Abstract—From currency to cloud storage systems, the continuous rise of the blockchain technology is moving various information systems towards decentralization. Blockchain-based decentralized storage networks (DSNs) offer significantly higher privacy and lower costs to customers compared with centralized cloud storage associated with specific vendors. Coding is required in order to retrieve data stored on failing components. While coding solutions for centralized storage have been intensely studied, topology-aware coding for heterogeneous DSNs have not yet been discussed. In this paper, we propose a joint coding scheme where each node receives extra protection through the cooperation with nodes in its neighborhood in a heterogeneous DSN with any given topology. As an extension of, which also subsumes, our prior work on coding for centralized cloud storage, our proposed construction preserves desirable properties such as scalability and flexibility in networks with varying topologies.

Index Terms—Joint hierarchical coding, cooperative data protection, blockchain technology, decentralized storage networks.

I. INTRODUCTION

The blockchain technology, first introduced by Satoshi Nakamoto as a technology supporting the digital currency called bitcoin, has been intensively discussed and regarded as a substantial innovation in cryptosystems [1]–[8]. Blockchain enables recording transactions through a decentralized deployment, which effectively addresses the potential issues of compromised data privacy and key abuse, arising from the existence of a central node that monopolizes all the actions and resource allocations in traditional centralized systems. Decentralization has the potential to universally revolutionize a variety of applications, one of which is cloud storage.

In contrast to traditional centralized storage based on the client-server model, where big companies monopolize renting the storage space to users, blockchain-based decentralized storage networks (DSNs) enable non-enterprise users to not only get access to the network storage space but also to contribute to increasing it via renting their remaining storage space on personal devices. Blockchain-based DSNs allocate the storage space and distribute the encrypted user data with the validation and integrity certification of a third-party, and thus have potential to offer higher privacy, higher reliability, and lower cost than currently available solutions. It has been found that the benefits in data integrity of decentralization are typically at the cost of higher latency and difficult maintenance.

While blockchain technology empowers DSNs to ensure the network-layer security, appropriate error-correction codes (ECC) are needed to further improve the physical-layer reliability. In a coded DSN, data are encoded via an ECC and stored in nodes, where each node is allowed to communicate and exchange information with its neighboring nodes. If data in each node are encoded independently, the system will be vulnerable to information leakage, data loss, and data tampering by malicious users. Therefore, enabling joint encoding of the data stored in all nodes such that nodes in the neighborhood cooperatively protect and validate their stored data collectively in the DSN is an essential requirement.

In existing literature on coding for distributed storage [4]–[6], there has been no explicit consideration of random distributions and clustering nature of network nodes. Clustered distributed storage has received attention in recent years in the context of multi-rack storage, where either the sizes of clusters and the capacities of the communication links are considered to be homogeneous [7]–[10], or the network topology has a simple structure [11], [12]. However, DSNs typically have more sophisticated topologies characterized by heterogeneity among capacities of communication links and erasure statistics of nodes due to the random and dynamic nature of practical networks [12]–[16]. Instead of solutions for simplified models, schemes that fit into any topology (referred to as “topology-awareness” later on) are desired to exploit the existing resources, achieving lower latency and higher reliability.

To further reduce latency and decoding complexity, we propose to provide each node with multiple ECC capabilities enabled by cooperating with neighboring nodes in a series of nested sets with increasing sizes: we refer to this as ECC hierarchy later on in the paper. As the size of the set increases, the associated ECC capability increases. In our method, we consider the case where each node stores encoded messages with local data protection, where the data length and codeword length can be customized by the user at each node.

In this paper, we introduce a topology-aware coding scheme that enables hierarchical cooperative data protection among nodes in a DSN, which is built upon our prior work in [17] in the context of centralized cloud storage and preserves desirable properties like scalability and flexibility. The rest of the paper is organized as follows. In Section II, we introduce the DSN model and necessary preliminaries. In Section III, we define ECC hierarchies as well as their depth, and present a coding scheme with depth 1. In Section IV, we define the notion of compatible cooperation graphs, and propose an explicit construction of hierarchical codes for nodes with their cooperation graph being compatible. Finally, we summarize our results and discuss future directions in Section V.

II. SYSTEM MODEL AND PRELIMINARIES

In this section, we discuss the model and mathematical representation of a DSN, as well as necessary preliminari-
Fig. 1: Decentralized storage network (DSN). For the cluster with
the master node \( v_i \), message \( m_i \) is encoded to \( c_i \) and symbols of
\( c_i \) are stored distributively among non-master nodes that are locally
connected to \( v_i \). In the figures after Fig. 1 we omit the local non-
master nodes for clarity of figures.

ies. Throughout the remainder of this paper, \( [N] \) refers to
\{1, 2, \ldots, N\}. For vectors \( u \) and \( v \) of the same length \( p \), \( u \succ v \)
and \( u \prec v \) implies \( u_i > v_i \) and \( u_i < v_i \), for all \( i \in [p] \),
respectively; \( u \succeq v \) means “\( u \succ v \) or \( u = v \)”, and \( u \preceq v \)
means “\( u \prec v \) or \( u = v \)”.

A. Decentralized Network Storage

As shown in Fig. 1, a DSN is modeled as a graph \( G(V, E) \),
where \( V \) and \( E \) denote the set of nodes (master only)
and edges, respectively. Codewords are stored among the nodes in
a cluster. A failed node in a cluster is regarded as an erased
symbol in the codeword stored in this cluster. A cluster is
represented in \( G \) by its master node \( v_i \) in \( V \) solely. Each
element \( e_{i,j} \in E \) represents a communication link connecting
node \( v_i \) and node \( v_j \), through which \( v_i \) and \( v_j \) are allowed
to exchange information. Denote the set of all neighbors of
node \( v_i \) by \( N_i \), e.g., \( N_i = \{ j_1, j_2, j_3 \} \) in Fig. 1 and refer to
it as the neighborhood of node \( v_i \). Messages \( \{ m_i \}_{i \in V} \)
are jointly encoded as \( \{ c_i \}_{i \in V} \), and \( c_i \) is stored in \( v_i \). For a DSN
denoted by \( G(V, E) \), let \( p = |V| \). Suppose \( G \) is associated with
a tuple \((n, k, r) \in \mathbb{N}^3\), where \( k, r > 0 \) and \( n = k + r \). Note
that \( k_i \) represents the length of the message \( m_i \) associated
with \( v_i \in V \); \( n_i \) and \( r_i \) denote the length of \( c_i \) stored in \( v_i \) and its
syndrome, respectively.

B. Preliminaries

Based on the aforementioned notation, a systematic generator
matrix of a code for \( G(V, E) \) has the following structure:

$$
G = \begin{bmatrix}
I_{k_1} & A_{1,1} & 0 & A_{1,2} & \cdots & 0 & A_{1,p} \\
0 & A_{2,1} & I_{k_2} & A_{2,2} & \cdots & 0 & A_{2,p} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & A_{p,1} & 0 & A_{p,2} & \cdots & I_{k_p} & A_{p,p}
\end{bmatrix}, \quad (1)
$$

where all elements are from a Galois field \( \text{GF}(q) \), \( q = 2^\theta \) and
\( \theta \geq 2 \). The major components of our construction are the so-
called Cauchy matrices specified in Definition 1.

Definition 1. (Cauchy matrix) Let \( s, t \in \mathbb{N} \) and \( \text{GF}(q) \) be a
finite field of size \( q \). Suppose \( a_1, \ldots, a_s, b_1, \ldots, b_t \) are \( s \times t \)
distinct elements in \( \text{GF}(q) \). The following matrix is known as a
Cauchy matrix.

$$
C(a_1, b_1) = \begin{bmatrix}
a_1 & b_1 \\
a_2 & b_2 \\
\vdots & \vdots \\
a_s & b_t
\end{bmatrix}.
$$

We denote this matrix by \( Y(a_1, \ldots, a_s; b_1, \ldots, b_t) \).

III. COOPERATIVE DATA PROTECTION

In this section, we first mathematically describe the ECC
hierarchy of a DSN and its depth, which specifies the ECC
capabilities of nodes while cooperating with different sets of
other nodes. We then propose a cooperation scheme where each
node only cooperates with its single-hop neighbors.

A. ECC Hierarchy

Denote the ECC hierarchy of node \( v_i \in V \) by a sequence
\( d_i = (d_{i,0}, d_{i,1}, \ldots, d_{i,L_i}) \), where \( L_i \) is called the depth of \( d_i \),
and \( d_{i,l} \) represents the maximum number of erased symbols \( v_i \)
can recover in its codeword \( c_i \) from the \( l \)-th level cooperation,
for all \( l \in [L_i] \). The 0-th level cooperation refers to local erasure
correction, i.e., the local node \( v_i \) recovers its data without
communicating with neighboring nodes.

For each \( v_i \in V \) such that \( L_i > 0 \), there exist two series of sets
of nodes, denoted by \( \varnothing \subset A_i^1 \subset A_i^2 \subset \cdots \subset A_i^{L_i} \subset V \)
and \( \{ B_i^l \}_{l=1}^{L_i} \), where \( A_i^l \cap B_i^l = \varnothing \) for all \( l \in [L_i] \), and a
sequence \( (\lambda_i,l,W)_{\varnothing \subseteq W \subseteq B_i^l} \). In the \( l \)-th level cooperation, node
\( v_i \in V \) tolerates \( \lambda_i,l,W \) erasures if all nodes in \( \bigcup_{l=1}^{L_i} W \) are able to decode their own messages, where the
maximum value is \( \lambda_i,l,W = d_{i,l} \) and is reached when \( W = \varnothing \).
See Fig. 2 We first take a look at the cooperation schemes with
ECC hierarchy of depth 1.

B. Single-Level Cooperation

We next discuss the case where each node only has coop-
eration of depth 1. Consider a DSN represented by \( G(V, E) \)
that is associated with parameters \((n, k, r)\) and a class of
sets \( \{ M_i \}_{i \in V} \) such that \( \varnothing \subset M_i \subseteq N_i \), for all \( v_i \in V \).
In Construction 1 we present a joint coding scheme where
node \( v_i \) only cooperates with nodes in \( M_i \), for all \( v_i \in V \).
Heterogeneity is obviously achieved since \( n_i \), \( k_i \), \( r_i \), are not
required to be identical for all \( v_i \in V \).

Construction 1. Let \( G(V, E) \) represent a DSN associated with
parameters \((n, k, r)\) and a local ECC parameter \( \delta \), where \( r > \delta \geq 0 \). Let \( p = |V| \) and \( \text{GF}(q) \) be a Galois field of size \( q \),
where \( q > \max_{v_i \in V} \left( n_i + \delta_i + \sum_{j \in M_i} \delta_j \right) \).
Theorem 1. Consider the DSN shown in Fig. 3; let $M_i = N_i$ in Construction 1 for all $i \in [12]$. The matrix in (3) is obtained by removing all the block components of identity surrounded by zero matrices from the generator matrix of $C_i$, and is referred to as the non-systematic component of the generator matrix.

Take node $v_2$ as an example. Observe that $A_1^2 = M_2 = \{v_1, v_3, v_5\}$, $A_1^3 = M_3 = \{v_2, v_4\}$, and $A_1^4 = M_4 = \{v_2, v_4, v_6, v_8\}$. Therefore, $B_1^4 = \bigcup_{i \in \{1,3,5\}} M_i \setminus \{v_1, v_3, v_5\}$ is a valid factor of $B_1^4$. Moreover, $d_2 = (v_2 - d_2, r_2 + \sum_{j \in \{1,3,5\}} \delta_j)$, $\lambda_{2,1;v_2} = \lambda_{3,1;v_2} = \lambda_{5,1;v_2} = r_2 + \delta_1$, and $\lambda_{2,1;v_4} = \lambda_{3,1;v_4} = \lambda_{5,1;v_4} = r_2 + \delta_2 + \delta_3$. Consider the case where the 1-st level cooperation of $v_2$ is initiated, i.e., the number of erasures lies within the interval $[r_2 - \delta_2 + 1, r_2 + \delta_1 + \delta_3]$. Then, if $m_1, m_3, m_5$ are all locally recoverable, the cross parities $m_1 B_1^2, m_3 B_1^2, m_5 B_2^2$ computed from the non-diagonal parts in the generator matrix can be subtracted from the parity part of $e_2$ to get $m_2 A_2^2$. Moreover, the successful decoding of $m_2$ makes $m_2 B_2^1$ known to $v_2$. This process provides $(r_2 + \delta_1)$ parities for $m_2$, and thus allows $v_2$ to tolerate $(r_2 + \delta_1)$ erasures.

In order to correct more than $(r_2 + \delta_1)$ erasures, we need extra cross parities generated from $B_2^3 U_3$ and $B_2^5 U_5$. However, local decoding only allows $v_3, v_5$ to know $m_3 B_2^3 + m_5 B_3^5 + m_5 B_5^5$, respectively. Therefore, $v_3$ needs $m_4$ to be recoverable to obtain the extra $\delta_3$ cross parities, and $v_5$ needs $m_6, m_8$ to be recoverable to obtain the extra $\delta_5$ cross parities.

As shown in Example 1 instead of presenting a rigid ECC capability, our proposed scheme enables nodes to have correction of a growing number of erasures with bigger sets of neighboring nodes recovering their messages. Therefore, nodes automatically choose the shortest path to recover their messages, significantly increasing the average recovery speed, especially when the erasures are distributed non-uniformly and sparsely, which is important for blockchain-based DSNs [18]. Moreover, nodes with higher reliabilities are utilized to help decode the data of less reliable nodes, enabling correction of error patterns that are not recoverable in our previous work in [17]. We show these properties in Example 2 and Example 3.

Example 2. (Faster Recovery Speed) Consider the DSN with the cooperation scheme specified in Example 1. Suppose the time to be consumed on transferring information through the communication link $e_{ij}$ is $t_{ij} \in \mathbb{R}^+$, where $t_{ij} = t_{ji}$, for all $i, j \in [12], i \neq j$, and $\max\{t_{1,2}, t_{2,5}\} < (t_{2,3} + t_{3,4}) < t_{2,5} + \min\{t_{4,5}, t_{5,6}, t_{5,8}\}$.

Consider the case where node $c_2$ has $(r_2+1)$ erasures, which
implies that apart from the case of \( m_1, m_3, m_5 \) being obtained locally, recovering \( m_4 \) is sufficient for \( v_3 \) to successfully obtain its message. The time consumed for decoding is \( (t_{2,3} + t_{3,4}) \). Therefore, any system using network coding with the property that a node failure is recovered through accessing more than 4 other nodes will need longer processing time for this case.

**Example 3. (Flexible Erasure Patterns)** Consider the DSN with the cooperation scheme specified in Example 2. Suppose \( \{ m_1 \}_{i \in \{2,4,8,10\}} \) are all locally recoverable. Then, consider the case where \( m_1 \) has \( r_i + 1 \) erasures for \( i \in \{2,4,8,10\} \), which exemplifies a correctable erasure pattern for our proposed codes.

The hierarchical coding scheme presented in [17] can recover from this erasure pattern only if the code used adopts a partition of all nodes into 4 disjoint groups, each of which contains exactly a node from \( \{ v_2, v_4, v_8, v_{10} \} \), as shown in Fig. 3. Moreover, the partition of the code in [17] results in a reduction of the ECC capability of the 1-st level cooperation at every node except for \( v_1, v_2 \) because the additional information originally flow through edges marked in blue no longer exist.

**IV. MULTI-LEVEL COOPERATION**

In this section, we extend the construction presented in Section 5-B to codes with ECC hierarchies of depth larger than 1. We first define the so-called **cooperation graphs** that describe how the nodes are coupled to cooperatively transmit information, and then prove the existence of hierarchical codes over a special class of cooperation graphs: the so-called **compatible graphs**.

**A. Cooperation Graphs**

Based on the aforementioned notation, for each \( v_i \in V \) and \( l \in [L_i] \), let \( I_i^l = A_i^l \setminus A_i^{l-1} \) and refer to it as the \( l \)-th helper of \( v_i \). We next define the so-called **cooperation matrix**.

**Definition 2.** For a joint coding scheme \( C \) for a DSN represented by \( G(V, E) \) with \( |V| = p \), the matrix \( D \in \mathbb{N}^{p \times p} \), in which \( D_{i,j} \) equals to \( l \) for all \( i,j \in [p] \) such that \( j \in I_i^l \), and zero otherwise, is called the **cooperation matrix**.

As an example, the cooperation matrix in Example 1 is exactly the adjacency matrix of the graph in Fig. 3. Note that not every matrix is a cooperation matrix of a set of joint coding schemes. In Section 4-B we prove the existence of codes if the cooperation matrix represents a so-called **compatible graph**.

Before going into details of the construction, we look at an example to obtain some intuition.

**Example 4. Recall the DSN in Example 1.** We present a coding scheme with the cooperation matrix specified in the left part (2) of Fig. 5. The non-systematic part of the generator matrix is shown in (4), which is obtained through the following process:

1. Partition all the non-zero-non-one elements into structured groups, each of which is marked by either a rectangle or a polygon in \( D \), as indicated in the left part of Fig. 5.
2. Replace the endpoints of each horizontal line segment in Step 1 with \( s \in S \) (\( S \) is a set of symbols), as indicated in the right part of Fig. 5.
3. Assign a parameter \( \gamma_{s} \in \mathbb{N} \) to each \( s \in S \), and a matrix \( B_{s} \in \mathbb{F}_q^{b \times \gamma_{s}} \) to any \( (i,j), s \) such that \( X_{i,j} = s \);
4. For each \( i \in [p], l \in [L_i] \) let \( |\eta \gamma_{s} |_{t} |_{n} |_{m} \), assign \( V_{t,n} \in \mathbb{F}_q^{n \times \gamma_{s}} \), to \( v_s \); let \( B_{s} = \left\{ B_{t,s} \right\}_{d_{s} = d_{s}} \), assign \( A_{i,j} = B_{s} \), \( V_{d,l} \) for \( s = X_{i,j}, l = D_{i,j} \).

Let us again focus on node \( v_2 \). Let \( l_{2} = \{ v_1, v_3, v_5 \}, l_{2} = \{ v_8, v_9 \}, l_{2} = \{ v_{10}, v_{11} \} \). Then, \( B_{s} = \{ v_4, v_6, v_8 \}, B_{s} = \{ v_4, v_6, v_8 \}, B_{s} = \{ v_4, v_6, v_8 \} \), \( \delta_{2} = \delta_{2} = \delta_{2} = \delta_{2} + \delta_{2} + \delta_{2} + \delta_{2} + \delta_{2} + \delta_{2} + \delta_{2} + \delta_{2} + \delta_{2} \). We first show that knowing \( \{ m_1 \}_{i \in \{1,2\}} \) is sufficient for removing \( s_2 = \sum_{t \in I_1} m_{1,t} B_{s} U_{2} + \sum_{t \in I_3} m_{1,t} B_{s} V_{2} \), from the parity part of \( c_{2} \). Note that if \( A_{i,j}, U_{i} \) and \( \{ V_{t,n} \}_{t \in \{2,3\}} \) are linearly independent, then for all \( l, \sum_{t \in I_1} m_{1,t} B_{s} X_{1}^{l} \), is recoverable if \( m_i \) is recoverable. In our example, this means that \( \{ m_{1,j} \}_{j=1,3,5} \), \( m_{2,J} + m_{2,J}, m_{11} B_{c} \) are known: the sum of them is exactly \( s_2 \). Therefore, \( s_2 \) is removed through the 1-st level cooperation.

We next show that additional parities are obtained through l-th level cooperations with \( l = 2, 3 \).

In the 2-nd level cooperation, \( m_8, m_9 \) are known. Therefore, \( m_{2,J} + m_{2,J} + m_{2,J} + m_{2,J} \) is also known. We remove \( m_{3,J} \) that is obtained via \( v_3 \). In order to obtain the \( \gamma_{s} \) parities from \( m_{2,J} \), one needs \( m_{1}, m_{6} \) to be recoverable. Therefore, \( B_{2} = \{ v_4, v_6, v_8 \}, \delta_{2} = \delta_{2} + \delta_{2} \).

As shown in the first step in Example 4, the cooperation matrix adopts a partition of non-zero-non-one elements into groups where each of them forms a cycle. Suppose there are \( T \) cycles. Represent each cycle with index \( t \in [T] \) by a tuple \( \{ X_t, Y_t, \{ X_{t,j} \}_{j \in Y_t}, \{ Y_{t,i} \}_{i \in X_t}, l_t \} \), where \( X_t, Y_t \) denote the indices of the rows and the columns of the cycle, respectively; \( l_t \) denotes the number assigned to the vertices of the cycle; \( X_{t,j} = \{ i_1, i_2 \} \) for \( j \in Y_t \) and \( (i_1, j), (i_2, j) \) are the vertices of cycle \( C_t; Y_{t,i} = \{ j_1, j_2 \} \) for \( i \in X_t \) and \( (i_1, i), (i_2, i) \) are the vertices of cycle \( t \). For example, let \( t = 1 \) for the blue cycle at the bottom left part of the matrices in Fig. 5. Then, it is represented by \( \{ 10, 11, 12 \}, \{ 4, 5, 6 \}, \{ X_{t,j} \}_{j \in \{4,5,6\}}, \{ Y_{t,i} \}_{i \in \{10,11,12\}} \).
where $X_{1;4} = \{10, 11\}$, $X_{1;5} = \{11, 12\}$, $X_{1;6} = \{10, 12\}$, $Y_{1;10} = \{4, 6\}$, $Y_{1;11} = \{4, 5\}$, $Y_{1;12} = \{5, 6\}$.

Observe that cycle $t \in [T]$ in Fig. 5 essentially represents a pair of disconnected edges or triangles with vertices from $X_t$ and $Y_t$. We mark $X_t$, $Y_t$, draw a directed edge with label $l$ from $X_t$ to $Y_t$ for each $t \in [T]$, and obtain the so-called cooperation graph. The cooperation graph for the coding scheme in Example 3 is shown in Fig. 6.

B. Construction Over Compatible Graphs

We have defined the notion of cooperation graphs in Section 4. Observe that the cooperation graph shown in Fig. 6 satisfies a set of conditions that define the so-called compatible graph. We show in Theorem 2 the existence of a hierarchical coding scheme with cooperation graph $G$ if $G$ is a compatible graph. The coding scheme is presented in Construction 2.

Definition 3. Let $G$ be a cooperation graph on $G(V, E)$ that is represented by $(X_t, Y_t, \{X_{t;i}\}_{i \in Y_t}, \{Y_{t;i}\}_{i \in X_t}, t \in [T])$, and let the $l$-st level cooperation graph be $G_l$. Let $M_l$ denote the set of nodes that have an outgoing edge pointing at $v_i$ in $G_l$. For each $v_i \in V$, $l \in [L_i]$, let $R_{l;i} = \{i : i \in Y_t, t \in [T], t_l = l\}$, $T_{l;i} = \{i : i \in R_{l;i} \cap T_{l;i}\}$, and $V_{l;i} = \bigcup_{t \in T_{l;i}} Y_t$. We call $G$ a compatible graph on $G$ if the following conditions are satisfied:

1) For each $v_i \in V$, sets in $\{Y_{t;i}\}_{i \in R_{l;i}}$ are disjoint;

2) For each $v_i \in V$, $l \in [L_i]$, and any node $v_j$ such that $j \in V_{l;i} \setminus \{v_i\}$, $V_{l;ij} \subseteq M_{l;i}$.

Note that in Definition 3 $R_{l;i}$ consists of indices of all the cycles that define a level $l$ cooperation at node $v_i$; $T_{l;i}$ consists of indices of all the cycles that provide extra parities in the $l$-th level cooperation of node $v_i$; $V_{l;i}$ consists of all the nodes that are required to be recovered in order to obtain the cross parities at $v_i$ resulting from the $l$-th level cooperation.

Construction 2. Let $G(V, E)$ represent a DSN with parameters $(n, k, r)$. Suppose $G$ is a compatible graph of depth $L$ on $G$, with parameters $(X_t, Y_t, \{X_{t;i}\}_{i \in Y_t}, \{Y_{t;i}\}_{i \in X_t}, t \in [T])$, and the $1$-st level cooperation graph is denoted by $G_1$ (other necessary parameters are as they are in Definition 3).

Fig. 6: Cooperation graph of Example 3

Fig. 7: Proof of Theorem 2

Let $\delta$ be the 1-st level cooperation parameter. For each $v_i \in V$, $1 \leq l \leq L_i$, and any $t \in T_{l;i}$, assign cooperation parameter $\gamma_{i;tl}$ to $C_l$; let $\eta_{l;i} = \max_{l \in L_i} \max_{l \in T_{l;i}} \gamma_{l;i}$. Let $u_i = k_t + \delta_t + \sum_{j=1}^{L_i} \eta_{l;i}$. Let $a_{i,s} = \min \{a_{i,s}, b_{i,s}\}$, for all $i, j, l, s \in [n]$. Then, we obtain $M_{l;i}, B_{l;i}, E_{l;i}, U_{l;i}, V_{l;i}$, for $i \in [p]$, $j \in p \setminus \{i\}$, according to the following partition of $T_l$.

Matrix $G$ in (4) is assembled as follows. Consider the Cauchy matrix $T_i$ on $GF(q^{u_i \times v_i})$, such that $T_i = \sum_{j=1}^{u_i} \left(\begin{array}{c}a_{i,j}1 
\end{array}\right)$, for $i \in [p]$. Then, we obtain $A_{i,j}, B_{i,j}, E_{i,j}, U_{i,j}, V_{i,j}$, for $i \in [p], j \in p \setminus \{i\}$, according to the following partition of $T_i$.

Theorem 2. The code $C_2$ has ECC hierarchies $d_i = \left(d_{i,0}, d_{i,1}, \ldots, d_{i,L_i}\right)$, for all $v_i \in V$, where $d_{i,0} = r_i - \delta_i - \sum_{i=1}^{L_i} \eta_{l;i}$, $d_{i,1} = r_i + \sum_{j \in M_{l;i}} \delta_j + \sum_{2 \leq l \leq L_i} \gamma_{l;i}$. Moreover, $T_i = M_{l;i}, B_{i}^1 =$
This is done if any one of the two nodes in $\sum_{l} I_l$ as "I" each cycle, which is \textit{operation with nodes in $\sum_{l} I_l$}. Therefore, $s_{i,l}$ is recoverable because $\{V_{i,l}\}_{2 \leq l \leq L}$ have linearly independent rows. Then on the other hand, we first calculate the sum of the cross parities on the vertices of each cycle, which is 0 since the cross parities at the endpoints of each horizontal edge of any cycle are identical, and thus sum up to zero. Therefore, $s_{i,l} + \sum_{V_{i,l} \neq \emptyset} s_{j,l}$ should also be zero, which indicates that $s_{i} = \sum_{V_{i,l} \neq \emptyset} s_{j,l}$.

Moreover, for $2 \leq l \leq L_{i}$, any cycle $C_{l}$ with index $l \in T_{i}$ has potential to provide additional $\gamma_{i,l}$ in $l$-th level cross parities to $v_{i}$. Therefore, $d_{i,l} = r_{i} + \sum_{j \in M_{i}} \delta_{j} + \sum_{2 \leq l' \leq l} \gamma_{i,l'}$, which is done if any one of the two nodes in $Y_{l,j} = \{j,j\}$ obtains its $\gamma_{i,l'}$ cross parities at $v_{i}$ through its $l_{i}$-th level cooperation. However, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities is that all other cross parities generated from nodes lying in the $l_{i}$-th level cooperation are calculated and subtracted from the $l_{i}$-th level parities of $v_{j}$, i.e., $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Similarly, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities at $v_{i}$ is described as $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Therefore, the overall requirement is stated as $\gamma_{l} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Similarly, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities at $v_{i}$ is described as $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Similarly, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities at $v_{i}$ is described as $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Similarly, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities at $v_{i}$ is described as $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Similarly, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities at $v_{i}$ is described as $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$. Similarly, the condition of $v_{j}$ successfully calculating these $\gamma_{i,l'}$ cross parities at $v_{i}$ is described as $I_{l,i} \setminus A_{l} \subseteq \{v_{i}\} \cup W$.

**V. CONCLUSION**

Hierarchically locally accessible codes in the context of centralized networks have been discussed in various prior works, whereas those of DSNs (no prespecified topology) have not been explored. In this paper, we presented a topology-aware cooperative data protection scheme for DSNs, which is an extension of, and subsumes, our previous work on hierarchical coding for centralized distributed storage. Our scheme achieves faster recovery speed compared with existing network coding methods, and corrects more erasure patterns compared with our previous work. Our work exploits the power of the blockchain technology in DSNs, and it has potential to be applied in WSNs.

**ACKNOWLEDGMENT**

This work was supported in part by NSF under the grants CCF-BSF 1718389 and CCF 1717602, and in part by AFOSR under the grant FA 9550-17-1-0291.

**REFERENCES**

[1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” Manubot, Tech. Rep., 2019.
[2] M. Crosby, P. Pattanayak, S. Verma, V. Kalyanaraman et al., “Blockchain technology: Beyond bitcoin,” Applied Innovation, vol. 2, no. 6-10, p. 71, 2016.
[3] V. Bagaria, S. Kannan, D. Tse, G. Fanti, and P. Viswanath, “Deconstructing the blockchain to approach physical limits,” 2018. [Online]. Available: https://arxiv.org/abs/1810.08092
[4] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. J. Wainwright, and K. Ramchandran, “Network coding for distributed storage systems,” IEEE Trans. Inf. Theory, vol. 56, no. 9, pp. 4539–4551, Sept 2010.
[5] Z. Kong, S. A. Aly, and E. Soljanin, “Decentralized coding algorithms for distributed storage in wireless sensor networks,” IEEE Journal on Selected Areas in Communications, vol. 28, no. 2, pp. 261–267, 2010.
[6] M. Ye and A. Barg, “Cooperative repair: Constructions of optimal mds codes for all admissible parameters,” IEEE Transactions on Information Theory, vol. 65, no. 10, pp. 6072–6088, 2019.
[7] H. Hou, P. P. Lee, K. W. Shum, and Y. Hu, “Rank-aware regenerating codes for data centers,” IEEE Transactions on Information Theory, 2019.
[8] Z. Chen and A. Barg, “Explicit constructions of msr codes for clustered distributed storage: The rack-aware storage model,” 2019. [Online]. Available: https://arxiv.org/abs/1901.04419
[9] N. Prakash, V. Abdeshahitov, and M. Médard, “The storage versus repair-bandwidth trade-off for clustered storage systems,” IEEE Transactions on Information Theory, vol. 64, no. 8, pp. 5783–5805, 2018.
[10] J. Li, S. Yang, X. Wang, and B. Li, “Tree-structured data regeneration in distributed storage systems with regenerating codes,” in 2010 Proceedings IEEE INFOCOM. IEEE, 2010, pp. 1–9.
[11] D. Pernas, C. Yuen, B. Gastón, and J. Pujol, “Non-homogeneous two-rack model for distributed storage systems,” in 2013 IEEE International Symposium on Information Theory. IEEE, 2013, pp. 1237–1241.
[12] Y. Wang, D. Wei, X. Yin, and X. Wang, “Heterogeneity-aware data regeneration in distributed storage systems,” in IEEE INFOCOM 2014-IEEE Conference on Computer Communications. IEEE, 2014, pp. 1878–1886.
[13] A. M. Ibrahim, A. A. Zewail, and A. Yener, “Green distributed storage using energy harvesting nodes,” IEEE Journal on Selected Areas in Communications, vol. 34, no. 5, pp. 1500–1603, 2016.
[14] M. Sipos, J. Gahn, N. Venkat, and D. Oran, “Network-aware feasible repairs for erasure-coded storage,” IEEE/ACM Transactions on Networking, vol. 26, no. 3, pp. 1404–1417, 2018.
[15] ———, “Erasure coded storage on a changing network: The untold story,” in 2016 IEEE Global Communications Conference (GLOBECOM). IEEE, 2016, pp. 1–6.
[16] S. Yang, A. Haredy, R. Calderbank, and L. Dolecek, “Hierarchical coding to enable scalability and flexibility in heterogeneous cloud storage,” in IEEE Global Communications Conference (GLOBECOM), 2019. [Online]. Available: https://arxiv.org/abs/1905.02279
[17] Y. Zhu, C. Lv, Z. Zeng, J. Wang, and B. Pei, “Blockchain-based decentralized storage scheme,” in Journal of Physics: Conference Series, vol. 1237, no. 4. IOP Publishing, 2019, p. 042008.
[18] S. Underwood, “Blockchain beyond bitcoin,” 2016.