IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 15 November 2023, accepted 21 December 2023, date of publication 28 December 2023,
date of current version 4 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3347928

RESEARCH ARTICLE

Data Dissemination Framework Using Low-Rank Approximation in Edge Networks

JUNGMIN KWON[1], (Graduate Student Member, IEEE), AND HYUNGGON PARK[1,2], (Senior Member, IEEE)

[1]Department of Electronic and Electrical Engineering, Ewha Womans University, Seoul 03760, South Korea
[2]Smart Factory Multidisciplinary Program, Department of Electronic and Electrical Engineering, Ewha Womans University, Seoul 03760, South Korea

Corresponding author: Hyunggon Park (hyunggon.park@ewha.ac.kr)

This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) funded by the Korean Government (MSIT) through the Development of Distributed/Cooperative AI-Based 5G+ Network Data Analytics Functions and Control Technology under Grant 2021-0-00739.

ABSTRACT This paper proposes a reliable data dissemination framework for edge networks, leveraging network coding combined with low-rank approximation. We consider an edge network that consists of a server and power-limited mobile devices, where the data is broadcasted by the server. In such networks, broadcasted data may be lost due to poor channel conditions or the interference caused by the mobility of edge mobile devices, particularly without a retransmission mechanism. This can cause application errors in edge devices, lower the Quality of Service (QoS), and compromise network stability. To overcome these challenges, we propose a framework for reliable edge networks in broadcasting without retransmissions. The edge network reliability can be achieved by the approximate decoding of broadcasted data. In the proposed framework, the edge server employs matrix factorization to encode data with principal components, ensuring a lower decoding error rate even with potential packet losses. Furthermore, the proposed framework can shift the computational complexity from mobile edge devices to the edge server using the low-rank approximation at the decoding stage, effectively mitigating power limitations on mobile devices. Through theoretical analysis, we demonstrate that the proposed algorithm outperforms typical broadcasting in terms of decoding accuracy, and present an upper bound error rate for the proposed algorithm. The simulation results confirm that the proposed algorithm outperforms other state-of-the-art algorithms in terms of decoding accuracy, time delay, and complexity.

INDEX TERMS Data dissemination, edge computing, edge network, network coding, low-rank approximation, decoding accuracy, low computational complexity, complexity shift.

I. INTRODUCTION

The significant amount of data and variety of services has led to an increased demand for innovative sixth-generation wireless technology (6G) networks. To ensure reliability while effectively managing the heterogeneous and large volumes of data, storing and processing data at the edge of the network has emerged as an attractive solution [1], [2]. Especially, edge computing is gaining prominence in 6G as it may include artificial intelligence (AI) native architectures by effectively integrating and utilizing AI models and data [3], [4], [5]. In such edge networks, reliable data dissemination is a fundamental requirement for transmitting large volumes of data to edge devices. A variety of services, including vehicular communications, extended reality (XR), and mobile video streaming, are expected to benefit from reliable data dissemination. This approach can be supported by a centralized infrastructure, where a central server establishes broadcasting connections with optimal control to achieve low latency and high efficiency [6]. However, the data dissemination system in an edge network can be highly susceptible to the network dynamics incurred by user

The associate editor coordinating the review of this manuscript and approving it for publication was Hang Shen.

activities and the unexpected channel conditions, which can compromise the reliability of data transmission [7], [8], [9]. This becomes an even more critical issue for broadcasting, in particular, without a retransmission mechanism. Packet loss can incur data errors or data loss in edge devices, triggering the straggling effect. This effect arises when a single missing data induces significant delays in the overall process. Therefore, it is crucial to implement reliable data dissemination to prevent data loss or errors at edge devices in broadcasting protocols.

Several reliability assurance solutions for data dissemination in edge networks have been developed. In [10], [11], [12], and [13], the probabilistic retransmission approach is considered by retransmitting each data with a given probability. In [14], packet-by-packet parallel processing is proposed by utilizing repeated transmission for low latency and reliable services. In [15], a compressed broadcasting approach with retransmission is proposed for federate learning, where it controls the number of packets to transmit in an error-prone wireless network. In [16], a data-importance aware retransmission protocol is proposed which adapts retransmission decisions on both data importance and reliability. The main idea of these studies is utilizing adaptive additional transmissions. However, many dissemination strategies present a common design pattern, mainly characterized by a retransmission policy, which is not applicable in broadcasting. Specifically, in broadcasting, employing retransmission can lead to excessive packet duplication, packet collisions, and network contention, ultimately resulting in the broadcast storm problem.

Unlike the retransmission-based data reliable dissemination approaches, network coding (NC) with broadcasting can be a solution to reliable and efficient data transmission [17], [18], [19]. NC technique can dramatically simplify the individual retransmission process into the broadcasting of equal data dissemination as NC can make packets anonymous [20]. The systematic network coding (SNC) [17] is proposed in a one-hop broadcasting network, which simply transmits additional packets encoded by the random linear network coding (RLNC) and enables each receiver node to recover the missing original packets. An SNC-based real-time multimedia streaming system is proposed in [21] to cope with dynamic channel conditions by adaptively determining the code rate based on the packet loss rate. In this way, the reliability of data dissemination in broadcasting can be provided in the application layer by accurately decoding multiple NC data.

Unfortunately, in practice, implementing NC in edge networks is a non-trivial task because of two critical drawbacks: high computational complexity and an 'all-or-nothing' problem.[1] The computational complexity becomes a critical issue for resource-limited edge devices, as NC often resorts to a significantly large amount of computation to the devices in mobile wireless networks (e.g., battery-powered devices). This is because mobile devices may suffer from significant computational complexity during the decoding process, even with only a few coded packets [23]. Furthermore, the computational complexity of NC in the decoding process can cause a throughput bottleneck in comparison to the achievable speed of a wireless local area network interface [24]. The computational complexity associated with NC data is determined by how the data are combined at the encoder. For example, random matrix multiplication is required for RLNC in the encoding process, and the decoding process at the receiver needs matrix inversion to recover the original packets. Gaussian elimination is one of the well-known approaches for matrix inversion in the RLNC decoding process. However, Gaussian elimination in NC generally requires high computational complexity and even causes decoding failure because of the matrix singularity. This issue also ties in with the second challenge of NC which is an all-or-nothing problem. The burst of packet loss can cause an all-or-nothing problem in NC. This is because the received NC data cannot guarantee the matrix inversion due to matrix singularity during the decoding process.

To overcome these NC drawbacks, special constructions of a coding matrix in the encoding process have been proposed. Table 1 presents the comparison of the algorithms and key findings of these studies. In [25], a weighted Vandermonde echelon fast coding scheme is proposed to reduce the dependence problem of the NC matrix with less computational overhead. Similarly, the sliding NC streaming code for ultra-reliable low-latency communication (URLLC) using RLNC is proposed in [26]. The authors reduced the decoding complexity compared to typical RLNC by skipping either the forward elimination or back substitution of Gaussian elimination. A low-complexity coding scheme is proposed in [27], where packets are encoded by random subsets sequentially formed from source packets. In this algorithm, sparse coding is used to achieve high encoding and decoding efficiency while they allow controlling the complexity. In [28], the Fulcrum sliding window coding is proposed with low computational complexity in the binary field to avoid in-order packet delay. As these approaches include RLNC operation, however, the decoding process still requires computationally inefficient matrix inversion approaches. Therefore, to expedite the realization of a reliable yet robust NC in edge networks, it is essential to develop new coding processes that embed simple and reliable decoding processes.

Our goal is to improve the reliability of data dissemination with low complexity, even when retransmissions are avoided in broadcasting. Therefore, we focus on achieving better Quality of Service (QoS) in data dissemination by enhancing decoding accuracy without retransmissions. For this, we propose a reliable data dissemination framework based on the low-rank approximation in edge networks. The proposed framework involves splitting the data at the

[1] A receiver cannot recover any information from received data, unless it receives at least the same number of innovative packets as are originally combined together [22].

TABLE 1. A comparison of the NC algorithms.

Ref.	Coding Matrix	Decoding	Advantages	Disadvantages
[25]	Vandermonde echelon fast coding	Reduce forward elimination of Gaussian elimination	Reduce dependence problem of NC matrix	Still require high complexity in Gaussian elimination process
[26]	Sliding streaming coding	Skip either the forward elimination or back substitution of Gaussian elimination	Low complexity	Matrix singularity
[27]	Random subsets coding	Gaussian elimination with reduced dimension	Allow complexity controlling	Still require high complexity in Gaussian elimination process, Matrix singularity
[28]	Fulcrum sliding window coding	Gaussian elimination with computation in binary field	Avoid in-order packet delay	Matrix singularity

edge server via matrix decomposition and merging them at the edge devices using low-rank approximation. The matrix decomposition includes singular value decomposition (SVD) to factorize the data into singular values and unitary matrices. This can improve the reliability of dissemination by differentiating the priority of data and facilitate real-time data decoding by ensuring the early recovery of crucial data. Hence, the proposed algorithm makes the best effort to receive packets containing significant information first. Moreover, the proposed framework can shift the complexity burden from mobile devices to edge servers, relieving the power constraints of the mobile devices. This is because the low-rank approximation of the proposed approach can simplify the decoding process, significantly reducing the decoding complexity than existing NC decoding algorithms.

We analytically prove that the proposed algorithm outperforms typical broadcasting, even in the worst-case scenario where the collected data are independent and identically distributed. In addition, we study the bound of decoding accuracy with the reception of the decodable packets over time. An extensive set of experiments confirms that the reliability of the proposed algorithm performs better than other algorithms in both synthetic data and real-world data. Furthermore, we confirm that the proposed algorithm reduces the decoding computational complexity compared to other state-of-the-art NC algorithms.

The main contributions of this paper are summarized as follows:

- We address the challenges in data dissemination of edge networks where packet loss is prevalent but retransmission is not allowed. We propose a data dissemination framework to support reliability by reducing decoding error rates, time delay, and computational complexity for mobile edge devices.
- We leverage network coding in conjunction with low-rank approximation to approximately decode lost packets, which improves decoding accuracy and facilitates real-time data decoding on mobile edge devices, ensuring the prompt recovery of crucial data.
- We theoretically show that the proposed algorithm outperforms typical broadcasting algorithms in terms of decoding accuracy and find the analytical upper bound of decoding error rate as a function of the number of decoded packets over time.

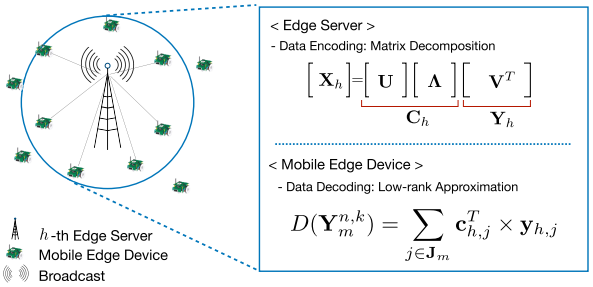


FIGURE 1. Illustrative examples of the edge network consisting of an edge server and multiple mobile edge devices.

- We assess the performance of the proposed algorithm in terms of decoding error rate, time delay, and computational complexity via a comprehensive set of simulations. Results show that the proposed algorithm not only reduces decoding errors but also offloads computational complexity from edge devices to servers and facilitates real-time data dissemination.

The rest of our paper is organized as follows. A system setup and problem formulation in the edge network are described in Section II. Section III provides the proposed data dissemination framework and corresponding procedures for the broadcasting protocol. Section IV demonstrates that the proposed algorithm outperforms the typical broadcasting approach in terms of decoding accuracy. An extensive set of simulation results is provided in Section V. Finally, we draw conclusions in Section VI.

II. SYSTEM SETUP AND PROBLEM FORMULATION

We consider a mobile network consisting of edge servers and massive mobile edge devices. In this network, a set of mobile edge devices are connected directly to the nearest edge server via wireless links and receive data packets. An illustrative example of the considered network is shown in Fig. 1.

An edge server is denoted by $g_h (h \in \mathbf{H})$ and a mobile edge device associated with the edge server g_h is denoted by $d_h(m)$ $(m \in \mathbf{M})$, where $\mathbf{H}$ and $\mathbf{M}$ correspond to the index sets of edge servers and the associated mobile edge devices, respectively. The edge servers aim to efficiently disseminate the data to the associated mobile edge devices with equal opportunities. We assume that an edge server disseminates

data in one transmission at a single time step. The features of the edge servers and the mobile edge devices are briefly summarized below.

- Edge servers: The edge servers employ an NC-based data encoding algorithm that utilizes matrix decomposition, and disseminates the encoded data.
- Mobile edge devices: Mobile edge devices receive the data that is encoded by the associated edge server. The mobile edge devices proceed with the decoding process of NC-encoded data and recover the original data using low-rank approximation. The mobile edge device cannot request packet retransmission because we consider the data broadcasting without acknowledgment.

Let $\mathbf{X}_h(\in \mathbb{R}^{N\times L}) = [\mathbf{x}_{h,1}^T, \cdots, \mathbf{x}_{h,N}^T]^T$ be a source dataset that is encoded together and disseminated to all the associated mobile edge devices, where $\mathbf{x}_{h,j} = [x_{h,j}(1), \cdots, x_{h,j}(L)]$ be the data with length L. The number of source data elements N that are combined by NC in each transmission is referred to as the *encoding number* [29]. Let $\mathbf{Y}_m^{n,k}(\in \mathbb{R}^{n\times L}) = \bigcup_{j\in\mathbf{J}_m} \mathbf{y}_{h,j}^T$ be a set of encoded data packets received at $d_h(m)$ by time k, where $\mathbf{y}_{h,j}^T$ is a packet of encoded data, and $\mathbf{J}_m$ correspond to the index set of received data packets at $d_h(m)$. We denote the number of elements of $\mathbf{J}_m$ as $|\mathbf{J}_m|$. Then, the number of elements $\mathbf{Y}_m^{n,k}$ are n (i.e., $|\mathbf{J}_m| = n$), and this implies that if $n < k$, $(k-n)$ packets have been lost. We assume that the packets are adequately protected against interference from other servers using several interference management approaches [30], [31], [32]. Therefore, packet loss is mainly attributed to device mobility or inter-symbol interference in this paper. We set a decoding function at the edge device as $D(\cdot)$ that partially decodes data from received packets and the decoded data as $D(\mathbf{Y}_m^{n,k})$.

As an evaluation of the decoding accuracy in the proposed data dissemination frameworks, we use the Normalized Root Mean Square Error (NRMSE). This facilitates the comparison between the datasets with different scales. Since we disseminate the source dataset through encoded data packets $\mathbf{y}_{h,j}^T$, we calculate the NRMSE of the cumulative decoded data observed over time. Therefore, we define the NRMSE estimator at time k for the dataset as

$$\mathcal{L}(D(\mathbf{Y}_m^{n,k})) = \frac{1}{\|\mathbf{X}_h\|_F}\|\mathbf{X}_h - D(\mathbf{Y}_m^{n,k})\|_F, \tag{1}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, which is the Euclidean norm of the matrix. Here, the estimator can measure the error between recovered data and source dataset from the received packets at time k, even when all packets are not perfectly received due to packet loss. Our goal is to minimize the $\mathcal{L}(D(\mathbf{Y}_m^{n,k}))$ at mobile edge devices for all time steps.

III. DATA DISSEMINATION FRAMEWORK

A. ENCODING WITH MATRIX DECOMPOSITION

In the proposed system, the data $\mathbf{X}_h$ is encoded based on the NC performed in the field of real numbers ($\mathbb{R}$). The encoding algorithm is based on a matrix decomposition for data splitting, which generates unitary matrices. A set of data $\mathbf{X}_h$ at an edge server g_h can be decomposed by SVD into

$$\mathbf{X}_h = \mathbf{U}\Lambda\mathbf{V}^T = \sum_{j=1}^{N} \lambda_j \mathbf{u}_j \mathbf{v}_j^T, \tag{2}$$

where $\mathbf{U}(\in \mathbb{R}^{N\times N}) = [\mathbf{u}_1, \cdots, \mathbf{u}_N]$ and $\mathbf{V}(\in \mathbb{R}^{L\times N}) = [\mathbf{v}_1, \cdots, \mathbf{v}_N]$ are both unitary matrices, and $\Lambda(\in \mathbb{R}^{N\times N})$ is a diagonal matrix with singular values. Here, the unitary matrices satisfy $\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I}$ with identity matrix $\mathbf{I}$, and singular values satisfy $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \geq 0$.

Given $\mathbf{U}$, Λ, and $\mathbf{V}^T$, the proposed encoding algorithm generates the coding coefficient matrix $\mathbf{C}_h(\in \mathbb{R}^{N\times N}) = [\mathbf{c}_{h,1}^T, \cdots, \mathbf{c}_{h,N}^T]^T$ and the corresponding encoded data $\mathbf{Y}_h(\in \mathbb{R}^{N\times L}) = [\mathbf{y}_{h,1}^T, \cdots, \mathbf{y}_{h,N}^T]$ as $\mathbf{C}_h = \mathbf{U}^T$ and $\mathbf{Y}_h = \Lambda \times \mathbf{V}^T$. The encoded data $\mathbf{Y}_h$ is then encapsulated as a packet $\mathbf{Z}_h$ with coding coefficient matrix $\mathbf{C}_h$ by appending each row, i.e., $\mathbf{Z}_h = \left[\mathbf{C}_h \middle| \mathbf{Y}_h\right]$, where $\mathbf{Z}_h(\in \mathbb{R}^{N\times(N+L)}) = [\mathbf{z}_{h,1}^T, \cdots, \mathbf{z}_{h,N}^T]^T$. For simplicity, we assume that the link capacity is a single packet size such that a server transmits a single packet per unit of time [33].

In the design of packets, we can arrange the packet $\mathbf{Z}_h = [\mathbf{z}_{h,1}^T, \cdots, \mathbf{z}_{h,N}^T]^T$ in the order of priority because singular values determined by SVD are in descending order. Specifically, $\mathbf{z}_{h,1}$ is the most informative, i.e., it contains most of the information of encoded data compared to $\mathbf{z}_{h,j}$ $(j > 1)$. Therefore, the proposed algorithm can maximize the decoding accuracy by simply receiving data packets from $\mathbf{z}_{h,1}$ to $\mathbf{z}_{h,N}$.

B. DECODING WITH LOW-RANK APPROXIMATION

For the broadcasted data packet $\mathbf{z}_{h,j}$ that each mobile edge device receives, an edge device decapsulates $\mathbf{z}_{h,j}$ into $\mathbf{c}_{h,j}$ and $\mathbf{y}_{h,j}$. Let $\mathbf{C}_m^{n,k} = \bigcup_{j\in\mathbf{J}_m} \mathbf{c}_{h,j}^T$ be the received set of coefficient matrix elements. The decoded data $D(\mathbf{Y}_m^{n,k})$ can be recovered by low-rank approximation as $D(\mathbf{Y}_m^{n,k}) = (\mathbf{C}_m^{n,k})^{-1} \times \mathbf{Y}_m^{n,k}$. Since the coding coefficient matrix is the transpose of a unitary matrix, the inverse of $\mathbf{C}_m^{n,k}$ can be simply obtained by its conjugate transpose, i.e., $(\mathbf{C}_m^{n,k})^{-1} = (\mathbf{C}_m^{n,k})^T$. This means that the data recovery does not require computing the inverse of the matrix. Rather, the decoding process can be significantly simplified as the transpose of the matrix can be directly used for the matrix inverse. Finally, the approximately decoded source dataset $D(\mathbf{Y}_m^{n,k})$ can be determined as

$$D(\mathbf{Y}_m^{n,k}) = \sum_{j\in\mathbf{J}_m} \mathbf{c}_{h,j}\mathbf{y}_{h,j} = \sum_{j\in\mathbf{J}_m} \lambda_j \mathbf{u}_j \mathbf{v}_j^T. \tag{3}$$

C. COMPLEXITY ANALYSIS

For the complexity analysis of the proposed algorithm, we denote Δ_{en}^* and Δ_{de}^* as the computational complexity associated with the encoding and decoding processes, respectively. Since matrix decomposition is the main operation of the proposed algorithm in the encoding process, the dominant time complexity is incurred by the operations for SVD, which

is given as $\Delta_{\text{en}}^* = O(2N^2L + N^3 + N + NL)$ under the assumption of $N \ll L$ [34]. In the decoding process, the proposed algorithm can decode the received data using low-rank approximation with simple matrix multiplications. Hence, the time complexity associated with the decoding process is given by $\Delta_{\text{de}}^* = O(N^2L)$. Then, the overall time complexity associated with encoding and decoding processes can be approximately expressed as

$$\begin{aligned}\Delta_{\text{en}}^* + \Delta_{\text{de}}^* &\approx O(2N^2L + N^3 + N + NL) + O(N^2L)\\ &\approx O(3N^2L + N^3 + N + NL).\end{aligned}$$

For comparison, we consider the computational complexity of RLNC, which is a type of NC commonly used in practice. The complexity of the encoding process in RLNC is determined by matrix multiplication and expressed as $\Delta_{\text{en}} = O(N^2L)$. Since the decoding process is based on the matrix inversion, which can be implemented by Gaussian elimination, the corresponding complexity for the decoding process is given by $\Delta_{\text{en}} = O(N^3L)$. Therefore, the overall complexity of the RLNC can be approximately estimated by

$$\Delta_{\text{en}} + \Delta_{\text{de}} \approx O(N^2L) + O(N^3L) \approx O(N^3L).$$

In summary, the proposed algorithm can reduce the computational complexity of not only the decoding process but also the overall process, approximately from $O(N^3L)$ to $O(N^2L)$.

D. PROCEDURES OF PROPOSED EQUAL DATA DISSEMINATION

The proposed data dissemination algorithm can be implemented by the two procedures of network registration and data dissemination.

1) NETWORK REGISTRATION

The network registration phase includes configuring groups for data dissemination, where a group consists of one edge server and several mobile edge devices. In this phase, the edge server forms groups of mobile edge devices that are within the transmission range. Hence, once the list of mobile edge devices is set, the edge server does not change the list of associated devices until the next network registration. The edge device that completes the network registration is able to receive the first packet without fail.

2) DATA DISSEMINATION

In the data dissemination process, the edge server disseminates the data to the edge devices. Each mobile edge device then attempts to decode the data.

IV. ANALYSIS OF DECODING ACCURACY

In this section, we analytically study the decoding accuracy by deploying the proposed system.

A. COMPARISON OF DECODING ACCURACY

We perform a comparison of decoding accuracy between the proposed dissemination algorithm and typical broadcasting, where k data are sequentially received at mobile edge devices. In typical broadcasting, the mobile edge devices receive the partial data of $\mathbf{X}_h$ as the edge server disseminates the data unencoded. Let $\mathbf{I}_k$ be a diagonal matrix in which the diagonal elements are equal to one at the indices corresponding to $\mathbf{J}_m$, while the rest of the elements are zero. We define the decoded data packets with typical broadcasting at edge device $d_h(m)$ by time k as $\mathbf{X}_m^{k,k} (\in \mathbb{R}^{N\times L}) = \mathbf{I}_k\mathbf{X}_h$, where $|\mathbf{J}_m| = k$. Then, we denote the received data packets in the proposed broadcasting and typical broadcasting at time k without any packet loss as $\mathbf{Y}_m^{k,k}$ and $\mathbf{X}_m^{k,k}$, respectively. The singular value of $\mathbf{X}_h$ is denoted as $\Lambda = [\lambda_1, \cdots, \lambda_N]$ and the singular value of $\mathbf{X}_m^{k,k}$ is denoted as $\Lambda' = [\lambda'_1, \cdots, \lambda'_k]$.

In order to demonstrate the superior decoding accuracy of the proposed algorithm, we first present the error performance in Lemma 1. Then, we verify that our algorithm consistently outperforms the typical broadcasting at every time step in Theorem 2.

Lemma 1: Given the packets $\mathbf{Y}_m^{k,k}$ sequentially received by time k, $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ in the proposed algorithm is the sum of the singular values of $\mathbf{X}_h$, excluding the first k largest values.

Proof: The partially decoded data $D(\mathbf{Y}_m^{k,k})$ can be expressed as $D(\mathbf{Y}_m^{k,k}) = \sum_{j=1}^{k} \lambda_j \mathbf{u}_j \mathbf{v}_j^T$ from (3). Then, $L\big(D(\mathbf{Y}_m^{k,k})\big)$ is expressed as

$$\begin{aligned}&\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)\\ &= \frac{1}{\|\mathbf{X}_h\|_F}\|\mathbf{X}_h - D(\mathbf{Y}_m^{k,k})\|_F\\ &= \frac{1}{\|\mathbf{X}_h\|_F}\sqrt{tr\Big(\big(\mathbf{X}_h - D(\mathbf{Y}_m^{k,k})\big)\big(\mathbf{X}_h - D(\mathbf{Y}_m^{k,k})\big)^T\Big)}\\ &= \frac{1}{\sqrt{tr(\Lambda\Lambda^T)}}\sqrt{tr\Big(\big(\sum_{j=k+1}^{N}\lambda_j\mathbf{u}_j\mathbf{v}_j^T\big)\big(\sum_{j=k+1}^{N}\lambda_j\mathbf{u}_j\mathbf{v}_j^T\big)^T\Big)}\\ &= \frac{\sqrt{\sum_{j=k+1}^{N}\lambda_j^2}}{\sqrt{\sum_{j=1}^{N}\lambda_j^2}} = \frac{\sqrt{\sum_{j=1}^{N}\lambda_j^2 - \sum_{j=1}^{k}\lambda_j^2}}{\sqrt{\sum_{j=1}^{N}\lambda_j^2}}.\end{aligned}$$

□

Theorem 2: Given the packets $\mathbf{X}_m^{k,k}$ sequentially received by time k in typical broadcasting, the proposed algorithm consistently shows better decoding accuracy across all time steps compared to typical broadcasting.

Proof: In typical broadcasting, the received subset data of $\mathbf{X}_h$ is denoted as $\mathbf{X}_m^{k,k}$. Accordingly, the decoding accuracy of the received data in typical broadcasting is given by

$$\begin{aligned}&\mathcal{L}\big(\mathbf{X}_m^{k,k}\big)\\ &= \frac{1}{\|\mathbf{X}_h\|_F}\|\mathbf{X}_h - \mathbf{X}_m^{k,k}\|_F\\ &= \frac{1}{\|\mathbf{X}_h\|_F}\sqrt{tr\Big((\mathbf{X}_h - \mathbf{X}_m^{k,k})(\mathbf{X}_h - \mathbf{X}_m^{k,k})^T\Big)}\\ &\overset{(a)}{=} \frac{1}{\|\mathbf{X}_h\|_F}\sqrt{tr\Big(\mathbf{X}_h\mathbf{X}_h^T + \mathbf{X}_m^{k,k}(\mathbf{X}_m^{k,k})^T - 2\langle\mathbf{X}_h, \mathbf{X}_m^{k,k}\rangle_F\Big)}\end{aligned}$$

$$\overset{(b)}{=} \frac{1}{\|\mathbf{X}_h\|_F}\sqrt{tr\left(\mathbf{X}_h\mathbf{X}_h^T + \mathbf{X}_m^{k,k}(\mathbf{X}_m^{k,k})^T - 2(\mathbf{X}_m^{k,k})(\mathbf{X}_m^{k,k})^T\right)}$$
$$= \frac{1}{\|\mathbf{X}_h\|_F}\sqrt{tr\left(\mathbf{X}_h\mathbf{X}_h^T - \mathbf{X}_m^{k,k}(\mathbf{X}_m^{k,k})^T\right)}$$
$$= \frac{1}{\sqrt{tr\left(\Lambda\Lambda^T\right)}}\sqrt{tr\left(\Lambda\Lambda^T - \Lambda'(\Lambda')^T\right)}$$
$$= \frac{\sqrt{\sum_{j=1}^{N}\lambda_j^2 - \sum_{j=1}^{k}\lambda_j'^2}}{\sqrt{\sum_{j=1}^{N}\lambda_j^2}},$$

where $\langle\cdot\rangle_F$ denotes the Frobenius inner product. In the above steps, for (a), we used the fact that $tr(\mathbf{X}_h, \mathbf{X}_m^{k,k}) = tr(\mathbf{X}_m^{k,k}, \mathbf{X}_h) = \langle\mathbf{X}_h, \mathbf{X}_m^{k,k}\rangle_F$, and for (b), we used $\langle\mathbf{X}_h, \mathbf{X}_m^{k,k}\rangle_F = \mathbf{X}_m^{k,k}(\mathbf{X}_m^{k,k})^T$.

Since $\mathbf{\Lambda}$ are the singular values of $\mathbf{X}_h$ and $\mathbf{\Lambda}'$ are the singular values of the $\mathbf{X}_m^{k,k}$ from which rows of $\mathbf{X}_h$ are deleted, it is obvious that $\lambda_j \geq \lambda_j' \geq \lambda_{j+1}$, $(\forall j)$, leading to $\sum_{j=1}^{k}\lambda_j^2 \geq \sum_{j=1}^{k}\lambda_j'^2$ [35]. Therefore, by combining Lemma 1, the proposed algorithm always satisfies $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) \leq \mathcal{L}\big(\mathbf{X}_m^{k,k}\big)$, implying that the proposed algorithm can achieve more accurate data recovery from the received packets. □

The channel packet loss rate of broadcasting can generally be modeled as dropping packets with equal probability independently at all times [36]. Therefore, the decoding accuracy performance can be simply calculated by multiplying the packet loss probability [37]. The decoding accuracy under such scenarios is quantitatively verified through several illustrative simulations in Section V-D.

B. UPPER BOUND OF DECODING ACCURACY

In order to study the worst-case performance of the proposed algorithm, we investigate the upper bound of the $\mathcal{L}(D(\mathbf{Y}_m^{k,k}))$ for the proposed algorithm. To specifically evaluate $\mathcal{L}(D(\mathbf{Y}_m^{k,k}))$ based on the characteristics of the data, we assume that the source dataset $\mathbf{X}_h$ is sampled from zero-mean Gaussian random variables with variance $\sigma^2(1+\Gamma_h)$, i.e., $\mathbf{X}_h \sim \mathcal{N}\big(\mathbf{0}_{N,L}, \sigma^2(1+\Gamma_h)\big)$. Here, $\mathbf{0}_{N,L}$ denotes $(N \times L)$ zero matrix, and Γ_h denotes diagonal matrix with variance weights as $\Gamma_h = \text{diag}(\Gamma_{h,1}, \cdots, \Gamma_{h,N})$, where the weights are arranged in descending order. In this assumption, it is well known that the contribution of $\Gamma_{h,j}$ to singular values is significant only if $\Gamma_{h,j} > \frac{1}{\sqrt{\alpha}}$ with $\alpha = \frac{N}{L}$ (< 1), and is negligible otherwise [38], [39]. Specifically, if

$$\Gamma_{h,1} > \cdots > \Gamma_{h,S} > \frac{1}{\sqrt{\alpha}} > \Gamma_{h,S+1} > \cdots > \Gamma_{h,N},$$

then the singular values estimated from the variance weights from $\Gamma_{h,S+1}$ to $\Gamma_{h,N}$ can be approximated by the singular values estimated from the variance σ^2. This means that $\Gamma_{h,S+1}, \ldots, \Gamma_{h,N} = 0$, and only S variance weights, $\Gamma_{h,1}, \ldots, \Gamma_{h,S}$, need to be considered [38]. This allows us to consider only S effective variance weights. As a result, the upper bound of the decoding accuracy estimated from the variance weights is stated in the Theorem 3.

Theorem 3: For a source dataset $\mathbf{X}_h$ consisting of S effective variance weights, the decoding accuracy of the proposed method, represented by $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$, is bounded by

$$\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) \leq \frac{\sigma^2}{\sqrt{\alpha N}}\sqrt{1 - \frac{\alpha}{A\sigma^2}\int_{\phi*}^{\infty}\phi f(\phi)d\phi}, \quad (4)$$

where $\alpha = \frac{N}{L}$, $A = \alpha(N-S) + NS(1+\sqrt{\alpha})$, $f(\phi)$ denotes asymptotic spectral density of $\mathbf{X}_h\mathbf{X}_h^T$ eigenvalues, and ϕ^* is such that $\int_{\phi^*}^{\infty} f(\phi)d\phi = \frac{k}{N}$.

Proof: Under Lemma 1, we confirm that $\mathcal{L}(D(\mathbf{Y}_m^{k,k}))$ is composed of the sum of singular values of $\mathbf{X}_h$. To analyze the singular values, we adopt eigenvalue analysis using the Wishart ensemble matrix $\mathbf{W} = \mathbf{X}_h\mathbf{X}_h^T$ [40], [41]. Let $\Phi = [\phi_1, \cdots, \phi_N]$ be the eigenvalues of $\mathbf{W}$. Then the zero-mean data that the singular value Λ of $\mathbf{X}_h$ can be expressed with eigenvalues of $\mathbf{W}$ as $\sqrt{\Phi}$.[2] Therefore, we can express $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ as

$$\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) = \frac{\sqrt{\sum_{j=k+1}^{N}\lambda_j^2}}{\sqrt{\sum_{j=1}^{N}\lambda_j^2}} = \frac{\sqrt{\sum_{j=k+1}^{N}\phi_j}}{\sqrt{\sum_{j=1}^{N}\phi_j}}, \quad (5)$$

where the eigenvalues are assumed to be arranged in descending order of absolute value. By defining the sample mean of each top k eigenvalues as $\overline{\Phi}_k = \frac{1}{k}\sum_{j=1}^{k}\phi_j$, we can rephrase (5) as

$$\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) = \frac{\sqrt{N \times \overline{\Phi}_N - k \times \overline{\Phi}_k}}{\sqrt{N \times \overline{\Phi}_N}}. \quad (6)$$

$\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ can then be represented by the subtraction of the sample mean of the eigenvalues.

To identify the sample means in (6), we consider the sampled eigenvalue spectrum for the statistics of $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$. Let $\hat{\mathbf{W}} = \frac{1}{N}\sum_{j=1}^{N}\mathbf{x}_{h,j}\mathbf{x}_{h,j}^T$ be the sample covariance matrix of source dataset. With a sufficiently large number of samples, the asymptotic spectral density $f(\phi)$ of $\hat{\mathbf{W}}$ can be used to derive the sample mean of eigenvalues [38], which is expressed as in (7), shown at the bottom of the next page.

Here, $\delta(\cdot)$ denotes the Dirac delta function and $\mathcal{H}(t)$ represents the unit step function. The asymptotic spectral density $f(\phi)$ includes parameters $\phi_{\max}$ and $\phi_{\min}$, which are the maximum and minimum eigenvalues determined by the non-effective variance weights, and $\phi_u(\Gamma_{h,j})$, which is determined by the effective variance weights. Then, the eigenvalues are defined as

$$\phi_{\max} = \frac{1}{\alpha}\sigma^2(1+\sqrt{\alpha})^2,$$
$$\phi_{\min} = \frac{1}{\alpha}\sigma^2(1-\sqrt{\alpha})^2,$$
$$\phi_u(\Gamma_{h,j}) = \sigma^2(1+\Gamma_{h,j})(1+\frac{1}{\alpha\Gamma_{h,j}}), \quad (8)$$

[2]The multiplicity of a singular value is the multiplicity of eigenvalues of Wishart ensemble matrix.

where $\phi_u(\Gamma_{h,j}) \geq \phi_{\max} \geq \phi_{\min}$. Then, the sample mean of $\overline{\Phi}_k$ can be expressed as $\overline{\Phi}_k = \frac{N}{k}\int_{\phi^*}^{\infty} \phi f(\phi)d\phi$, where ϕ^* is such that $\int_{\phi^*}^{\infty} f(\phi)d\phi = \frac{k}{N}$. Therefore, we can rephrase $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ in (6) with the sample distribution as

$$\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) = \frac{\sqrt{\int_0^{\infty} \phi f(\phi)d\phi - \int_{\phi^*}^{\infty} \phi f(\phi)d\phi}}{\sqrt{\int_0^{\infty} \phi f(\phi)d\phi}}. \quad (9)$$

Before deriving the upper bound of NRMSE, we note that both the NRMSE and the singular values are non-negative real numbers. Consequently, the decoding accuracy of the proposed algorithm exhibits a decreasing convex function with respect to time k. Hence,

$$\mathcal{L}\Big((1-\beta)D(\mathbf{Y}_m^{k_1,k_1}) + \beta D(\mathbf{Y}_m^{k_2,k_2})\Big) \leq (1-\beta)\mathcal{L}\Big(D(\mathbf{Y}_m^{k_1,k_1})\Big) + \beta\mathcal{L}\Big(D(\mathbf{Y}_m^{k_2,k_2})\Big)$$

and

$$\mathcal{L}\big(D(\mathbf{Y}_m^{k_1,k_1})\big) \geq \mathcal{L}\big(D(\mathbf{Y}_m^{k_2,k_2})\big)$$

for $\beta \in [0, 1]$ and $k_1, k_2 \geq 0$. The maximum of $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ can be achieved if (5) linearly decreases with respect to k. This occurs when eigenvalues are all equal with non-negative real numbers. Eigenvalues are determined by the value of variance weight $\Gamma_{h,j}$ as shown in (8). The maximum value of $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ is then attained when the maximum effective variance weight, $\Gamma_{h,1}$, approaches the minimum value of $\frac{1}{\sqrt{\alpha}}+$. This ensures that all effective variance weights $\Gamma_{h,j}$ for $j \leq S$ are equal. Hence,

$$\begin{aligned}\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) &\leq \lim_{\Gamma_{h,1}\to\frac{1}{\sqrt{\alpha}}+} \mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) \quad (10)\\ &= \lim_{\Gamma_{h,1}\to\frac{1}{\sqrt{\alpha}}+} \frac{\sqrt{\int_0^{\infty} \phi f(\phi)d\phi - \int_{\phi^*}^{\infty} \phi f(\phi)d\phi}}{\sqrt{\int_0^{\infty} \phi f(\phi)d\phi}}.\end{aligned}$$

From (7), the integration of $\phi f(\phi)$ over the range of $[0, \infty]$, which is the denominator of (9), can be reformulated as

$$\begin{aligned}\int_0^{\infty} \phi f(\phi)d\phi &= \int_{\phi_{\min}}^{\phi_{\max}} \phi f(\phi)d\phi + \int_{\phi_{\max}}^{\Gamma_{h,1}} \phi f(\phi)d\phi\\ &= \int_{\phi_{\min}}^{\phi_{\max}} \phi f(\phi)d\phi + \sum_{j=1}^{S} \phi_u(\Gamma_{h,j}). \quad (11)\end{aligned}$$

The integration of the first term can be simplified as

$$\begin{aligned}&\int_{\phi_{\min}}^{\phi_{\max}} \phi f(\phi)d\phi\\ &= \left(1-\frac{S}{N}\right) \times \frac{\alpha}{2\pi\sigma^2} \times \int_{\phi_{\min}}^{\phi_{\max}} \sqrt{(\phi-\phi_{\min})(\phi_{\max}-\phi)}d\phi\\ &= \left(1-\frac{S}{N}\right) \times \frac{\alpha}{2\pi\sigma^2} \times \frac{\pi(\phi_{\max}-\phi_{\min})^2}{8} = \left(1-\frac{S}{N}\right)\sigma^2 \quad (12)\end{aligned}$$

as $\phi_{\max} - \phi_{\min} = \frac{4\sigma^2}{\sqrt{\alpha}}$ in (8). Since S effective variance weights equals to $\frac{1}{\sqrt{\alpha}}$ (i.e., $\Gamma_{h,1} = \cdots = \Gamma_{h,S} = \frac{1}{\sqrt{\alpha}}$) as $\Gamma_{h,1}$ approaches to $\frac{1}{\sqrt{\alpha}}$, the second term of (11) can be simplified as

$$\sum_{j=1}^{S} \phi_u(\Gamma_{h,j}) = \frac{S\sigma^2}{\alpha}(1+\sqrt{\alpha})^2 \quad (13)$$

by substituting $\frac{1}{\sqrt{\alpha}}$ for $\Gamma_{h,j}$ in (8). Therefore, we can reformulate (10) by substituting each terms with (11)–(13) as

$$\begin{aligned}&\lim_{\Gamma_{h,1}\to\frac{1}{\sqrt{\alpha}}+} \frac{\sqrt{\int_0^{\infty} \phi f(\phi)d\phi - \int_{\phi^*}^{\infty} \phi f(\phi)d\phi}}{\sqrt{\int_0^{\infty} \phi f(\phi)d\phi}}\\ &= \frac{\sigma^2}{\sqrt{\alpha N}}\sqrt{1 - \frac{\alpha}{A\sigma^2}\int_{\phi*}^{\infty} \phi f(\phi)d\phi}, \quad (14)\end{aligned}$$

where $A = \alpha(N-S) + NS(1+\sqrt{\alpha})$. □

The further theoretical validation of the upper bound of decoding accuracy is discussed in the Appendix.

V. SIMULATION RESULTS

A. SIMULATION SETUP

This section presents the evaluation of the proposed algorithm against several existing data dissemination approaches both on synthetic and real-world datasets. The network size, the area of the bounded region for mobile edge devices, is set to 324 km^2. The mobile edge devices are randomly distributed with a node density of 40 [devices/km^2], and edge devices that are fixed in position are distributed with a node density of 10 [devices/km^2]. The mobility of mobile edge devices is implemented according to a two-dimensional Gaussian random walk model with a mean of 60 km/h and variance of 1 for the x-axis and y-axis. In this paper, we consider the experimental environment in [42] of 5G broadcasting, which has been standardized in Release 17 of the 3rd Generation Partnership Project (3GPP). The environmental parameters of 5G broadcasting are shown in Table 2. Here, the performance of the data channels of standards is evaluated when the target data rate is 10 Mbps. We conduct simulations with several

$$f(\phi) = (1-\alpha)\mathcal{H}(1-\alpha)\delta(\phi) + \frac{1}{N}\sum_{j=1}^{S}\delta\big(\phi-\phi_u(\Gamma_{h,j})\big)\mathcal{H}(\alpha-\Gamma_{h,j}^{-2}) + \left(1-\frac{1}{N}\sum_{j=1}^{S}\mathcal{H}(\alpha-\Gamma_{h,j}^{-2})\right) \times \frac{\alpha}{2\pi\phi\sigma^2}\sqrt{\max(0,(\phi-\phi_{\min})(\phi_{\max}-\phi))} \quad (7)$$

TABLE 2. Environmental parameters of 5G broadcasting.

Parameters	Values
Effective radiated power	52.8 mW
Center frequency	500 MHz
Bandwidth	8 MHz
Subcarrier spacing	1.25 kHz
Man-made noise	2.3 dB
Antenna gain (at edge device)	9.5 dBi
Noise figure (at edge device)	7.5 dB
Outdoor shadowing standard deviation	5.5 dB
Required carrier-to-noise ratio (CNR)	24 dB

algorithms using synthetic datasets. In the simulations, the data size for broadcasting is set to 300 KByte. The default encoding number is set as $N = 25$, and the default length of encoded data is set as $L = 200$. In the simulations, we conduct $1,000$ trials for each algorithm while varying the encoding number and the data length. We report on the average performance of these $1,000$ trials. We consider two datasets as below.

1) Synthetic dataset: $\mathbf{X}_h$ is generated from a standard multivariate normal distribution (i.e., $\mathbf{X}_h \sim \mathcal{N}(\mathbf{0}_{N,L}, \mathbf{I})$).
2) Real-world dataset: $\mathbf{X}_h$ is a collection of thermal data from South Korea ASOS Dataset in 2022 [43].

We compare the performance of the proposed algorithms with that of the following NC algorithms:

- RLNC: Simple yet powerful encoding scheme that transmits random linear combinations of the packets. It requires high computation complexity because of the Gaussian elimination process in the decoding.
- Sliding NC [26]: Packets are generated using a sliding window of original data packets to avoid long decoding delay.
- Vandermonde echelon fast coding (VEFC) [25]: Combine the packets while distinguishing between the real-time requirements of different packets. It reduces the dependency problem of NC such that increases the decoding accuracy.
- Sparse NC [27]: Encode packets using sparse coding coefficient matrix. This lowers computational complexity by reducing the number of nonzero coding coefficients.
- Fulcrum non-systematic sliding window (FNSW) coding [28]: Consider subsets of the coefficient matrix while partially overlapping the coefficient matrix. This algorithm substantially reduces the in-order packet delay.

For typical broadcasting without applying NC, we also consider state-of-the-art algorithms as follows:

- Data importance aware (DIA) scheduling [44]: Transmit important data without using NC. It takes into account the informativeness of data samples, besides communication reliability.
- Top-k with retransmissions (Top-k) [15]: To avoid packet loss, it systematically integrates model compression and retransmission towards Federated Learning. It reduces communication traffic and ensures model accuracy. In the simulations, we only broadcast the top 80% of the original data.
- Repeated transmission (RT) [14]: To avoid wasting wireless resources, packet-by-packet parallel processing is conducted in the wireless access networks, supporting the RT technique with individual feedback.

In this simulation, we evaluate the performance of the proposed algorithm in terms of decoding accuracy and time delay. We measure the averaged decoding accuracy among the registered mobile edge devices at time k, defined as

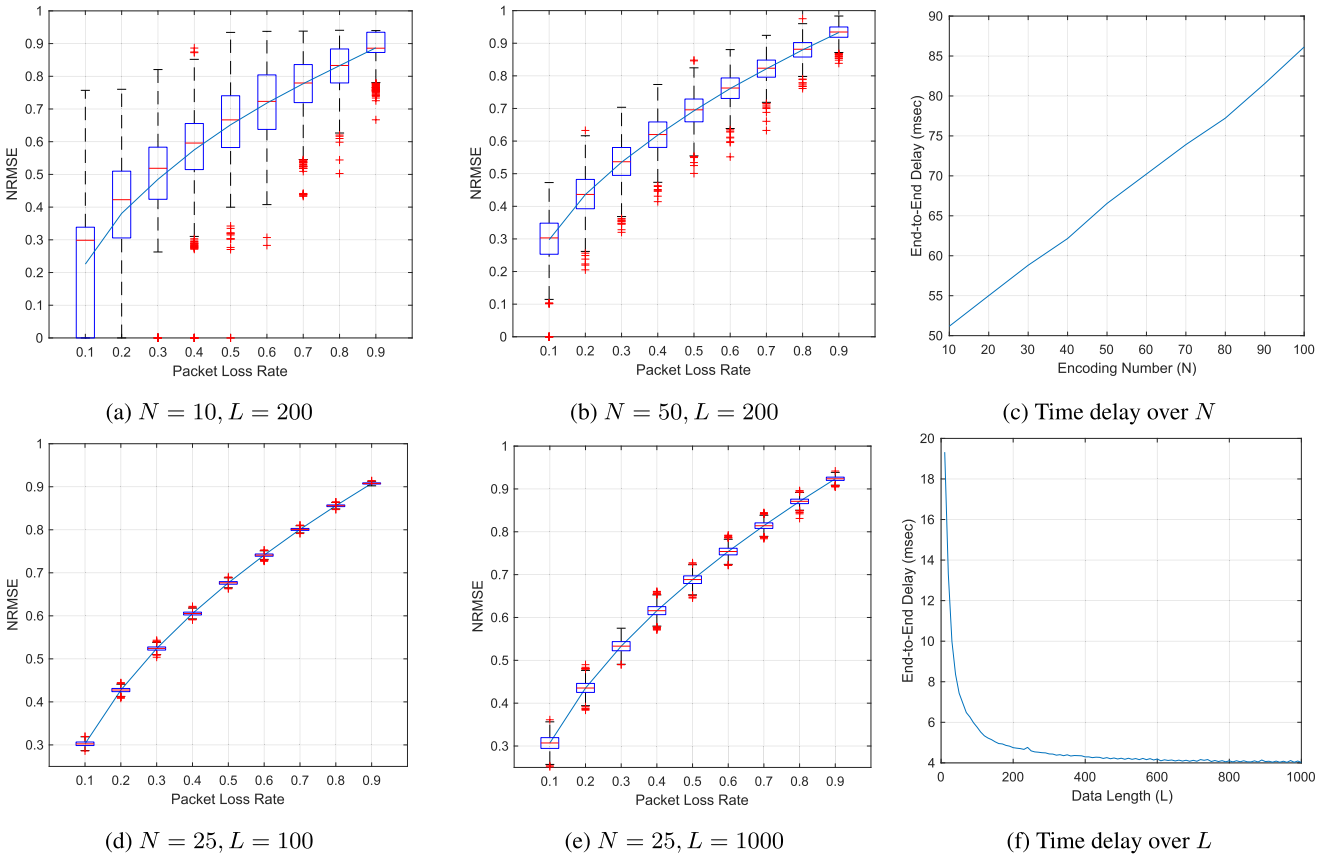
$$\overline{\mathcal{L}}\big(D(\mathbf{Y}_m^{n,k})\big) = \frac{1}{|\mathbf{M}|}\sum_{m\in\mathbf{M}} \mathcal{L}\big(D(\mathbf{Y}_m^{n,k})\big), \forall n,$$

where $|\mathbf{M}|$ denotes the number of devices registered in the edge server. For time delay evaluation, we consider the transmission delay required to disseminate the entire dataset. All the results presented in this section are obtained in MATLAB implementation on the system with Intel Core i7-4770 3.40GHz CPU, 8.00GB RAM, and Windows 10 Pro.

B. PERFORMANCE EVALUATION

To verify the performance of the proposed algorithm, we examined the decoding and delay performance based on packet loss rates. In this section, we evaluated the performance using synthetic data. The first row of Fig. 2 shows the results of the proposed algorithm with varying encoding rates. Fig. 2a and Fig. 2b show the decoding performance if encoding number (N) is relatively small ($N = 10$) and is relatively large ($N = 50$), respectively. While average decoding accuracy shows a similar pattern regardless of the encoding number, a wide distribution of the maximum and minimum decoding error is observed with the smaller encoding number. With larger encoding numbers, the variance of the distribution of maximum and minimum decoding errors becomes smaller. This implies that the decoding performance can be more stable as more packets are encoded together (i.e., a larger encoding number). At the cost of stable decoding error performance, however, more time is required for transmission as shown in Fig. 2c. Fig. 2c shows the time delay where the delay increases if we continuously increase the value of N. This is a clear trade-off between decoding accuracy and time delay.

Alternatively, Fig. 2d-2e show the decoding accuracy for the length of encoded data (L). While decoding accuracy is stable with small L (e.g., $L = 100$), the distribution of NRMSE becomes wide as L increases (e.g., $L = 1,000$). This is because the amount of data lost from the decoding failure may become large with lengthy data encoded, thereby leading to a longer burst decoding error. However, more time is required with larger encoded data length as shown in Fig. 2f, which again implies a trade-off between decoding accuracy and time delay.



(a) $N = 10, L = 200$ (b) $N = 50, L = 200$ (c) Time delay over N

(d) $N = 25, L = 100$ (e) $N = 25, L = 1000$ (f) Time delay over L

FIGURE 2. The average NRMSE and time delay (msec) performances of the proposed algorithm across different packet loss rates. The top row indicates the performance when varying the encoding number (N), while the bottom row reflects performance changes with variations in the length of encoded data (L).

C. THE IMPACT OF DATA CORRELATION ON DECODING ACCURACY

In this section, we verify the impact of data correlation on decoding accuracy using synthetic and real-world datasets. In this simulation, we assume that there is no packet loss during the data transmissions. As shown in Fig. 3, the proposed algorithm outperforms all other methods in all time periods including the worst-case scenario (synthetic data). In the case of RLNC, VEFC, and sparse NC, the NRMSE values remain at their maximum until all the data packets are received. This is because those algorithms can start the decoding process only when the N number of the encoded packets has arrived. The proposed algorithm also outperforms the other real-time NC algorithms (sliding NC and FNSW coding), typical broadcasting (TB), DIA scheduling, and Top-k.[3] Not only real-time decoding but also a larger amount of information transmission at the beginning of the data dissemination stage of the proposed algorithm can lead to outperformance.

It is also observed that the proposed algorithm shows significantly superior performance with real-world data sets than with synthetic data sets. This is because real-world data is much more correlated than synthetic data, which leads to higher variance and singular values.

D. PERFORMANCE COMPARISON WITH PACKET LOSS

In this section, we compare the decoding and time delay performance of the proposed algorithm with state-of-the-art algorithms especially where there is packet loss. We assume that packet loss may occur due to the broadcasting channel state and the mobility of edge devices.

We first compare the decoding accuracy and delay performance of our algorithm with other data dissemination algorithms in a broadcasting environment without a retransmission mechanism. In this scenario, for a fair comparison, we consider real-time decodable algorithms, which are TB, DIA scheduling, FNSW, sliding NC, RT, and DR. In algorithms that consider redundant packet transmissions or retransmissions, the maximum additional transmission ratio is limited to 20% of the original packet transmission to avoid broadcast storm problems.

Fig. 4a shows the average decoding accuracy performance of the stationary edge mobiles that remain fixed in the

[3]RT algorithm behaves the same as TB if there is no packet loss. Thus, the RT algorithm is omitted in this simulation.

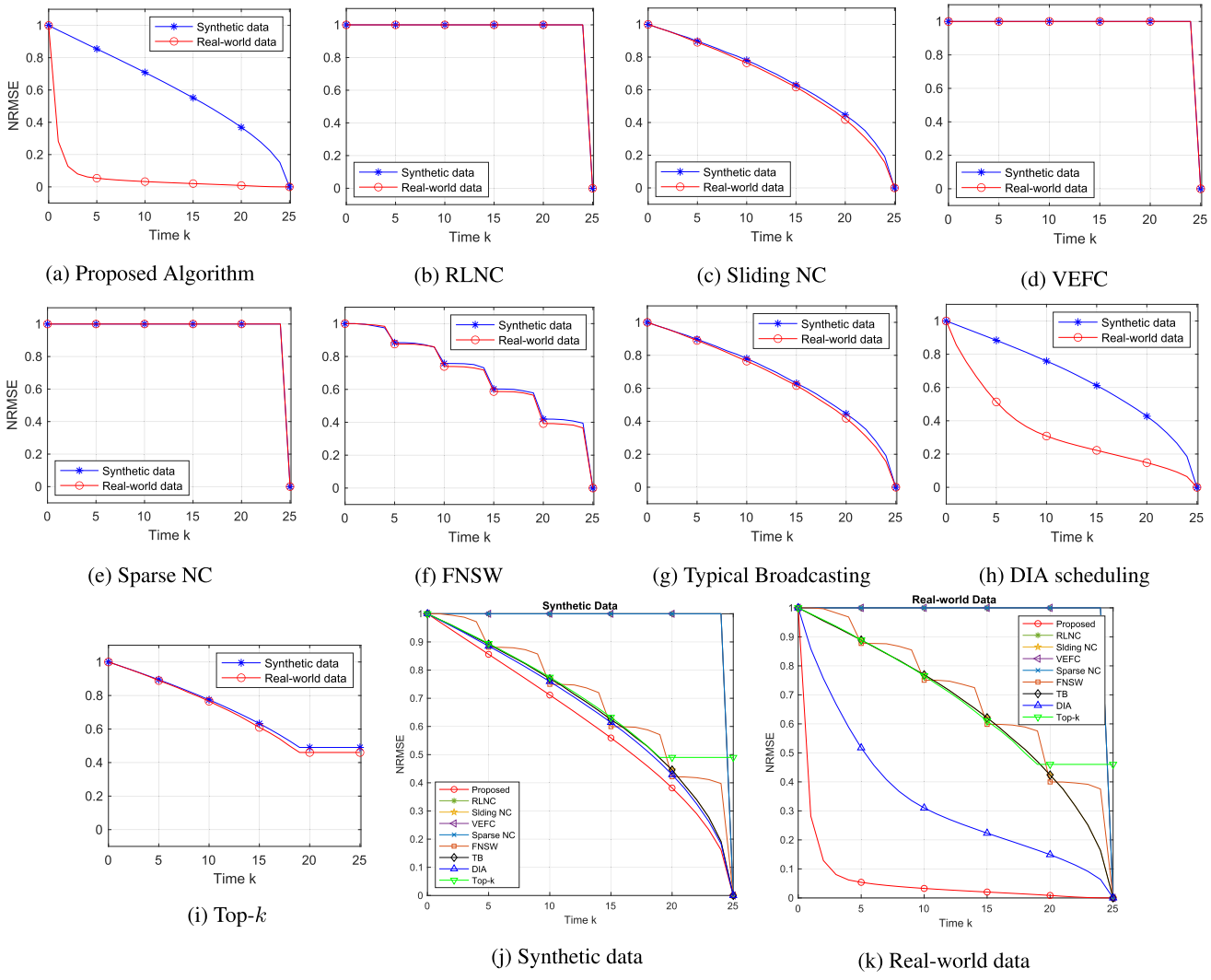


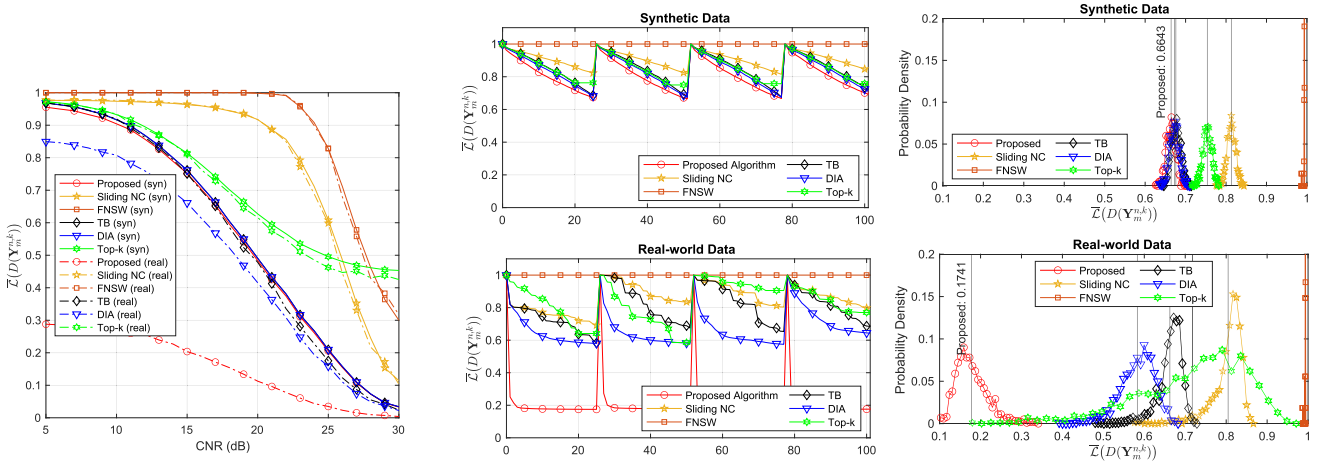
FIGURE 3. The decoding accuracy of each algorithm with no packet loss.

edge server, taking into account. As the positions of the edge devices are stationary, packet loss occurs due to the channel conditions. It can be observed that the proposed algorithm outperforms other algorithms even at a lower carrier-to-noise ratio (CNR) than the required CNR. If the broadcasting protocol considers dissemination with data importance (i.e., the proposed algorithm, DIA scheduling, and Top-k algorithm), the performance tends to improve with higher levels of data correlation.

Fig. 4b and Fig. 4c show the average decoding accuracy performance over the network that includes both mobile edge devices and stationary edge devices. As shown in Fig. 4b, the proposed algorithm shows a superb performance in the early data dissemination stage. This is more highlighted when real-world data is considered. However, the decoding accuracy degrades because subcarrier spacing 1.25kHz against the Doppler effect causing inter-carrier interference (ICI) in mobile environments [42], [45]. In this case, FNSW shows the worst performance because of the ICI and singularity issues in the coefficient matrix.

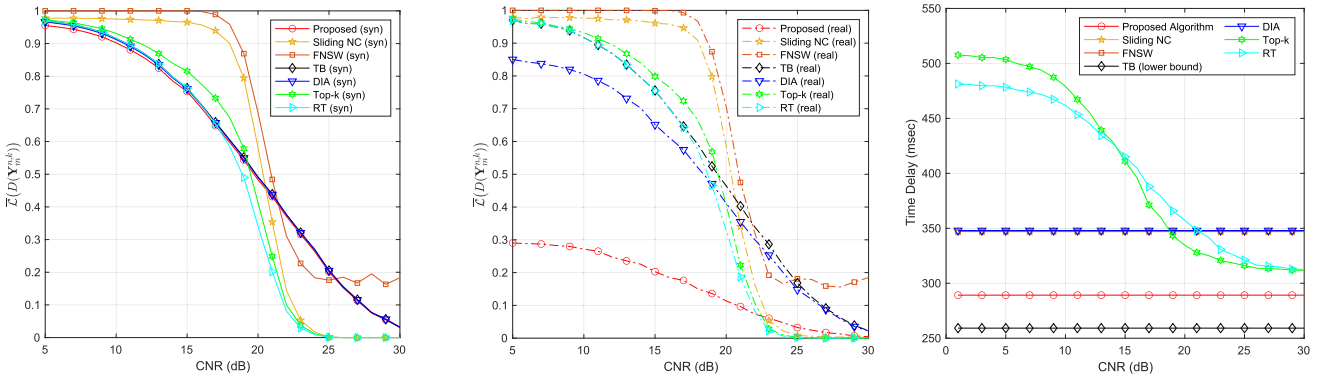
Fig. 4c shows the distribution of $\overline{\mathcal{L}}\big(D(\mathbf{Y}_m^{n,N})\big)$. While the proposed algorithm shows slightly lower NRMSE performance with synthetic data sets, performance is similar to TB and DIA scheduling. However, the proposed algorithm achieves a significantly lower NRMSE (0.1741) with real-world data sets than the FNSW (0.9933), sliding NC (0.8009), TB (0.6575), DIA scheduling (0.5767), and Top-k (0.7171). Hence, it can be concluded that the proposed algorithm can be practically adopted in data dissemination networks and be robust against packet losses.

In a broadcasting protocol that permits additional packet transmissions or retransmissions, the decoding performance can vary accordingly. Fig. 5 illustrates the performance comparison where NC algorithms transmit additional packets, while Top-k and RT utilize retransmissions within the broadcasting. As shown in Fig. 5a and Fig. 5b, when the



(a) NRMSE over CNR (b) NRMSE over time (c) Distribution of NRMSE

FIGURE 4. **The average NRMSE of each algorithm with packet loss under both synthetic data (syn) and real-world data (real) in broadcasting without retransmission and additional packet transmission mechanisms. (a) is simulated in a network with stationary edge devices, and the average NRMSE was measured as a function of the CNR. (b) and (c) is simulated in a network with both stationary and mobile edge devices.**



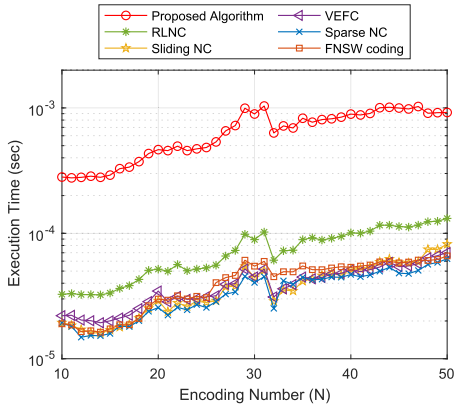
(a) NRMSE over CNR with synthetic data (b) NRMSE over CNR with real-world data (c) Time delay over CNR

FIGURE 5. **The performance comparison in broadcasting that permits additional packet transmissions or retransmissions.**

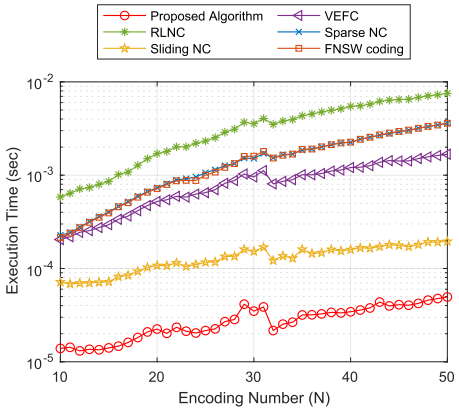
system is allowed to send additional packets, we observe an enhancement in the performance of other NC algorithms as well as the Top-k and RT algorithms. Thus, the proposed algorithm, which consistently outperformed others when not considering retransmissions and additional packets, experiences a shift in NRMSE performance relative to other algorithms when the CNR crosses the 20 dB threshold. However, when considering time delay, it can be observed that the proposed algorithm transmits data very efficiently even at CNR levels above 20 dB. As shown in Fig. 5c, when TB is the lower bound of the transmission delay, the proposed algorithm outperforms other algorithms regardless of CNR. This is because NC algorithms consistently transmit additional packets, resulting in higher transmission delays compared to the proposed algorithm. On the other hand, although Top-k and RT exhibit lower time delays than NC algorithms at CNR values above 20 dB, they experience higher delays and even indirectly encounter broadcast storms at CNR values below 20 dB. Therefore, it can be concluded that the proposed algorithm is the most efficient and reliable for disseminating data in a broadcasting environment compared to state-of-the-art algorithms.

E. COMPLEXITY COMPARISON WITH NETWORK CODING ALGORITHMS

Fig. 6 shows the complexity results of NC algorithms, measured by the execution time for the different numbers of encoding numbers. It can be clearly shown that the complexity of the encoding process in our algorithm is higher compared to other algorithms, while the decoding process is significantly less complex. Specifically, as detailed in Section III-C, Fig. 6b shows that the proposed algorithm reduces the decoding complexity by approximately 90% compared to RLNC. This implies that the proposed algorithm shifts the complexity burden of the decoding process to the encoding process. Hence, most of the computational



(a) Encoding Complexity



(b) Decoding Complexity

FIGURE 6. Execution time over the encoding number.

operations can be performed in the edge server while reducing them in the mobile edge devices, making it an appropriate solution for data dissemination with power-limited edge devices.

VI. CONCLUSION

In this paper, we propose a reliable data dissemination framework in edge networks using the NC technique with low-rank approximation. In the considered edge networks, data packets are broadcasted without retransmission, so that mobile edge devices may fail to receive the entire data packets. We propose a data dissemination framework where the edge server broadcasts encoded data using matrix factorization. Edge devices decode this data based on low-rank approximation, even when only a partial set of data is received. We demonstrate analytically that the proposed algorithm provides higher decoding accuracy than typical broadcasting, and this result is also validated through our experiments. Moreover, we verify that the proposed algorithm shows superior decoding accuracy performance even in environments where packets are lost. Furthermore, the proposed algorithm can reduce computational complexity by shifting the burden of computation from edge devices to the server, thus mitigating the issue of high complexity for the decoding process of NC in mobile edge devices.

TABLE 3. Description of notations.

Notation	Description
$\mathbf{X}_h$	Source data sampled from $\mathcal{N}(\mathbf{0}_{N,L}, \sigma^2(1+\mathbf{\Gamma}_h))$
$\hat{\mathbf{X}}_h$	Source data sampled from $\mathcal{N}(\mathbf{0}_{N,L}, \sigma^2(1+\hat{\mathbf{\Gamma}}_h))$
$\mathbf{\Gamma}_h$	Variance weights where $\mathbf{\Gamma}_{h,1} \to \frac{1}{\sqrt{\alpha}}$
$\hat{\mathbf{\Gamma}}_h$	Variance weights where $\hat{\mathbf{\Gamma}}_{h,S} > \frac{1}{\sqrt{\alpha}}$
$\mathbf{Y}_m^{k,k}$	Set of encoded data packets from $\mathbf{X}_h$
$\hat{\mathbf{Y}}_m^{k,k}$	Set of encoded data packets from $\hat{\mathbf{X}}_h$
$f(\phi)$	Sampled eigenvalue distribution of $\mathbf{X}_h$
$\hat{f}(\phi)$	Sampled eigenvalue distribution of $\hat{\mathbf{X}}_h$

APPENDIX A
VALIDATION OF UPPER BOUND

To verify the limit of decoding accuracy is the upper bound, we demonstrate that the decoding accuracy with $\Gamma_{h,1} \to \frac{1}{\sqrt{\alpha}}$ is always upper than the other decoded data. Let $\mathbf{Y}_m^{k,k}$ be the encoded data from the source data $\mathbf{X}_h$, where $\mathbf{X}_h \sim \mathcal{N}\big(\mathbf{0}_{N,L}, \sigma^2(1+\Gamma_h)\big)$ with S effective variance weights $\Gamma_{h,1}, \cdots, \Gamma_{h,S}$. Here, we assume $\Gamma_{h,1} \to \frac{1}{\sqrt{\alpha}}$. For comparison, we define $\hat{\mathbf{Y}}_m^{k,k}$ as the encoded data from the source data $\hat{\mathbf{X}}_h$, where each element $\hat{\mathbf{X}}_h \sim \mathcal{N}\big(\mathbf{0}_{N,L}, \sigma^2(1+\hat{\Gamma}_h)\big)$ with S effective variance weights $\hat{\Gamma}_{h,1}, \cdots, \hat{\Gamma}_{h,S}$. The sampled eigenvalue distribution of $\mathbf{X}_h$ can be represented by $f(\phi)$, while the sampled eigenvalue distribution of $\hat{\mathbf{X}}_h$ can be represented by $\hat{f}(\phi)$. Then, we show that $\mathbf{Y}_m^{k,k}$ always satisfies $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) \geq \mathcal{L}\big(D(\hat{\mathbf{Y}}_m^{k,k})\big)$. The notations are summarized in Table 3.

Let $F(k) = \int_{\phi^*}^{\infty} \phi f(\phi) d\phi$, $F(N) = \int_{0}^{\infty} \phi f(\phi) d\phi$, $\hat{F}(k) = \int_{\phi^*}^{\infty} \phi \hat{f}(\phi) d\phi$, and $\hat{F}(N) = \int_{0}^{\infty} \phi \hat{f}(\phi) d\phi$. Table 4 summarizes $F(k)$, $F(N)$, $\hat{F}(k)$, and $\hat{F}(N)$. Then the difference between $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ and $\mathcal{L}\big(D(\hat{\mathbf{Y}}_m^{k,k})\big)$ can be written as

$$\begin{aligned}
&\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big) - \mathcal{L}\big(D(\hat{\mathbf{Y}}_m^{k,k})\big) \\
&= \frac{\sqrt{\int_0^{\phi^*} \phi f(\phi) d\phi}}{\sqrt{\int_0^{\infty} \phi f(\phi) d\phi}} - \frac{\sqrt{\int_0^{\phi^*} \phi \hat{f}(\phi) d\phi}}{\sqrt{\int_0^{\infty} \phi \hat{f}(\phi) d\phi}} \\
&= \frac{\sqrt{F(N) - F(k)}}{\sqrt{F(N)}} - \frac{\sqrt{\hat{F}(N) - \hat{F}(k)}}{\sqrt{\hat{F}(N)}} \\
&= \frac{\sqrt{\hat{F}(N)\big(F(N) - F(k)\big)} - \sqrt{F(N)\big(\hat{F}(N) - \hat{F}(k)\big)}}{\sqrt{F(N)\hat{F}(N)}},
\end{aligned} \tag{15}$$

where ϕ^* satisfies $\int_{\phi^*}^{\infty} f(\phi) d\phi = \int_{\phi^*}^{\infty} \hat{f}(\phi) d\phi = \frac{k}{N}$ for given S. If $\hat{F}(N)\big(F(N) - F(k)\big) - F(N)\big(\hat{F}(N) - \hat{F}(k)\big) \geq 0$, then $\sqrt{\hat{F}(N)\big(F(N) - F(k)\big)} - \sqrt{F(N)\big(\hat{F}(N) - \hat{F}(k)\big)} \geq 0$ since all the integrals of sampled eigenvalue distribution are real numbers and greater than or equal to 0.

TABLE 4. The integrals of sampled eigenvalue distribution.

	Range	Integral
$F(k)$	$k \le S$	$\frac{1}{N}\sum_{j=1}^{k}\phi_u(\Gamma_{h,j})$
	$k > S$	$\frac{1}{N}\sum_{j=1}^{S}\phi_u(\Gamma_{h,j}) + \int_{\phi^*}^{\phi_{\max}}\phi f(\phi)d\phi$
$\hat{F}(k)$	$k \le S$	$\frac{1}{N}\sum_{j=1}^{k}\phi_u(\hat{\Gamma}_{h,j})$
	$k > S$	$\frac{1}{N}\sum_{j=1}^{S}\phi_u(\hat{\Gamma}_{h,j}) + \int_{\phi^*}^{\phi_{\max}}\phi \hat{f}(\phi)d\phi$
$F(N)$	$\forall k$	$\frac{1}{N}\sum_{j=1}^{S}\phi_u(\Gamma_{h,j}) + \int_{\phi_{\min}}^{\phi_{\max}}\phi f(\phi)d\phi$
$\hat{F}(N)$	$\forall k$	$\frac{1}{N}\sum_{j=1}^{S}\phi_u(\hat{\Gamma}_{h,j}) + \int_{\phi_{\min}}^{\phi_{\max}}\phi \hat{f}(\phi)d\phi$

In case of $k \le S$, (15) can be expressed as follows by considering (12) and (13):

$$\begin{aligned}
&\lim_{\Gamma_{h,1}\to\frac{1}{\sqrt{\alpha}}+} \hat{F}(N)\big(F(N)-F(k)\big) - F(N)\big(\hat{F}(N)-\hat{F}(k)\big) \\
&= \lim_{\Gamma_{h,1}\to\frac{1}{\sqrt{\alpha}}+} F(N)\hat{F}(k) - \hat{F}(N)F(k) \\
&= \frac{1}{kSN^2}\frac{\sigma^2}{\alpha}(1+\sqrt{\alpha})^2\left(\frac{1}{k}\sum_{j=1}^{k}\phi_u(\hat{\Gamma}_{h,j}) - \frac{1}{S}\sum_{j=1}^{S}\phi_u(\hat{\Gamma}_{h,j})\right) \\
&\quad + \left(1-\frac{S}{N}\right)\sigma^2\left(\sum_{j=1}^{k}\phi_u(\hat{\Gamma}_{h,j}) - k\frac{\sigma^2}{\alpha}(1+\sqrt{\alpha})^2\right) \ge 0,
\end{aligned} \tag{16}$$

where the equality is achieved when $S = N$. The result follows by noting that the $\frac{1}{k}\sum_{j=1}^{k}\phi_u(\hat{\Gamma}_{h,j}) - \frac{1}{S}\sum_{j=1}^{S}\phi_u(\hat{\Gamma}_{h,j}) \ge 0$ for $k \le S$ as $\phi_u(\hat{\Gamma}_{h,j})$ is arranged in descending order. Furthermore, $\sum_{j=1}^{k}\phi_u(\hat{\Gamma}_{h,j}) - k\frac{\sigma^2}{\alpha}(1+\sqrt{\alpha})^2 \ge 0$ as $\phi_u(\hat{\Gamma}_{h,j}) > \frac{\sigma^2}{\alpha}(1+\sqrt{\alpha})^2\ \forall j$. Therefore, we can conclude that $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ is a upper bound of $\mathcal{L}\big(D(\hat{\mathbf{Y}}_m^{k,k})\big)$ for $k \le S$, where $\Gamma_{h,1}$ is close to $\frac{1}{\sqrt{\alpha}}$.

In case of $k > S$, we can also find that $\mathcal{L}\big(D(\mathbf{Y}_m^{k,k})\big)$ is a upper bound of $\mathcal{L}\big(D(\hat{\mathbf{Y}}_m^{k,k})\big)$ by substituting $\hat{F}(k)$ and $F(k)$ in Table 4. Therefore, we can conclude the proof of the validation.

REFERENCES

[1] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
[2] Q. Luo, S. Hu, C. Li, G. Li, and W. Shi, "Resource scheduling in edge computing: A survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 4, pp. 2131–2165, 4th Quart., 2021.
[3] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.
[4] L. Chang, Z. Zhang, P. Li, S. Xi, W. Guo, Y. Shen, Z. Xiong, J. Kang, D. Niyato, X. Qiao, and Y. Wu, "6G-enabled edge AI for metaverse: Challenges, methods, and future research directions," *J. Commun. Inf. Netw.*, vol. 7, no. 2, pp. 107–121, Jun. 2022.
[5] E. Li, L. Zeng, Z. Zhou, and X. Chen, "Edge AI: On-demand accelerating deep neural network inference via edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 447–457, Jan. 2020.
[6] M. Lévesque, F. Aurzada, M. Maier, and G. Joós, "Coexistence analysis of H2H and M2M traffic in FiWi smart grid communications infrastructures based on multi-tier business models," *IEEE Trans. Commun.*, vol. 62, no. 11, pp. 3931–3942, Nov. 2014.
[7] J. Li, W. Liang, M. Huang, and X. Jia, "Reliability-aware network service provisioning in mobile edge-cloud networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 31, no. 7, pp. 1545–1558, Jul. 2020.
[8] M. Huang, W. Liang, X. Shen, Y. Ma, and H. Kan, "Reliability-aware virtualized network function services provisioning in mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 11, pp. 2699–2713, Nov. 2020.
[9] Y. Chen, Z. Liu, Y. Zhang, Y. Wu, X. Chen, and L. Zhao, "Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4925–4934, Jul. 2021.
[10] W. Wang, T. Luo, and Y. Hu, "An adaptive information quantity-based broadcast protocol for safety services in VANET," *Mobile Inf. Syst.*, vol. 2016, pp. 1–12, 2016.
[11] O. Urmonov and H. Kim, "A multi-hop data dissemination algorithm for vehicular communication," *Computers*, vol. 9, no. 2, p. 25, Mar. 2020.
[12] R. Hu, "Efficient probabilistic information broadcast algorithm over random geometric topologies," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–6.
[13] M. A. Raza, M. Abolhasan, J. Lipman, N. Shariati, and W. Ni, "Statistical learning-based dynamic retransmission mechanism for mission critical communication: An edge-computing approach," in *Proc. IEEE 45th Conf. Local Comput. Netw. (LCN)*, Nov. 2020, pp. 393–396.
[14] E. Kim, Y. Lee, and H. Lee, "An applicable repeated transmission for low latency and reliable services," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8468–8482, Aug. 2020.
[15] X. Su, Y. Zhou, L. Cui, and J. Liu, "On model transmission strategies in federated learning with lossy communications," *IEEE Trans. Parallel Distrib. Syst.*, vol. 34, no. 4, pp. 1173–1185, Apr. 2023.
[16] D. Liu, G. Zhu, Q. Zeng, J. Zhang, and K. Huang, "Wireless data acquisition for edge learning: Data-importance aware retransmission," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 406–420, Jan. 2021.
[17] D. E. Lucani, M. Médard, and M. Stojanovic, "Systematic network coding for time-division duplexing," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2010, pp. 2403–2407.
[18] E. M. Ar-Reyouchi, K. Ghoumid, R. Yahiaoui, and O. Elmazria, "Optimized reception sensitivity of WBAN sensors exploiting network coding and modulation techniques in an advanced NB-IoT," *IEEE Access*, vol. 10, pp. 35784–35794, 2022.
[19] R. Torre, I. Leyva-Mayorga, S. Pandi, H. Salah, G. T. Nguyen, and F. H. P. Fitzek, "Implementation of network-coded cooperation for energy efficient content distribution in 5G mobile small cells," *IEEE Access*, vol. 8, pp. 185964–185980, 2020.
[20] M. Kwon and H. Park, "Network coding based evolutionary network formation for dynamic wireless networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1316–1329, Jun. 2019.
[21] M. Kwon, H. Kim, I. Zhang, S. Bai, and H. Park, "Systematic network coding based reliable real-time multimedia streaming system," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2018, pp. 1–2.
[22] M. Kwon, H. Park, and P. Frossard, "Compressed network coding: Overcome all-or-nothing problem in finite fields," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2014, pp. 2851–2856.
[23] R. Su, Q. T. Sun, and Z. Zhang, "Delay-complexity trade-off of random linear network coding in wireless broadcast," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5606–5618, Sep. 2020.
[24] J. Heide, M. V. Pedersen, F. H. P. Fitzek, and T. Larsen, "Cautious view on network coding—From theory to practice," *J. Commun. Netw.*, vol. 10, no. 4, pp. 403–411, Dec. 2008.
[25] C. Han, Y. Yang, and X. Han, "A fast network coding scheme for mobile wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 2, Feb. 2017, Art. no. 155014771769324.
[26] J. Choi, "Sliding network coding for URLLC," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 4424–4433, Jun. 2022.
[27] Y. Li, S. Zhang, J. Wang, X. Ji, H. Wu, and Z. Bao, "A low-complexity coded transmission scheme over finite-buffer relay links," *IEEE Trans. Commun.*, vol. 66, no. 7, pp. 2873–2887, Jul. 2018.
[28] E. Tasdemir, V. Nguyen, G. T. Nguyen, F. H. P. Fitzek, and M. Reisslein, "FSW: Fulcrum sliding window coding for low-latency communication," *IEEE Access*, vol. 10, pp. 54276–54290, 2022.
[29] J. Le, J. C. S. Lui, and D.-M. Chiu, "On the performance bounds of practical wireless network coding," *IEEE Trans. Mobile Comput.*, vol. 9, no. 8, pp. 1134–1146, Aug. 2010.

[30] D. Zhang, M. Piao, T. Zhang, C. Chen, and H. Zhu, "New algorithm of multi-strategy channel allocation for edge computing," *AEU-Int. J. Electron. Commun.*, vol. 126, Nov. 2020, Art. no. 153372.
[31] A. Hisham, D. Yuan, E. G. Ström, and F. Brännström, "Adjacent channel interference aware joint scheduling and power control for V2V broadcast communication," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 443–456, Jan. 2021.
[32] C. Masouros, T. Ratnarajah, M. Sellathurai, C. B. Papadias, and A. K. Shukla, "Known interference in the cellular downlink: A performance limiting factor or a source of green signal power?" *IEEE Commun. Mag.*, vol. 51, no. 10, pp. 162–171, Oct. 2013.
[33] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 497–510, Jun. 2008.
[34] X. Li, S. Wang, and Y. Cai, "Tutorial: Complexity analysis of singular value decomposition and its variants," 2019, *arXiv:1906.12085*.
[35] R. A. Horn, R. A. Horn, and C. R. Johnson, *Topics in Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
[36] D. Gesbert, H. Bolcskei, D. A. Gore, and A. J. Paulraj, "Outdoor MIMO wireless channels: Models and performance prediction," *IEEE Trans. Commun.*, vol. 50, no. 12, pp. 1926–1934, Dec. 2002.
[37] B. T. Swapna, A. Eryilmaz, and N. B. Shroff, "Throughput-delay analysis of random linear network coding for wireless broadcasting," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6328–6341, Oct. 2013.
[38] D. C. Hoyle and M. Rattray, "Principal-component-analysis eigenvalue spectra from data with symmetry-breaking structure," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 2, Feb. 2004, Art. no. 026124.
[39] D. C. Hoyle and M. Rattray, "PCA learning for sparse high-dimensional data," *Europhys. Lett.*, vol. 62, no. 1, p. 117, 2003.
[40] E. Alan, "Eigenvalues and condition numbers of random matrices," *Soc. Ind. Appl. Math. J. Matrix Anal. Appl.*, vol. 9, no. 4, pp. 543–560, 1988.
[41] R. J. Muirhead, *Aspects of Multivariate Statistical Theory*, vol. 197. Hoboken, NJ, USA: Wiley, 2009.
[42] S.-K. Ahn, S. Ahn, J. Kim, H. Kim, S. Kwon, S. Jeon, M. Ek, S. Simha, A. Saha, P. M. Maru, P. Naik, M. Aitken, P. Angueira, Y. Wu, and S.-I. Park, "Evaluation of ATSC 3.0 and 3GPP rel-17 5G broadcasting systems for mobile handheld applications," *IEEE Trans. Broadcast.*, vol. 69, no. 2, pp. 338–356, Jun. 2023.
[43] Korea Meteorological Administration (KMA). *Automated Synoptic Observing System (ASOS) Dataset*. accessed: Dec. 1, 2021. [Online]. Available: https://data.kma.go.kr/data/grnd/selectAsosRltmList.do?pgmNo=36
[44] D. Liu, G. Zhu, J. Zhang, and K. Huang, "Data-importance aware user scheduling for communication-efficient edge machine learning," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 1, pp. 265–278, Mar. 2021.
[45] Y. Li and L. J. Cimini, "Bounds on the interchannel interference of OFDM in time-varying impairments," *IEEE Trans. Commun.*, vol. 49, no. 3, pp. 401–404, Mar. 2001.

JUNGMIN KWON (Graduate Student Member, IEEE) received the B.S. and M.S. degrees from the Department of Electronic and Electrical Engineering, Ewha Womans University, Seoul, South Korea, where she is currently pursuing the Ph.D. degree. Her research interests include the intersection of networks and distributed learning, in particular, in using network coding and federated learning approaches to facilitate data dissemination within dynamic and complex systems. Relevant applications can be wired/wireless networks and autonomous networks/systems with massive clients.

HYUNGGON PARK (Senior Member, IEEE) received the B.S. degree in electronics and electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 2004, and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Los Angeles (UCLA), in 2006 and 2008, respectively. He was a Senior Researcher with the Signal Processing Laboratory (LTS4), Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, from 2009 to 2010, and a Researcher with the Alan Turing Institute, London, U.K., from 2020 to 2021. He is currently a Professor with the Department of Electronic and Electrical Engineering, Ewha Womans University, Seoul, South Korea. His research interests include machine learning-based distributed decision making strategies for multiagent network systems and efficient and robust data streaming strategies using network coding. He was a recipient of the Graduate Study Abroad Scholarship from the Korea Science and Engineering Foundation, from 2004 to 2006, and the Electrical Engineering Department Fellowship, UCLA, in 2008.