftrightarrow\ &2^x(2 + \lfloor x \rfloor - x) \geq 2\cdot 2^{\lfloor x \rfloor}\\ \Longleftrightarrow\ &2^y(2-y) \geq 2.\end{aligned}$$

Therefore, it is equivalent to proving that $2^y(2-y) \geq 2$ *for* $y \in [0,1)$*. Let* $f(y) := 2^y(2-y)$*. By calculating the derivative, we know that* $f(y)$ *is strictly monotonically increasing over the interval* $[0, y_0)$ *and strictly monotonically decreasing over the interval* $[y_0, 1)$*, where* $y_0 = 2 - \dfrac{1}{\ln 2}$*. Therefore, for* $y \in [0,1)$*, we have*

$$f(y) \geq \min\{f(0), f(1)\} = 2.$$

From the proof process of Lemma 2, we know that when $y = 0$, i.e., n is a power of 2, inequality given in (2) becomes equal. Due to the monotonicity of the function $f(y)$, it is reasonable to guess that at the midpoint $2^m + 2^{m-1}$ between 2^m and 2^{m+1} (i.e., the average of two adjacent powers of 2), the difference between the two sides of (2) is large. Thus, Table I lists some values comparing $nm + 2l$ and $n\log_2 n$. Table I confirms that, when $n = 2^m$, $nm + 2l = m\cdot 2^m = n\log_2 n$. In addition, it can be found that the difference at $2^m + 2^{m-1}$ is larger when m is larger. Interestingly, the difference at $2^m + 2^{m-1}$ is 2 times that at $2^{m-1} + 2^{m-2}$. This finding is verified below.

Let $n_1 := 2^m + 2^{m-1}$ and $n_2 := 2^{m-1} + 2^{m-2}$. Let $d(n) := nm + 2l - n\log_2 n$, we obtain

$$\begin{aligned}d(n_1) &= n_1 m + 2^m - n_1\log_2 n_1\\ &= 2n_2 m + 2^m - 2n_2\log_2(2n_2)\\ &= 2n_2(m-1) + 2^m - 2n_2\log_2 n_2\\ &= 2d(n_2).\end{aligned}$$

TABLE I: Comparison of some values of $nm+2l$ and $n\log_2 n$

n	$nm+2l$	$n\log_2 n$	difference
2	2	2	0
4	8	8	0
8	24	24	0
16	64	64	0
32	160	160	0
64	384	384	0
2^7	896	896	0
2^8	2048	2048	0
2^9	4608	4608	0
2^{10}	10240	10240	0
2^{11}	22528	22528	0
2^{12}	49152	49152	0
2^{13}	106496	106496	0
2^{14}	229376	229376	0
3	5	4.75	0.25
6	16	15.51	0.49
12	44	43.02	0.98
24	112	110.04	1.96
48	272	268.08	3.92
96	640	632.16	7.84
$2^7 + 2^6$	1472	1456.31	15.69
$2^8 + 2^7$	3328	3296.63	31.37
$2^9 + 2^8$	7424	7361.25	62.75
$2^{10} + 2^9$	16384	16258.50	125.50
$2^{11} + 2^{10}$	35840	35589.00	251.00
$2^{12} + 2^{11}$	77824	77322.01	501.99
$2^{13} + 2^{12}$	167936	166932.02	1003.98
$2^{14} + 2^{13}$	360448	358440.04	2007.96

Note: After rounding $n\log_2 n$, retain two decimal places.

This shows that when m tends to infinity, the difference at $2^m + 2^{m-1}$ tends to infinity. Therefore, the achievable lower bound $nm + 2l$ proved in this paper is not only a new result but also very meaningful.

IV. A New Metric for $(2,\infty)$-Threshold Secret Sharing Schemes

For simplicity, integer codes mentioned in this paper refer to the prefix coding of integers. Theorem 2 shows that there is a one-to-one correspondence between the $(2,\infty)$-thresholded secret sharing scheme and integer codes. In this section, we strive to formulate a metric that can be used to determine which integer code is more suitable for constructing $(2,\infty)$-thresholded secret sharing schemes.

In the traditional secret sharing, since shares are distributed at one time, the metric to measure is the sum of share sizes, and the smaller sum is better. In the evolving secret sharing, the specific number of parties participating in secret sharing is not known, and even the number of parties may be infinitely countable. Therefore, it is impossible to calculate the sum of share sizes by distributing all the shares at one time.

In this paper, the metrics we focus on are for the global performance rather than the asymptotic performance of $(2,\infty)$-thresholded secret sharing. Two approaches are discussed first.

A. Two approaches

A simple approach is to construct a $(2,\infty)$-threshold secret sharing scheme so that the share length generated at any t time is the smallest. Unfortunately, such a scheme does not exist.

From Theorem 2, for any $(2,\infty)$-thresholded secret sharing scheme Σ, there exists an integer code σ, such that the t-th

codeword length $L_\sigma(t)$ is exactly the size of share distributed by the scheme Σ at the t-th moment. From the theory of prefix codes, it is impossible to have a complete integer code σ that, for any complete integer code ψ and any positive integer t, $L_\sigma(t) \leq L_\psi(t)$, where an integer code is called *complete* if the integer code makes Kraft's inequality [23] equal. The reason is as follows.

Lemma 3 *Considering two different complete integer codes ψ and σ, if t is sufficiently large and $L_\sigma(t) \leq L_\psi(t)$, then there must be a smaller positive integer t_0 such that $L_\sigma(t_0) > L_\psi(t_0)$.*

Proof *Assuming that such t_0 does not exist, then for any $t \in \mathbb{N}$, we have $L_\sigma(t) \leq L_\psi(t)$. Due to $\psi \neq \sigma$ and both codes being complete prefix, we further obtain that there exists $t_1 \in \mathbb{N}$ such that $L_\sigma(t_1) < L_\psi(t_1)$. Then,*

$$\begin{aligned}\sum_{t\in\mathbb{N}} \frac{1}{2^{L_\sigma(t)}} &= \frac{1}{2^{L_\sigma(t_1)}} + \sum_{t\in\mathbb{N}\setminus\{t_1\}} \frac{1}{2^{L_\sigma(t)}} \\ &> \frac{1}{2^{L_\psi(t_1)}} + \sum_{t\in\mathbb{N}\setminus\{t_1\}} \frac{1}{2^{L_\psi(t)}} = 1.\end{aligned}$$

This contradicts that σ is a complete prefix code.

Lemma 3 shows that the codeword length advantage of the σ at larger integers is at the expense of codeword length at smaller integers. So this simple approach does not work.

Another approach is to consider whether there is a $(2,\infty)$-threshold secret sharing scheme such that the share size $L(t)$ generated at sufficiently large moments t are all the smallest. This essentially considers the asymptotic performance of $(2,\infty)$-threshold secret sharing schemes. Although the previous works [9], [11] did not precisely formulate metrics, they all believed that the standard for a good scheme is that the integer code is at a sufficiently large time t, and the codeword length $L(t)$ is as small as possible.

It is known that the codeword length advantage at larger integers is at the expense of codeword length at smaller integers. However, in evolving secret sharing, the share size $L(t)$ of the larger moment t is obviously not as important as the share size of the smaller moment due to the unknown number of parties. Therefore, this idea is unreasonable when considering global performance, and we need to find more reasonable and feasible metrics.

B. The global metric K_Σ

In fact, evolving secret sharing has one thing in common with universal coding of integers: they both face unknowns. Evolving secret sharing has no prior knowledge of the number of parties, and universal coding of integers has no prior knowledge of probability distributions. Therefore, we suggest proposing a metric similar to the metric minimum expansion factor K_σ^* to evaluate the overall performance of $(2,\infty)$-threshold secret sharing schemes. The new metric is defined as follows.

Definition 8 *Let Σ be a $(2,\infty)$-threshold secret sharing scheme, which corresponds to the integer code σ, and the size of the share distributed at t-th moment is $L_\sigma(t)$ bits. The global metric K_Σ of the scheme Σ is defined as follows:*

$$\begin{aligned}K_\Sigma &:= \sup_{2\leq n\in\mathbb{N}} \frac{\sum_{t=1}^{n} L_\sigma(t)}{\min[sum(2,n)]} \\ &= \sup_{2\leq n\in\mathbb{N}} \frac{\sum_{t=1}^{n} L_\sigma(t)}{nm+2l},\end{aligned} \tag{3}$$

where $m := \lfloor \log_2 n \rfloor$ and $l := n - 2^m$.

The meaning of the global metric K_Σ is that no matter how many parties participate in secret sharing when any fixed number of parties is n_0, the sum of the share sizes for the $(2,\infty)$-threshold secret sharing scheme Σ is less than or equal to K_Σ times of the sum of the share sizes for the optimal $(2,n_0)$-threshold secret sharing scheme.

Finally, the optimal $(2,\infty)$-threshold secret sharing scheme and the optimal integer code for $(2,\infty)$-threshold secret sharing schemes are defined below.

Definition 9 *The $(2,\infty)$-threshold secret sharing scheme with the smallest global metric K_Σ is called the optimal $(2,\infty)$-threshold secret sharing scheme. The integer code corresponding to the optimal $(2,\infty)$-threshold secret sharing scheme is called the optimal integer code for $(2,\infty)$-threshold secret sharing schemes.*

V. The Range of K_Σ for the Optimal $(2,\infty)$-Threshold Secret Sharing Scheme

In this section, we study the range of the global metric K_Σ for the optimal $(2,\infty)$-threshold secret sharing scheme. First, the lower bound of the global metric K_Σ is given. For simplicity, let

$$\mathcal{L}(n,\sigma) := \frac{\sum_{t=1}^{n} L_\sigma(t)}{nm+2l}. \tag{4}$$

Consider the case where the number of parties $n = 2$. Due to $L_\sigma(1) \geq 1$ and $L_\sigma(2) \geq 2$ for any integer code σ, we obtain

$$\mathcal{L}(2,\sigma) = \frac{\sum_{t=1}^{2} L_\sigma(t)}{2\times 1 + 2\times 0} \geq \frac{3}{2}.$$

Therefore, the global metric K_Σ satisfies $K_\Sigma \geq 1.5$.

Next, we will construct schemes Σ in the following two subsections so that its the global metric K_Σ is close to the lower bound 1.5.

A. Panning Code

In this subsection, we construct $(2,\infty)$-threshold secret sharing schemes with small K_Σ using known universal coding of integers. The length of the second codeword of the exisiting constructed universal coding of integers σ is strictly greater than 2 [16], [19], [20], that is, $L_\sigma(2) \geq 3$ and

$$\mathcal{L}(2,\sigma) = \frac{\sum_{t=1}^{2} L_\sigma(t)}{2} \geq 2.$$

Hence, the global metric K_Σ is far from the lower bound 1.5. Therefore, we proposed a novel panning code σ^+ which

is constructed from an existing integer code σ. The panning code $\sigma^+ : \mathbb{N} \to \mathbb{B}^*$ is constructed as follows.

$$\sigma^+(m) := \begin{cases} 1, & \text{if } m = 1, \\ 0\sigma(m-1), & \text{otherwise,} \end{cases}$$

for all $m \in \mathbb{N}$. If the length of the first codeword of integer code σ is 1, then the length of the first codeword of σ^+ is 1 and the length of the second codeword of σ^+ is 2.

Before considering which integer code to be used to construct the $(2, \infty)$-threshold secret sharing scheme, we first prove the following lemma.

Lemma 4 *Let the integer code σ satisfy $L_\sigma(1) = 1$, $L_\sigma(2) = 2$ and*

$$L_\sigma(t) \le a + b\lfloor \log_2(t-1) \rfloor, \tag{5}$$

for all $3 \le t \in \mathbb{N}$, where a and b are positive constants. Then

$$\lim_{n \to +\infty} \mathcal{L}(n, \sigma) \le b. \tag{6}$$

Furthermore, when (5) *meets the equal sign,* (6) *also meets the equal sign.*

Proof *Let $m = \lfloor \log_2 n \rfloor$ and $l = n - 2^m$. When the integer $n \ge 3$, we have*

$$\begin{aligned} \sum_{t=1}^{n} L_\sigma(t) &\le 3 + \sum_{t=3}^{n} \Big(a + b\lfloor \log_2(t-1) \rfloor \Big) \\ &= 3 + a(n-2) + b\sum_{t=2}^{n-1} \lfloor \log_2 t \rfloor \\ &= 3 + a(n-2) + b\Big(lm + \sum_{d=1}^{m-1} d \cdot 2^d \Big) \\ &= 3 + a(n-2) + b\Big[lm + (m-2)2^m + 2 \Big] \\ &= 3 + a(n-2) + b(nm + 2l - 2n + 2). \end{aligned} \tag{7}$$

Then, we obtain

$$\begin{aligned} \lim_{n \to +\infty} \mathcal{L}(n, \sigma) &\le \lim_{n \to +\infty} \frac{3 + a(n-2) + b(nm + 2l - 2n + 2)}{nm + 2l} \\ &= b + \lim_{n \to +\infty} \frac{3 + an - 2a + 2b - 2bn}{nm + 2l} \\ &= b + \lim_{n \to +\infty} \frac{(a - 2b)n}{nm + 2l} \\ &= b. \end{aligned}$$

From the above calculation process, it is easy to see that when (5) *meets the equal sign,* (6) *also meets the equal sign.*

Lemma 4 shows that the integer code σ should be chosen so that the constant b for the panning code σ^+ in (5) is as small as possible. Because the global metric $K_\Sigma \ge 1.5$, we want that the constant b in (5) takes 1.5. In this case, when $n = 2$ and n tends to infinity, $\mathcal{L}(n, \sigma^+)$ is less than or equal to 1.5, and it is reasonable to hope that the global metric at other time instant is better.

Both ι code and η code satisfy that the length of the first codeword is 1, and the constant b in (5) can be set to be 1.5. Therefore, at the end of this subsection, we analyze the global metrics for $(2, \infty)$-threshold secret sharing schemes corresponding to ι^+ code and η^+ code respectively.

First, we analyze the global metrics K_{I^+} corresponding to ι^+ code. The ι^+ code satisfy $L_{\iota^+}(1) = 1$, $L_{\iota^+}(2) = 2$ and

$$L_{\iota^+}(t) = L_\iota(t-1) + 1 \le \frac{7}{2} + \frac{3}{2}\lfloor \log_2(t-1) \rfloor.$$

for all $3 \le t \in \mathbb{N}$. When $n = 4$, we obtain

$$\mathcal{L}(4, \iota^+) = \frac{\sum_{t=1}^{4} L_{\iota^+}(t)}{4 \times 2 + 2 \times 0} = 1.625.$$

When $2 \le n < 15$, we can directly verify that $\mathcal{L}(n, \iota^+) \le 1.625$. When $n \ge 16$ (i.e. $m \ge 4$), $a = \frac{7}{2}$ and $b = \frac{3}{2}$ can be substituted into (7) to get

$$\sum_{t=1}^{n} L_\sigma(t) \le \frac{3}{2}(nm + 2l) + \frac{1}{2}n - 1.$$

Thus, we have

$$\begin{aligned} \mathcal{L}(n, \iota+) &\le \frac{1.5(nm + 2l) + 0.5n - 1}{nm + 2l} \\ &= 1.5 + \frac{0.5n - 1}{nm + 2l} \\ &< 1.5 + \frac{0.5n}{nm} \\ &\le 1.625. \end{aligned}$$

Therefore, the global metric corresponding to the ι^+ code is $K_{I^+} = 1.625$.

Second, we analyze the global metrics K_{H^+} corresponding to η^+ code. The η^+ code satisfy $L_{\eta^+}(1) = 1$, $L_{\eta^+}(2) = 2$ and

$$L_{\eta^+}(t) = L_\eta(t-1) + 1 \le 4 + \frac{3}{2}\lfloor \log_2(t-2) \rfloor.$$

for all $3 \le t \in \mathbb{N}$. When $n = 32$, we obtain

$$\mathcal{L}(32, \eta^+) = \frac{\sum_{t=1}^{32} L_{\eta^+}(t)}{32 \times 5 + 2 \times 0} = 1.61875.$$

When $2 \le n < 512$, we can directly verify that $\mathcal{L}(n, \eta^+) \le 1.61875$. When $n \ge 512$ (i.e. $m \ge 9$), we consider the following two cases.

1) n is a power of 2:
   In this case, $n = 2^m$ and $nm + 2l = nm$. When $m \ge 9$, we obtain

$$\begin{aligned} \sum_{t=1}^{n} L_{\eta^+}(t) &\le 3 + \sum_{t=3}^{n} \Big(4 + \frac{3}{2}\lfloor \log_2(t-2) \rfloor \Big) \\ &= 4n - 5 + \frac{3}{2}\sum_{t=2}^{n-2} \lfloor \log_2 t \rfloor \\ &= 4n - 5 + \frac{3}{2}\Big[\sum_{d=1}^{m-1} d \cdot 2^d - (m-1) \Big] \\ &= 4n - 5 + \frac{3}{2}\Big[(m-2)2^m - m + 3 \Big] \\ &= \frac{3}{2}nm + n - \frac{3}{2}m - \frac{1}{2}. \end{aligned}$$

   Thus, we have

$$\begin{aligned} \mathcal{L}(n, \eta^+) &\le 1.5 + \frac{n - 1.5m - 0.5}{nm} \\ &< 1.5 + \frac{n}{nm} \\ &< 1.61875. \end{aligned}$$

2) n is not a power of 2:
In this case, $n = 2^m + l$ and $1 \le l \le 2^m - 1$. When $m \ge 9$, we obtain

$$\begin{aligned}\sum_{t=1}^{n} L_{\eta^+}(t) &\le 4n - 5 + \frac{3}{2}\sum_{t=2}^{n-2}\lfloor \log_2 t \rfloor \\ &= 4n - 5 + \frac{3}{2}\Big[(l-1)m + \sum_{d=1}^{m-1} d \cdot 2^d\Big] \\ &= 4n - 5 + \frac{3}{2}\Big[lm - m + (m-2)2^m + 2\Big] \\ &= \frac{3}{2}\Big(nm + 2l\Big) + n - 1.5m - 2.\end{aligned}$$

Thus, we have

$$\begin{aligned}\mathcal{L}(n, \eta^+) &\le 1.5 + \frac{n - 1.5m - 2}{nm + 2l} \\ &< 1.5 + \frac{n}{nm} \\ &< 1.61875.\end{aligned}$$

In summary, when $m \ge 9$, we obtain $\mathcal{L}(n, \eta^+) < 1.61875$. Therefore, the global metric corresponding to the η^+ code is $K_{H^+} = 1.61875$.

It can be seen that the global metrics $K_{I^+} = 1.625$ and $K_{H^+} = 1.61875$ are close to the lower bound 1.5. Hence, $(2, \infty)$-threshold secret sharing schemes I^+ and H^+ corresponding to ι^+ code and η^+ code have good performance under the new metric defined in Definition 8.

B. λ code achieves $K_\Lambda = 1.59375$

In this subsection, we construct a new integer code, termed λ code, to achieve the global metric $K_\Lambda = 1.59375$. In the previous subsection, the panning codes constructed using the existing integer codes achieve good global metrics. At present, the best performance is the global metric $K_{H^+} = 1.61875$, which is achieved by η^+ code. The structure of λ code is related to η^+ code, and the specific structure is as follows.

Let $\boldsymbol{a} \in \mathbb{B}^m$ denote a codeword of length m, then $\boldsymbol{a}0$ and $\boldsymbol{a}1$ are two codewords of length $m+1$. We call the process from $\boldsymbol{a}$ to $\boldsymbol{a}0$ and $\boldsymbol{a}1$ as *the splitting of* $\boldsymbol{a}$. If $\boldsymbol{a}$ is a codeword of σ code, and $\boldsymbol{a}$ is replaced by two codewords after $\boldsymbol{a}$ splits, then σ code is said to be split at $\boldsymbol{a}$, and σ code after the split is noted as $\sigma[\boldsymbol{a}]$. Obviously, if σ code is a prefix code, then $\sigma[\boldsymbol{a}]$ code is still a prefix code.

λ code is essentially the code obtained after η^+ code is splits at $\eta^+(5)$, $\eta^+(8)$, $\eta^+(9)$, $\eta^+(14)$, $\eta^+(15)$, $\eta^+(16)$ and $\eta^+(17)$, that is,

$$\lambda = \eta^+[\eta^+(5), \eta^+(8), \eta^+(9), \eta^+(14), \eta^+(15), \eta^+(16), \eta^+(17)].$$

Table II lists the first 24 codewords of η^+ code and λ code. The underlined part in Table II is related to the split codeword. $\sum_{t=1}^{24} L_\lambda(t) = 174$ can be obtained by simple calculations. When $n \ge 25$, $\lambda(n) = \eta^+(n-7) = 0\eta(n-8)$. Therefore, we obtain

$$L_\lambda(n) = L_\eta(n-8) + 1 \le 4 + \frac{3}{2}\lfloor \log_2(n-9) \rfloor,$$

for all $25 \le n \in \mathbb{N}$.

TABLE II: The first 24 codewords of η^+ code and λ code

n	η^+ code	λ code
1	1	1
2	01	01
3	0010	0010
4	00110	00110
5	00111	001110
6	0001000	001111
7	0001001	0001000
8	0001010	0001001
9	0001011	00010100
10	00011000	00010101
11	00011001	00010110
12	00011010	00010111
13	00011011	00011000
14	00011100	00011001
15	00011101	00011010
16	00011110	00011011
17	00011111	000111000
18	0000100000	000111001
19	0000100001	000111010
20	0000100010	000111011
21	0000100011	000111100
22	0000100100	000111101
23	0000100101	000111110
24	0000100110	000111111

Next, we analyze the global metric K_Λ corresponding to λ code. When $n = 16$, we obtain

$$\mathcal{L}(16, \lambda) = \frac{\sum_{t=1}^{16} L_\lambda(t)}{16 \times 4 + 2 \times 0} = 1.59375.$$

When $2 \le n < 2048$, we can directly verify that $\mathcal{L}(n, \lambda) \le 1.59375$. When $n \ge 2048$ (i.e. $m \ge 11$), we have

$$\begin{aligned}\sum_{t=1}^{n} L_\lambda(t) &= 174 + \sum_{t=25}^{n} L_\lambda(t) \\ &\le 174 + 4(n-24) + \frac{3}{2}\sum_{t=25}^{n}\lfloor \log_2(t-9) \rfloor \\ &= 78 + 4n + \frac{3}{2}\sum_{t=16}^{n-9}\lfloor \log_2 t \rfloor.\end{aligned}$$

According to the value of $l = n - 2^m$, the following two cases are discussed.

1) $8 \le l \le 2^m - 1$:
When $m \ge 11$, we obtain

$$\begin{aligned}\sum_{t=16}^{n-9}\lfloor \log_2 t \rfloor &= \sum_{d=4}^{m-1} d \cdot 2^d + m(l-8) \\ &= (m-2)2^m - 32 + ml - 8m.\end{aligned}$$

Thus, we have

$$\begin{aligned}\mathcal{L}(n, \lambda) &\le \frac{78 + 4n + 1.5\Big[(m-2)2^m - 32 + ml - 8m\Big]}{nm + 2l} \\ &= 1.5 + \frac{n - 12m + 30}{nm + 2l} \\ &< 1.5 + \frac{n}{nm + 2l} \\ &< 1.5 + \frac{1}{m} \\ &< 1.59375.\end{aligned}$$

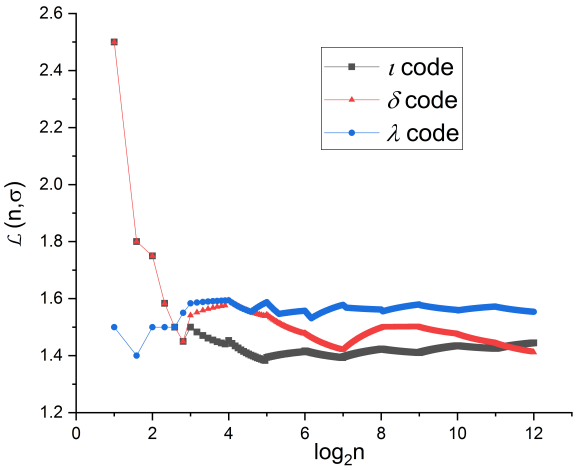

Fig. 1: $\mathcal{L}(n,\sigma)$ of δ code, ι code and λ code when $2 \le n \le 2^{12}$.

2) $0 \le l < 8$:
When $m \ge 11$, we obtain

$$\sum_{t=16}^{n-9} \lfloor \log_2 t \rfloor = \sum_{d=4}^{m-1} d \cdot 2^d + (m-1)(l-8) = (m-2)2^m - 24 + ml - 8m - l.$$

Thus, we have

$$\begin{aligned}\mathcal{L}(n,\lambda) &\le \frac{78+4n+1.5\left[(m-2)2^m-24+ml-8m-l\right]}{nm+2l} \\ &= 1.5 + \frac{n-12m-1.5l+42}{nm+2l} \\ &< 1.5 + \frac{n}{nm+2l} \\ &< 1.59375.\end{aligned}$$

In summary, when $m \ge 11$, we obtain $\mathcal{L}(n,\lambda) < 1.59375$. Therefore, the global metric corresponding to the λ code is $K_\Lambda = 1.59375$. Furthermore, we prove that the range of the global metric K_Σ for the optimal $(2,\infty)$-threshold secret sharing scheme is $1.5 \le K_\Sigma \le 1.59375$.

VI. Comparisons

In this section, we compare the global performance of $(2,\infty)$-threshold secret sharing schemes constructed by Elias δ code [16], ι code [20], and the proposed λ code. Komargodski *et al.* use δ code to construct a $(2,\infty)$-threshold secret sharing scheme that is almost optimal in terms of asymptotic performance [9]. ι code is currently the best universal coding of integers with the smallest expansion factor [20]. A $(2,\infty)$-threshold secret sharing scheme constructed by λ code is currently the best integer code in terms of global performance.

$\mathcal{L}(n,\sigma)$ is defined in (4), which represents the ratio of the sum of the share sizes for scheme Σ at the first n moments to the capacity of $(2,n)$-threshold secret sharing schemes. Figure 1 is drawn according to the code lengths of these three integer codes, the abscissa is the logarithmic value of the number of parties n, and the ordinate is $\mathcal{L}(n,\sigma)$. From Figure 1, we obtain $K_\Lambda = 1.59375 < K_\Delta = K_I = 2.5$ (strict proof similar to calculation in Section V). Therefore, λ code is better than δ code and ι code in terms of global performance. The fold line of λ code is flat overall. The advantage of δ code and ι code over λ code at relatively large n is obtained at the expense of the first four points in Figure 1. Similarly, the advantage of δ code over ι code at sufficiently large n is to sacrifice about the first 2^{11} points in Figure 1.

Although the metric K_Σ reflects the global performance of $(2,\infty)$-threshold secret sharing scheme, it can be seen from Figure 1 that this metric has limitation. When the number of parties n is clearly greater than 6, the scheme of λ code will not be better than the scheme of δ code and ι code. Therefore, if the number of parties n is known to be greater than or equal to n_l, we can define a new metric $K_\Sigma(n_l)$ dependent on n_l as follows.

$$\begin{aligned}K_\Sigma(n_l) &:= \sup_{n_l \le n \in \mathbb{N}} \frac{\sum_{s=1}^{n} L_\sigma(s)}{\min[sum(2,n)]} \\ &= \sup_{n_l \le n \in \mathbb{N}} \mathcal{L}(n,\sigma).\end{aligned}$$

In particular, when $n_l = 2$, the metric $K_\Sigma(n_l)$ becomes K_Σ. In addition, $\lim_{n_l \to +\infty} K_\Sigma(n_l) = \lim_{n \to +\infty} \mathcal{L}(n,\sigma)$ can be considered as a metric for asymptotic performance. The study of the properties of $K_\Sigma(n_l)$ is a promising future work.

VII. Conclusions

In this paper, we propose a new metric K_Σ for evolving 2-threshold secret sharing schemes Σ and study the range of the global metric K_Σ for the optimal $(2,\infty)$-threshold secret sharing scheme. First, we show that the global metric $K_\Sigma \ge 1.5$ and use the existing universal coding of integers to construct schemes with good global metrics. Second, we construct a new integer code, termed λ code, to achieve the global metric $K_\Lambda = 1.59375$. This work shows that the range of the global metric K_Σ for the optimal $(2,\infty)$-threshold secret sharing scheme is $1.5 \le K_\Sigma \le 1.59375$. Furthermore, the capacity of $(2,n)$-threshold secret sharing scheme is proved. The future work is listed as follows.

1) Is it possible to construct an integer code whose global metric is strictly less than 1.59375?
2) The explicit value of K_Σ of the optimal $(2,\infty)$-threshold secret sharing scheme is still unknown.
3) Let n_l be a known lower bound on the number of parties n. Research on general metric $K_\Sigma(n_l)$, where $n_l \ge 2$ is an arbitrary integer.

References

[1] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.
[2] G. R. Blakley, "Safeguarding cryptographic keys," in *1979 International Workshop on Managing Requirements Knowledge (MARK)*, 1979, pp. 313–318.
[3] M. K. Franklin and M. K. Reiter, "Verifiable signature sharing," in *Advances in Cryptology — EUROCRYPT '95*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 50–63.

[4] Y. Desmedt and Y. Frankel, "Threshold cryptosystems," in *Advances in Cryptology — CRYPTO' 89 Proceedings*. New York, NY: Springer New York, 1990, pp. 307–315.

[5] Y. Desmedt and Y. Frankel, "Shared generation of authenticators and signatures," in *Advances in Cryptology — CRYPTO '91*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992, pp. 457–469.

[6] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," in *Advances in Cryptology — CRYPTO' 99*. Berlin, Heidelbergz: Springer Berlin Heidelberg, 1999, pp. 148–164.

[7] M. Ito, A. Saito, and T. Nishizeki, "Secret sharing scheme realizing general access structure," *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, vol. 72, no. 9, pp. 56–64, Sept. 1989.

[8] I. Komargodski, M. Naor, and E. Yogev, "How to share a secret, infinitely," in *Theory of Cryptography*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 485–514.

[9] I. Komargodski, M. Naor, and E. Yogev, "How to share a secret, infinitely," *IEEE Transactions on Information Theory*, vol. 64, no. 6, pp. 4179–4190, Jun. 2018.

[10] P. D'Arco, R. De Prisco, and A. De Santis, "On the equivalence of 2-threshold secret sharing schemes and prefix codes," in *10th International Symposium, CSS 2018*, 2018, pp. 157–167.

[11] R. Okamura and H. Koga, "New constructions of an evolving 2-threshold scheme based on binary or d-ary prefix codes," in *2020 International Symposium on Information Theory and Its Applications (ISITA)*, 2020, pp. 432–436.

[12] I. Komargodski and A. Paskin-Cherniavsky, "Evolving secret sharing: Dynamic thresholds and robustness," in *Theory of Cryptography*. Cham: Springer International Publishing, 2017, pp. 379–393.

[13] P. D'Arco, R. D. Prisco, A. D. Santis, A. P. del Pozo, and U. Vaccaro, "Probabilistic secret sharing," in *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), vol. 117. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018, pp. 64:1–64:16.

[14] A. Beimel and H. Othman, "Evolving ramp secret-sharing schemes," in *Security and Cryptography for Networks*. Cham: Springer International Publishing, 2018, pp. 313–332.

[15] A. Beimel and H. Othman, "Evolving ramp secret sharing with a small gap," in *Advances in Cryptology – EUROCRYPT 2020*. Cham: Springer International Publishing, 2020, pp. 529–555.

[16] P. Elias, "Universal codeword sets and representations of the integers," *IEEE Transactions on Information Theory*, vol. 21, no. 2, pp. 194–203, Mar. 1975.

[17] K. B. Lakshmanan, "On universal codeword sets," *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 659–662, Sep. 1981.

[18] H. Yamamoto, "A new recursive universal code of the positive integers," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 717–723, Mar. 2000.

[19] W. Yan and S.-J. Lin, "On the minimum of the expansion factor for universal coding of integers," *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7309–7319, Nov. 2021.

[20] W. Yan and S.-J. Lin, "A tighter upper bound of the expansion factor for universal coding of integers and its code constructions," *IEEE Transactions on Communications*, vol. 70, no. 7, pp. 4429–4438, Jul. 2022.

[21] J. Kilian and N. Nisan, Unpublished result, 1990.

[22] I. Cascudo, R. Cramer, and C. Xing, "Bounds on the threshold gap in secret sharing and its applications," *IEEE Transactions on Information Theory*, vol. 59, no. 9, pp. 5600–5612, Sep. 2013.

[23] L. G. Kraft, "A device for quantizing, grouping, and coding amplitude-modulated pulses," Master's thesis, Thesis (M.S.) Massachusetts Institute of Technology. Dept. of Electrical Engineering, Cambridge, Mass., 1949.

[24] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.

Wei Yan received the B.Sc. degree in mathematics and applied mathematics and the Ph.D. degree in Cyberspace Security from the University of Science and Technology of China (USTC), Hefei, China, in 2017 and 2022, respectively. He is currently a Researcher with the Theory Laboratory, 2012 Labs, Huawei Technologies Company Ltd. His research interest includes coding theory and data compression.

Sian-Jheng Lin received the B.Sc., M.Sc., and Ph.D. degrees in computer science from National Chiao Tung University, Hsinchu, Taiwan, in 2004, 2006, and 2010, respectively. From 2010 to 2014, he was a postdoc with the Research Center for Information Technology Innovation, Academia Sinica. From 2014 to 2016, he was a postdoc with the Electrical Engineering Department at King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. From 2016 to 2021, he was a researcher with the School of Information Science and Technology at University of Science and Technology of China (USTC), Hefei, China. He is currently a Senior Researcher with the Theory Laboratory, 2012 Labs, Huawei Technologies Company Ltd. In recent years, his research focuses on the codes for storage systems and data compressions.

Yunghsiang S. Han was born in Taipei, Taiwan, in 1962. He received B.Sc. and M.Sc. degrees in electrical engineering from the National Tsing Hua University, Hsinchu, Taiwan, in 1984 and 1986, respectively, and a Ph.D. from the School of Computer and Information Science, Syracuse University, Syracuse, NY, in 1993. From 1986 to 1988, he was a lecturer at Ming-Hsin Engineering College, Hsinchu, Taiwan. He was a teaching assistant from 1989 to 1992 and a research associate in the School of Computer and Information Science at Syracuse University from 1992 to 1993. From 1993 to 1997, he was an Associate Professor in the Department of Electronic Engineering at Hua Fan College of Humanities and Technology, Taipei Hsien, Taiwan. He was with the Department of Computer Science and Information Engineering at National Chi Nan University, Nantou, Taiwan from 1997 to 2004. He was promoted to Professor in 1998. He was a visiting scholar in the Department of Electrical Engineering at the University of Hawaii at Manoa, HI from June to October 2001, the SUPRIA visiting research scholar in the Department of Electrical Engineering and Computer Science and CASE center at Syracuse University, NY from September 2002 to January 2004 and July 2012 to June 2013, and the visiting scholar in the Department of Electrical and Computer Engineering at the University of Texas at Austin, TX from August 2008 to June 2009. He was with the Graduate Institute of Communication Engineering at National Taipei University, Taipei, Taiwan from August 2004 to July 2010. From August 2010 to January 2017, he was Chair Professor with the Department of Electrical Engineering at the National Taiwan University of Science and Technology. From February 2017 to February 2021, he was with the School of Electrical Engineering & Intelligentization at Dongguan University of Technology, China. Now he is with the Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China. He is also a Chair Professor at National Taipei University since February 2015. His research interests are in error-control coding, wireless networks, and security.

Dr. Han was a winner of the 1994 Syracuse University Doctoral Prize and a Fellow of IEEE. One of his papers won the prestigious 2013 ACM CCS Test-of-Time Award in cybersecurity.