

RESEARCH

Open Access



# Network coding for reliable video distribution in multi-hop device-to-device communications

Lei Wang<sup>1\*</sup> , Yu Liu<sup>2</sup>, Jia Xu<sup>2</sup>, Jun Yin<sup>1</sup>, Lijie Xu<sup>1</sup> and Yuwang Yang<sup>3</sup>

\*Correspondence:

leiwang@njupt.edu.cn

<sup>1</sup> Jiangsu Key Laboratory of Big Data Security and Intelligent Processing of NUPT, 9, Wenyuan Road, Nanjing, China

Full list of author information is available at the end of the article

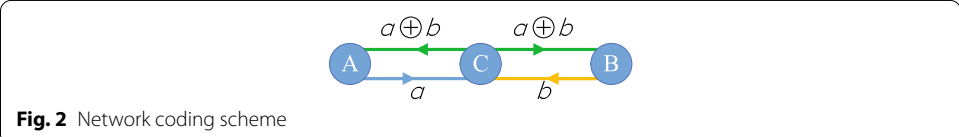
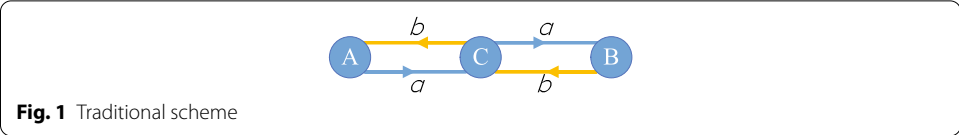
## Abstract

It is becoming more and more popular to share videos among multiple users. However, sharing video in traditional cellular networks will incur high expenses. Device-to-device (D2D) communication is one of the crucial technologies in the fifth-generation network, and it enables the devices to transmit data directly without the relay of base stations. This paper proposes a network-coding-based video distribution scheme for the D2D communication environment. The proposed scheme applies the network coding technology in the H.264 video transmission, which can protect crucial information of the video. This scheme enables the receivers to decode the original video with a high probability, especially in the networks with interferences. Both the simulation results and the actual experimental results show that using network coding technology in video transmission can improve the quality of the received video. Compared with the traditional scheme, the successful decoding rate of the proposed scheme is increased by 46% in our experimental settings.

**Keywords:** Video distribution, Network coding, Device-to-device, Wireless networks

## 1 Introduction

Currently, sharing videos among users is becoming a popular application with the development of the fourth-generation network [1, 2]. More and more people prefer watching videos on their mobile devices. In the traditional cellular networks, even if the sender and the receiver are very close to each other, connections between them are established via the relay of the base station, which will cause transmission delay and increase the burden on the base station. Device-to-device (D2D) communication is a crucial supporting technique in the fifth-generation network [3, 4], and it enables the devices to directly communicate with other nearby devices without the relay of the base station, which could significantly increase the bandwidth efficiency of the entire network. D2D communication has attracted much attention [5, 6], and the standardization organization 3rd-Generation Partnership Project (3GPP) has included D2D communication technology in the development framework of the fifth-generation mobile communication system [7].



However, D2D communication is susceptible to interferences from base stations and other devices working in the same frequency band, which will reduce transmission reliability, especially during the video transmission process [8]. Therefore, it has become an essential goal in the D2D communication environment to provide high-quality and stable video streams to mobile users [9]. As a high-resolution video is usually larger in size, it is necessary to compress the video to improve the efficiency of transmission and storage. H.264 is a typical video compression standard with a high data compression ratio. In H.264 video streams, video frames are of different importance, which means that the loss of crucial information will significantly reduce the video quality, especially in the environment with limited bandwidth and poor signal quality. To minimize the negative effect of interferences caused by the instability of the network environment, a robust and stable video transmission scheme in the D2D communication environment is critical to be designed [10]. In recent years, related researches showed that network coding has great potential in improving transmission reliability [11, 12] and throughput [13, 14] in D2D communication.

Ahlsweede et al. initially proposed network coding [15], and they proved that network coding could increase reliability and bandwidth efficiency. The principle of network coding is to re-encode the data at the intermediate nodes of the network. Through the re-encoding operation at intermediate nodes, the overall network could obtain some additional performance gain [16–18]. Figures 1 and 2 show the operations at intermediate nodes in the traditional store-and-forward scheme and the network-coding-based scheme, respectively. In these figures, node A needs to send a packet  $a$  to both node B and node C. Similarly, node B needs to send a packet  $b$  to both node A and node C. In the traditional communication scheme, the intermediate node C needs to store and forward the packets  $a$  and  $b$  in sequence, so it requires four transmissions in total. In the network coding scheme, node C generates a new packet  $(a \oplus b)$  by performing the bitwise logical XOR operation after receiving packets  $a$  and  $b$ . Then, the new packet  $a \oplus b$  is sent to both A and B in one transmission through the wireless channel. Therefore, three times of transmission are enough. In this example, the network coding scheme requires 25% fewer transmissions compared to the traditional scheme. Network coding includes linear network coding [19] and nonlinear network coding. Linear network coding includes deterministic linear network coding (DLNC) and random linear network coding (RLNC). DLNC requires the knowledge of global topology, which is challenging to implement in a wireless network, so it is

seldom used in practical applications. RLNC is a significant breakthrough in the studies of network coding [20]. In the scheme based on RLNC, the nodes in the network are not required to have the knowledge of global topology at runtime, which implies that RLNC is more suitable for networks in which the topology changes frequently. Therefore, using RLNC is feasible in multi-hop D2D communication networks [21, 22].

In the traditional wired networks, re-encoding operations at intermediate nodes (such as routers) are not supported, which becomes an obstacle to carry out the advantage of the network coding technique fully. In a multi-hop D2D communication environment, mobile devices can be used as intermediate nodes to perform complicated re-encoding operations and provide direct service to other devices. Therefore, the obstacle in traditional networks no longer exists in D2D communication networks. D2D communication provides an ideal application scenario for network coding technology and takes full advantage of network coding [23]. The combination of the two techniques can improve the overall network performance, and the users can obtain more stable video streams when network coding is applied in video services [6, 24].

The rest of this paper is organized as follows: In Sect. 2, some closely related studies are introduced. In Sect. 3, a network-coding-based video distribution scheme is proposed. In Sect. 4, some results of mathematical analysis are provided for both the proposed scheme and existing schemes. In Sect. 5, the performance of the proposed scheme is evaluated in both the simulated network and the actual experimental testbed. Finally, the conclusion is drawn in Sect. 6.

## 2 Related works

At present, researchers have carried out many studies on video transmission in different network environments [25–29]. Nguyen et al. [25] proposed a network coding framework for efficient video streaming transmission in peer-to-peer (P2P) networks. Their framework introduced multiple servers as peers for video transmission. The technology of layered network coding is applied to scalable video streams to deal with bandwidth fluctuations on the Internet. The simulation results showed that network coding technology could save significant bandwidth overhead compared with the traditional schemes. Although D2D communication is essentially a kind of P2P communication, it has distinctive characteristics. Therefore, we cannot directly use the technology designed for P2P networks in our work, and we need to design optimal schemes in multi-hop D2D networks.

To meet consumers' demand for high-quality video in wireless networks in crowded spaces and reduce the transmission overhead, Ferreira et al. [26] proposed a real-time streaming media solution based on wireless multicast. It utilizes partial feedback real-time network coding to generate a repaired package that is maximally useful for all receivers based on feedback messages. The scheme can achieve a balance between the timeliness of the packets and the coding overhead. In their scheme, all the data in the video stream has equal coding importance. While in an H.264 video stream, the frames are of different importance. If this property is taken into account during the design of the encoding scheme, the users will be able to obtain better service.

Rhaim et al. [27] proposed a transmission scheme for the hierarchical transmission of data packets in H.264, which can improve the quality of video streams in mobile ad hoc networks. An Extended Multicast Scalable Video Transmission using Classification Scheduling Algorithms and Network Coding (EMSCNC) over MANET is proposed based on Multicast Scalable Video Transmission (MSVT) [30]. In EMSCNC, the source nodes group the packets and then perform the encoding operation. The intermediate nodes decode the encoded packets generated by the source nodes and then re-encode them according to the hierarchical design of H.264 before forwarding. In this paper, the scheme is proposed for D2D communications to reduce the negative impact caused by interferences, so we addressed the network-coding-based video transmission from a probabilistic analysis perspective. Moreover, we evaluated the feasibility and performance of the proposed scheme in an actual experimental testbed.

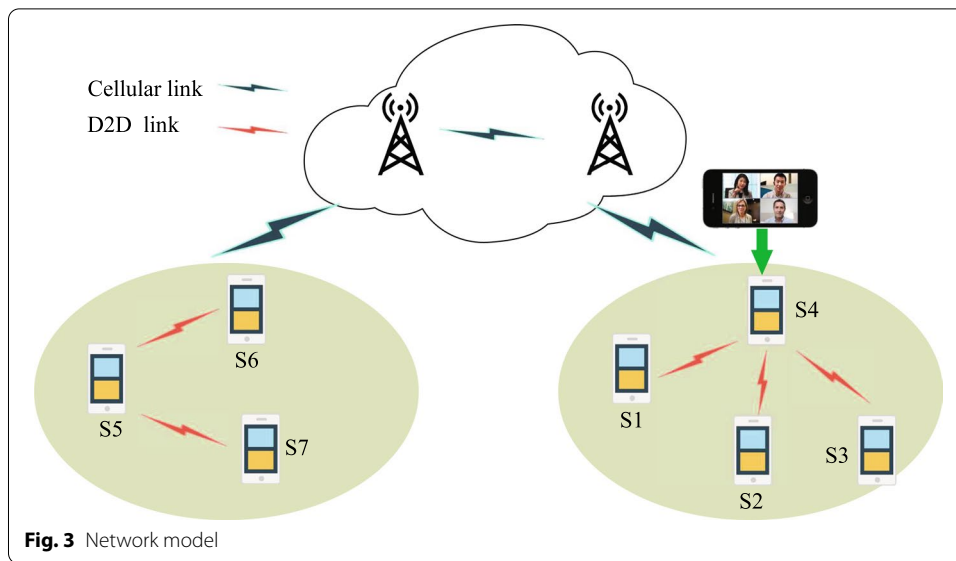
Wang et al. [28, 29] studied the application of network coding in wireless sensor networks and WiFi networks. In the literature [28], the authors applied network coding to improve transmission reliability in wireless sensor networks. Although this paper is about video transmission in the D2D communication environment, it is still applicable to enhance the reliability of the transmission process by using network coding. The authors designed and implemented a reliable network-coding-based video conferencing system (NCVCS) [29] to improve the user experience. In NCVCS, an encoding server is introduced as the intermediate node to perform the re-encoding operation, which can improve the utilization rate of network bandwidth. Moreover, NCVCS adopts a unified coding scheme during data frame transmission without providing additional protection to keyframes. In this paper, mobile devices are used as intermediate nodes, and we provide a network coding scheme based on the priority of different frames. Moreover, NCVCS cannot work without the access point, while all the devices in the testbed in this research could directly communicate with each other without the relay of the access point. Although both NCVCS and the proposed scheme in this paper focus on the implementation of network-coding-based transmission, the network architectures of these two schemes are different.

Our contributions can be summarized as follows: We proposed a network-coding-based video distribution scheme in the D2D communication environment, which can protect critical information; we established a probability-based mathematical analysis model for transmission in the D2D environment; we implemented the proposed scheme and evaluated the performance with actual experiments.

### 3 Methods

#### 3.1 Network model

In the traditional unicast environment, when users working in the same cell need to obtain video information from other devices, they need to establish multiple connections with all other devices. When the problem occurs in the D2D environment that introduces intermediate nodes, the users need to establish a D2D connection with a particular device. As shown in Fig. 3, the mobile devices  $S_1$ ,  $S_2$ , and  $S_3$  act as the source nodes, and the mobile device  $S_4$  acts as an intermediate node. In our network model,  $S_1$ ,  $S_2$ , and  $S_3$  send video streams to  $S_4$ , respectively. After receiving the video from other devices,  $S_4$  performs the re-encoding operation. The re-encoding operations protect the



critical information in the video streams, and then, the generated re-encoded data are sent back to the network. Each device can decode the video stream as long as sufficient encoded data are received.

### 3.2 Coding scheme

In the conventional data transmission, the intermediate node only forwards the packets after receiving them from the source node. The network coding technique allows intermediate nodes to perform additional coding operations on the received packets before forwarding. Encoding, re-encoding, and decoding operations of network coding are linear operations performed over a Galois field (GF) with a size of  $2^q$  where  $q$  is a natural number. According to previous studies [31, 32], the Galois field  $GF(256)$  can provide a good balance between computational efficiency and resource overhead. Therefore, all the coding operations in this paper are based on  $GF(256)$ .

Equation (1) shows a typical linear coding operation [28, 29]. The coding operation is conducted to obtain a linearly independent combination of original data blocks, as shown in Eq. (1).

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1k} \\ c_{21} & c_{22} & \cdots & c_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nk} \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{pmatrix} \tag{1}$$

In Eq. (1),  $k$  stands for the total number of original blocks,  $n$  stands for the number of encoded blocks,  $c_{ij}$  refers to a coefficient randomly selected from  $GF(256)$ , and  $b_i$  refers to the  $i^{th}$  original block. The symbol  $y_i (i \in [1, n])$  stands for an encoded block.

In most RLNC-based schemes [33, 34], the Gauss–Jordan elimination method [35] with excellent decoding efficiency is adopted. The Gauss–Jordan elimination method is a classical decoding method for network coding schemes. Compare to other decoding methods such as matrix inversion method, the Gauss–Jordan elimination method has

higher decoding efficiency. In this sense, it could reduce the decoding delay. This method has the following advantages: When the packet is linearly dependent of the existing packets, it does not contain meaningful information. In this case, an all-zero row will appear after matrix transformation during the decoding operation, and then, the linearly dependent packet will be removed. Besides, the receiving node can start decoding after receiving part of the data packet, instead of waiting for  $k$  linearly independent encoded packets to arrive, which improves the decoding efficiency and reduces the waiting time of the receiving devices.

### 3.3 Reliability analysis

In the video codec system H.264/AVC(Advanced Video Coding), video frames have different types, such as I-frame, P-frame, and B-frame. The I-frame is intra-coded, which has crucial information about other frames [36]. In a video sequence, the first data frame is always an I-frame, followed by a series of P-frames or B-frames [37]. P-frame and B-frame are generated based on temporal and spatial correlations. However, this structure also makes the video sequence more susceptible to error propagation caused by inter-frame dependency, and corruption of the previous frames may result in incorrect propagation of consecutive frames in the Group of Pictures (GOP) [38]. If there is an error in the I-frame, only a small amount of related video information can be recovered in the GOP, while a broad range of frames is lost. Moreover, since the data length of the I-frame is significantly larger than that of other frames, I-frame is more likely to be lost during transmission. Therefore, it is necessary to protect the I-frame during the transmission of the video stream.

Assume that there are multiple GOPs in a video, and the  $x^{th}$  GOP is represented by  $G(x)$ ; an I-frame belonging to  $G(x)$  is represented by  $I(x)$ ;  $P(x)$  represents P-frame of the  $x^{th}$  GOP;  $G_s(x)$  refers to the successful delivery rate (SDR) of all frames in  $x^{th}$  group;  $I_s(x)$  and  $P_s(x)$  are the SDRs of I-frame and P-frame, respectively.

Some studies had analyzed the decoding rate of RLNC based on SDR [39, 40]. Therefore, we use the SDR as an indicator to analyze the performance of the proposed scheme. In the transmission process,  $I_l$  refers to the size of the I-frame,  $P_l$  is the size of the P-frame, and the bit error ratio (BER) is  $p_b$ . The successful transmission probabilities of I-frame and P-frame are  $I_t$  and  $P_t$ , respectively. Although the transmission process of the I-frame and P-frame is independent of each other, their decoding processes are correlated. The SDR of I-frame at the decoding device equals the probability of successful transmission of the I-frame, so  $I_s$  equals  $I_t$ . The decoding operation of the P-frame requires the information of the I-frame, so the SDR of P-frame at the decoding device can be expressed by Eq. (2).

$$P_s = I_s \times P_t = (1 - p_b)^{I_l} \times (1 - p_b)^{P_l} = (1 - p_b)^{\sum_{i=1}^n I_i(x) + P_i(x)} \quad (2)$$

In the network-coding-based scheme,  $I'_l$  stands for the size of the I-frame after the re-encoding operation, and  $P'_l$  is the size of the P-frame after the re-encoding operation. The successful transmission probabilities of encoded I-frame and encoded P-frame are  $I'_t$  and  $P'_t$ , respectively. The encoded I-frame is divided into  $n$  packets, the size of each packet is  $\frac{I'_l}{n}$ , and the successful receiving rate of each packet is  $p$ .

$$p = (1 - p_b)^{\frac{I'_1}{n}} \quad (3)$$

The intermediate node will perform RLNC with  $n_1$  different data packets before transmission. The receiving devices need to receive  $k_1$  linearly independent data packets ( $k_1 < n_1$ ) to decode and obtain the original data packet. The probability of successfully decoding the I-frame at the receiving devices after encoding is calculated as follows.

$$\begin{aligned} I'_s &= \sum_{k_1}^{n_1} \binom{n_1}{k_1} p^{k_1} (1 - p)^{n_1 - k_1} \\ &= \sum_{k_1}^{n_1} \binom{n_1}{k_1} (1 - p_b)^{\frac{k_1 I'_1}{n_1}} \left( 1 - (1 - p_b)^{\frac{k_1 I'_1}{n_1}} \right)^{n_1 - k_1} \end{aligned} \quad (4)$$

When the packet coding is performed on a P-frame, the method based on RLNC is also adopted. P-frame is divided into  $n_2$  packets, the size of each packet is  $\frac{P'_1}{n}$ , and the rate of successful receiving each packet is  $p'$ . The initial data can be successfully decoded as long as  $k_2$  packets are received. After encoding, the successful receiving rate  $P'_t$  of P-frame transmission is shown in Eq. (5).

$$\begin{aligned} P'_t &= \sum_{k_2}^{n_2} \binom{n_2}{k_2} p'^{k_2} (1 - p')^{n_2 - k_2} \\ &= \sum_{k_2}^{n_2} \binom{n_2}{k_2} (1 - p_b)^{\frac{k_2 P'_1}{n_2}} \left( 1 - (1 - p_b)^{\frac{k_2 P'_1}{n_2}} \right)^{n_2 - k_2} \end{aligned} \quad (5)$$

The successful decoding rate  $P'_s$  of the encoded P-frame is obtained with Eq. (6).

$$P'_s = I'_s \times P'_t \quad (6)$$

The constraints are set as follows to provide priority protection to I-frame during the encoding process:

$$n_2 - k_2 \leq n_1 - k_1, n_2 \leq n_1 \quad (7)$$

The probability of successfully decoding the GOP at the decoding node after the network coding operation is finally obtained, which is shown in equation.

$$G'_s = I'_s \times P'_s \quad (8)$$

### 3.4 Algorithms in the proposed scheme

In the process of network transmission, to deal with video frame loss and transmission errors, video frames need to be encoded redundantly. Part of the original data can be recovered by processing the redundant data. However, data redundancy is limited. Repeated transmissions will reduce the utilization efficiency of the transmission bandwidth. Therefore, how to obtain optimal transmission with limited redundancy is a problem that we need to address.

Since the length of each frame in the video stream is different, it is necessary to design an appropriate transmission method to maximize the utilization of transmission resources. We designed a dynamic adaptive algorithm to transmit video and improve data transmission efficiency.

---

**Algorithm 1** Dynamic adaptive transmission algorithm
 

---

```

1: The linked device sends a message to intermediate device  $R$ ;
2: Device  $R$  obtains information about participating transmission devices;
3: Video transmission process begins;
4: while Device  $R$  receives the video stream from all other devices do
5:   Extract the first byte of the header of the video frame;
6:   Set the slice size;
7:   Split the new frame into  $k$  slices according to the length of the new frame;
8:   Set redundancy  $r$ ;
9:   Generating an  $(r + k)$ -by- $k$  coefficient matrix  $M$ ;
10:  Multiply the sliced frame by the matrix  $M$ ;
11:  Generate a packet including  $k$ , coding coefficient, and the encoded frame slice, and then send it out.
12: end while

```

---

In this algorithm, we have chosen an appropriate value as the packet size for one transmission. Therefore, the number of slices changes dynamically with the change of the size of the video frames. We add the slice information and encoding coefficients to the header of each encoded frame, which will be used to assist the decoding operation at receiving devices. RLNC is essentially a linear coding scheme. Therefore, the encoding and decoding operations are to obtain the product of two matrices. The time complexity of multiplying two matrices is  $O(n^3)$ , which is a polynomial-time complexity and is feasible to implement in practical applications.

---

**Algorithm 2** Decoding waiting algorithm
 

---

```

1: while a packet is received do
2:   count++;
3:   if  $count > k$  then
4:     Skip and waiting for the packets of next frame;
5:   else
6:     Extract the first byte and obtain the generation size  $k$ ;
7:     Extract the coding coefficients and the encoded data;
8:     Update the received coding coefficient matrix;
9:     Calculate the rank of the coefficient matrix;
10:    if rank is less than  $k$  then
11:      Recover part of the data according to the current coefficient matrix;
12:    else
13:      Decode all the original data;
14:    end if
15:  end if
16: end while

```

---

The advantage of network coding comes from the re-encoding operation, which will increase independence among data packets during transmission. However, the re-encoding operation will affect the recovery of video frames at the receiving device. In the conventional replication-based transmission scheme, after the device receives the frame information, it can immediately recover the original video frame.



In the proposed scheme, the packets generated by RLNC are linearly mixed. In the theory of linear network coding, the original data are encoded into  $n$  ( $n \geq k$ ) linearly independent packets in which any  $k$  out of  $n$  encoded packets are sufficient to recover the original data. Therefore, only when at least  $k$  linearly independent packets are successfully received, can the receivers decode the original data. Therefore, waiting time is necessary for the decoding operation. In all the linearly coding schemes, the waiting time cannot be avoided. In our scheme, during the transmission of a frame of video, the device can fully recover the original data through the decoding operation after receiving  $k$  linearly independent packets and does not have to wait for the subsequent data packets of the current video frame. Besides, to ensure the real-time performance of the video, it is not necessary to keep waiting for the old video frame when the new frame arrives.

When a mobile device works as an intermediate node, it overtakes the re-encoding operation, which causes extra energy overhead of batteries. Therefore, the intermediate node will run out of energy first, and then, it will be eliminated from the network. To increase the network lifetime for all the mobile devices, we need to design a strategy to periodically select the devices as intermediate nodes so that the energy overhead could be balanced. The strategy needs to take account of at least two factors, namely location and remaining power. Moreover, the strategy needs to assign weights for these factors.

---

**Algorithm 3** Strategy for intermediate node selection
 

---

```

1: Each node  $D_i$  broadcasts its own status with packet  $PS_i$  every ten minutes.
2: while Node  $D_j$  receives a packet  $PS_i$  from node  $D_i$ . do
3:   Node  $D_j$  calculates the priority value  $PV_{j,i}$  for node  $D_i$ .
4:   Node  $D_j$  stores the value  $PV_{j,i}$  in its buffer.
5:   if Node  $D_j$  receives all the packet  $PS_i$  from other nodes. then
6:     Node  $D_j$  votes for the node with greatest priority value.
7:     Node  $D_j$  sends its decision with packet  $P_d$  to current intermediate node  $D_c$ .
8:   end if
9: end while
10: while Node  $D_c$  receives  $P_d$ . do
11:   Node  $D_c$  stores packets  $P_d$  in its buffer.
12:   if Node  $D_c$  receives all the packet  $P_d$  from other nodes. then
13:     Node  $D_c$  selects the node with greatest number of supporters as the intermediate node in the next period.
14:     Node  $D_c$  broadcasts the final decision to all the nodes in the networks.
15:   end if
16: end while

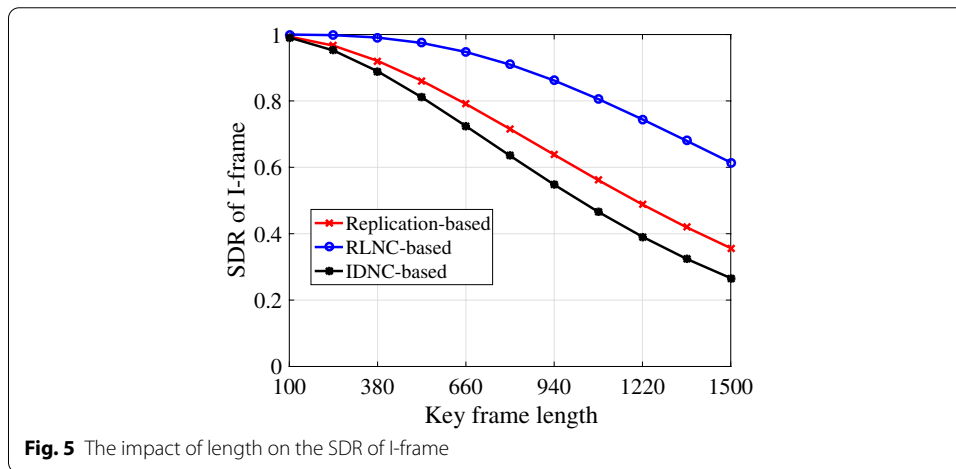
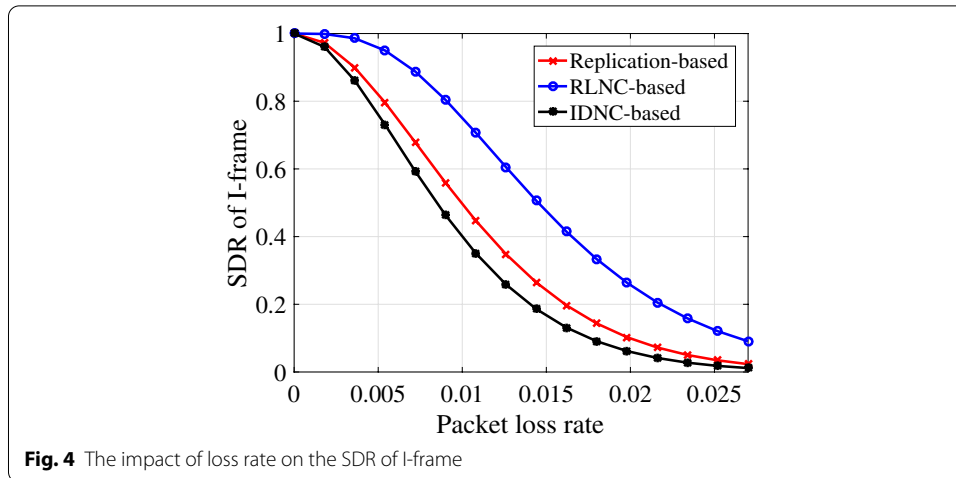
```

---

In this algorithm, each node  $D_i$  broadcasts its status packet  $PS_i$  every ten minutes, which contains its remaining power and location coordinates. When any other node  $D_j$  receives  $PS_i$ , it extracts the location and remaining power of node  $D_i$  and then calculates the distance from node  $D_i$  to itself. The priority value (PV) is obtained with equation.

$$PV = w_d \times (100 - DM \times 20) + w_b \times E \quad (9)$$

In our scheme, the distance and remaining power are assigned different weights, namely  $w_d$  and  $w_b$ . The distance metric  $DM$  is divided into five levels, and the remaining power is quantified as  $E$ .

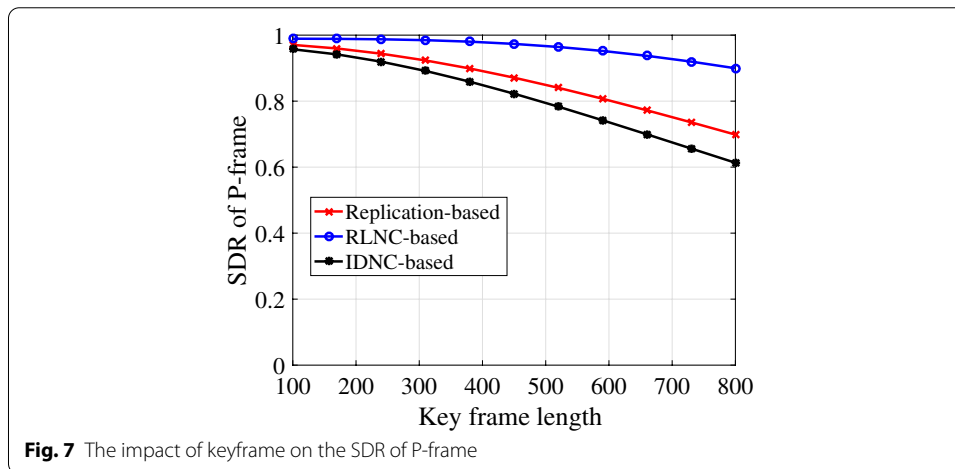
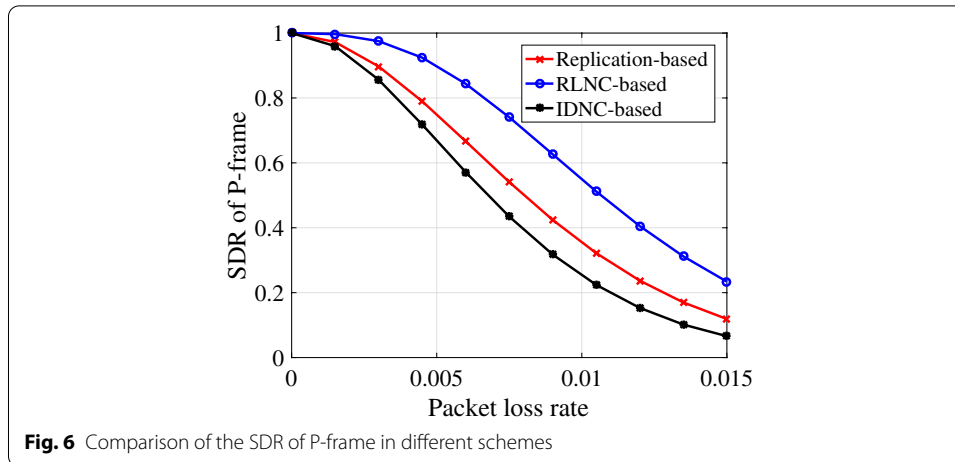


#### 4 Performance analysis

In this section, we verify the reliability of the proposed solution through mathematical computation. The influence of video frame length and packet loss rate on the SDR of I frame, P frame, and GOP in the scheme is analyzed, and we compared the results of the proposed scheme with the replication-based transmission scheme [29] and the transmission scheme based on instantly decodable network coding (IDNC) [41, 42].

From Fig. 4, the decoding rates of all schemes are high when the packet loss rate is low. The SDR gradually decreases as the packet loss rate increases, which is consistent with the analysis of Eq. (4). Compared with IDNC-based and replication-based solutions, the RLNC-based scheme's descent rate is more stable. Moreover, in an extreme network environment with a high packet loss rate, the RLNC-based transmission scheme is more stable.

The impact of frame length on the decoding rate of the I-frame is shown in Fig. 5. As the length of keyframes increases, the SDR of I-frames decreases gradually, which is the reason why I-frames are more likely to be lost. According to Eq. (4), when the video frame to be transmitted is large, the RLNC-based scheme can provide a higher frame recovery rate than other schemes.



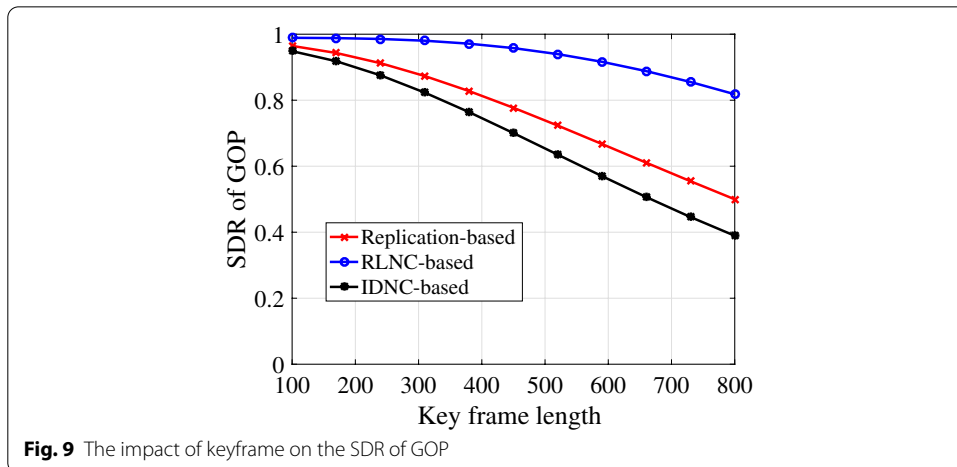
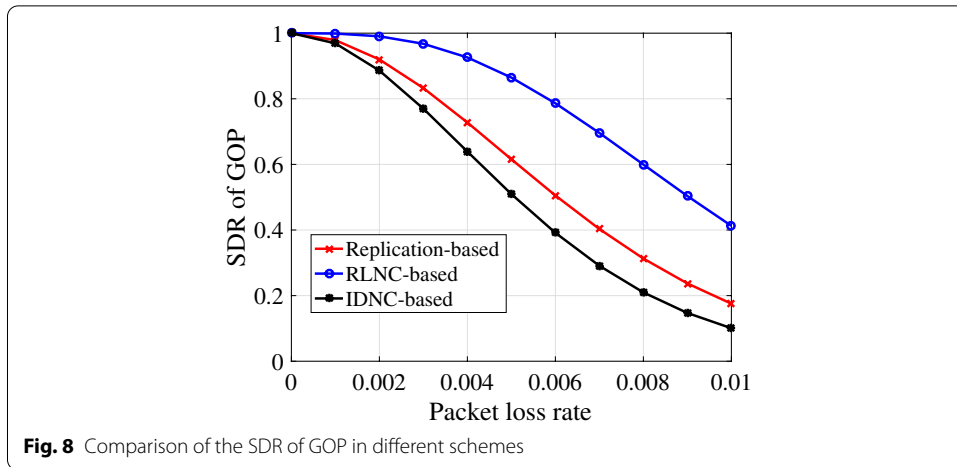
The impact of the packet loss rate on the SDR of P-frame is shown in Fig. 6. In the figure, the packet loss rate is inversely proportional to SDR. According to our previous analysis of Eqs. (5) and (6), the information contained in keyframes has a significant impact on the decoding of P-frame. Figure 7 shows the impact of keyframe length on the SDR of P-frame. As the length of video frames increases, it is more likely to be lost in the transmission process, and the decoding rate of P frames will be affected accordingly. Figures 8 and 9 show the impact of packet loss rate and keyframe length on the SDR of the whole GOP, respectively, according to Eq. (8).

## 5 Performance evaluation

In this section, we implement the proposed scheme in the simulation environment. In the simulation, there are 20 nodes. Moreover, to evaluate the feasibility of the proposed scheme, we implemented it in a real-world testbed consisting of 6 mobile devices.

### 5.1 Setting of simulation

The simulation tool we used is OMNeT++5.4. In our simulation scenario, nodes were randomly deployed in a 1000 m  $\times$  800 m area. The transmission radius of nodes



was 800 m. The loss rates of different links were independent of each other. The packet loss rate can be calculated by the BER. We assume that the BER is  $p'_b$ , and the loss rate of a packet  $p_k$  with transmission data size  $l_s$  is

$$p_k = 1 - (1 - p'_b)^{l_s}. \tag{10}$$

Each simulation experiment lasted for 120 seconds. Unless otherwise stated, we used the parameters shown in Table 1.

In the simulation, we used the default physical layer and transport layer protocols provided by OMNeT++, and we modified the application layer protocol. We designed a new application module to realize the encoding and decoding functions of RLNC. First, the device sends its data to the device used as an intermediate node. After all data from the participating devices are received, the intermediate device encodes the received data and then broadcasts it back to other devices. Figure 10 shows the experimental scene.

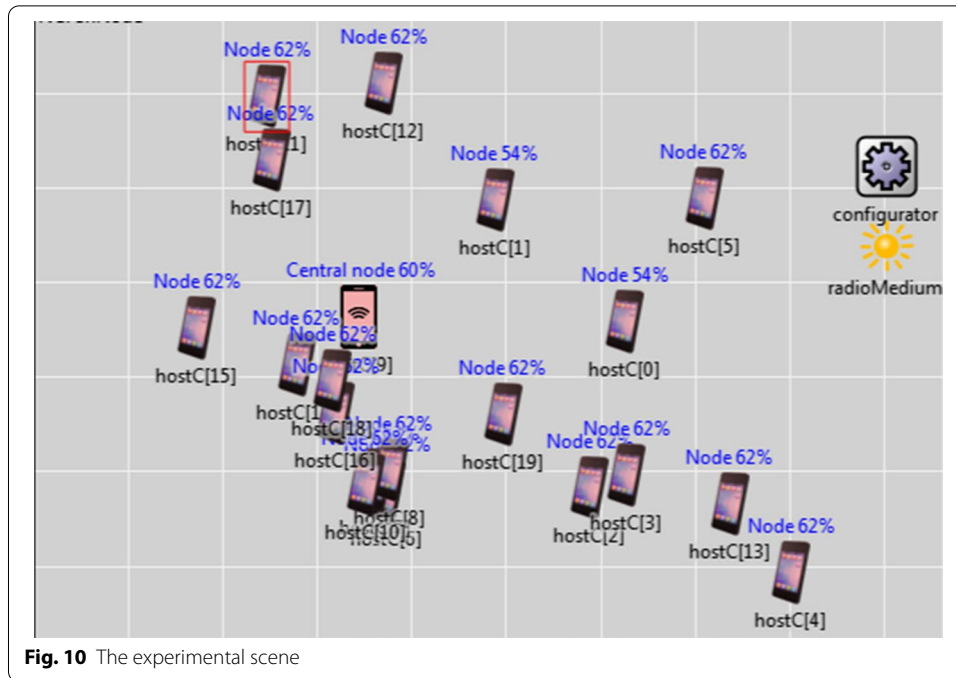


Fig. 10 The experimental scene

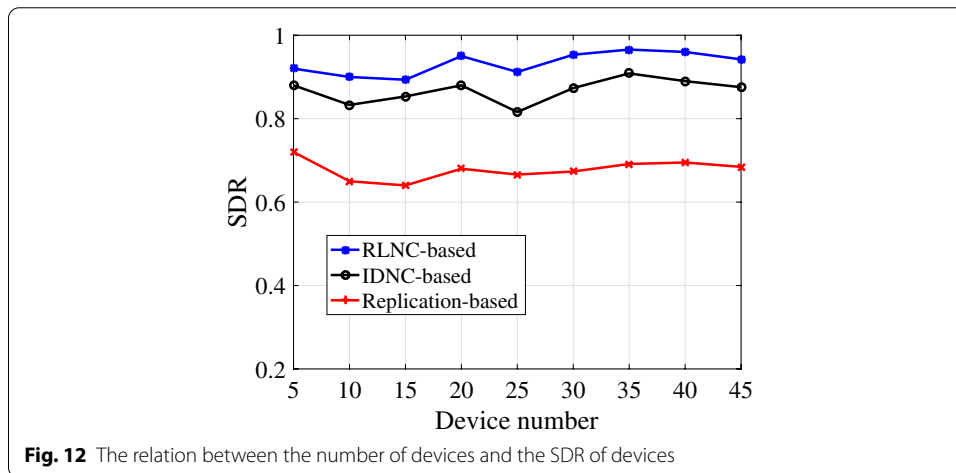
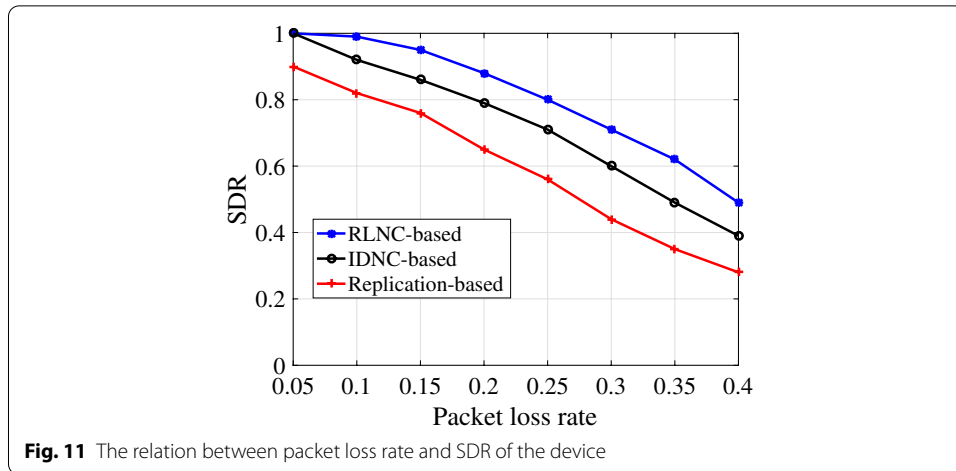
Table 1 Parameters

Item	Value
Area	1000 m × 800 m
Transport layer protocol	UDP
MAC layer protocol	IEEE 802.11g
Transmission control protocol	CSMA
Device type	Wireless Host
Device number	20
Sending interval	exponential(5s)
BER	$2 \times 10^{-5}$
Transmission radius	800m
Transmission rate	10Mbps

### 5.2 Illustrative results

In the simulation environment, we simulated the data transmission process and analyzed the impact of BER, the number of devices participating in the transmission, and the data redundancy on the SDR. We also compared the performance of our scheme with that of the traditional replication-based scheme and IDNC-based scheme. The experimental results were obtained by averaging ten experimental results in order to reduce the influence of unforeseen factors.

Figure 11 shows the relation between BER and SDR. In this experiment, the generation size  $k$  was set at 4, and  $n$  was set at 6. According to Fig. 11, as BER gradually increases, SDR of all three schemes decreases. However, the SDR of the RLNC-based scheme is still higher than that of the other two schemes, because some packets are allowed to be lost after the re-encoding operation. RLNC is not sensitive to changes in



the packet loss rate, and it has higher stability than the other two schemes. Therefore, it is more suitable for transmission in the networks with extreme channel conditions.

The relation between the number of devices involved in transmission and SDR is shown in Fig. 12. In this simulation,  $k$  was set at 4,  $n$  was set at 6, and  $p$  was set at 0.1. According to Fig. 12, in the RLNC-based scheme, after the transmission, the probability of successfully recovering the original data at the receiving devices is the highest, while that of the traditional replication scheme is the lowest. When there are few devices, the results fluctuate greatly. However, as the number of devices increases, the trend of three curves is gradually stabilized. The decoding rate of the RLNC-based scheme is above 0.95, the successful decoding rate of IDNC is above 0.85, and the traditional transmission scheme only has a decoding rate of around 0.65. The experiment results are consistent with the previous theoretical analysis. Therefore, the performance of the scheme based on RLNC is significantly better than that of the other two schemes.

We also analyzed the impact of the relation between  $n$  and  $k$  on the performance of RLNC. The experiment was carried out in a network with 41 devices, and the packet loss rate was set at 0.15. The experiments were carried out with the simulation time of 25s, 50s, 75s, 100s, and 125s, respectively. Then, the experiment results were obtained

through averaging the experiment results in different simulation times. The redundancy rate was set at 50%. From Fig. 13, as  $k$  increases, only the RLNC-based coding scheme maintains a high decoding rate, while the decoding rate of the other two schemes gradually declines, which is because RLNC increases the independence among packets, and it can successfully decode as long as an adequate number of packets are received. In the other two schemes, the probability of receiving duplicated packets or linearly dependent packets is higher than that in the RLNC scheme.

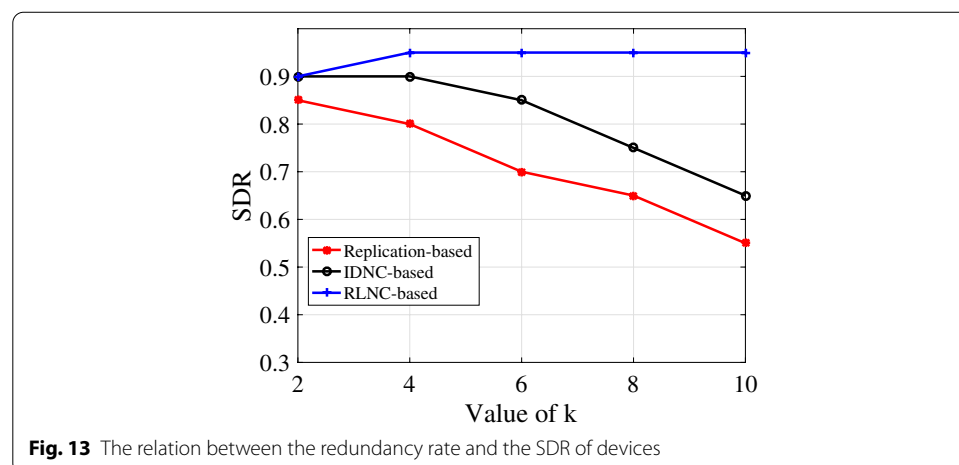
### 5.3 Testbed settings

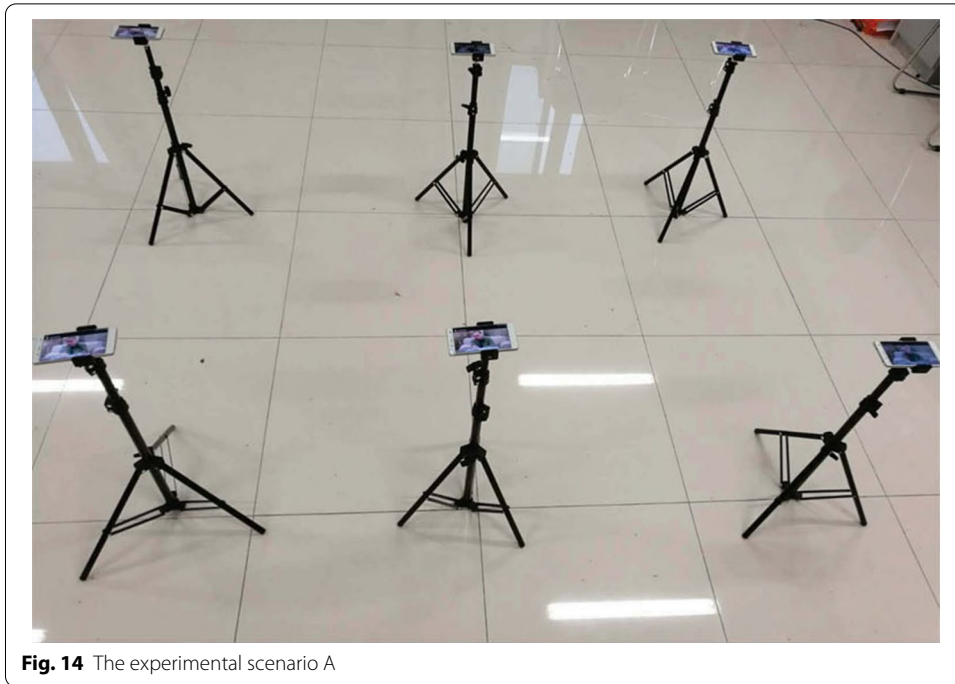
The model of the mobile device we used for testing is MI 4LTE. The operating system is Android 6.0.1. The frequency of the CPU processor is 2.45 GHz, and the running memory is 2GB. The media access control (MAC) layer protocol we used is IEEE 802.11g. Because the 5G network has not been widely deployed, and there is not any commercial device supporting D2D communications, we used the technology of WIFI direct to implement device-to-device direct connections. The links between different devices are independent of each other. Figures 14 and 15 show the real experiment scenes.

### 5.4 Analysis of experimental results

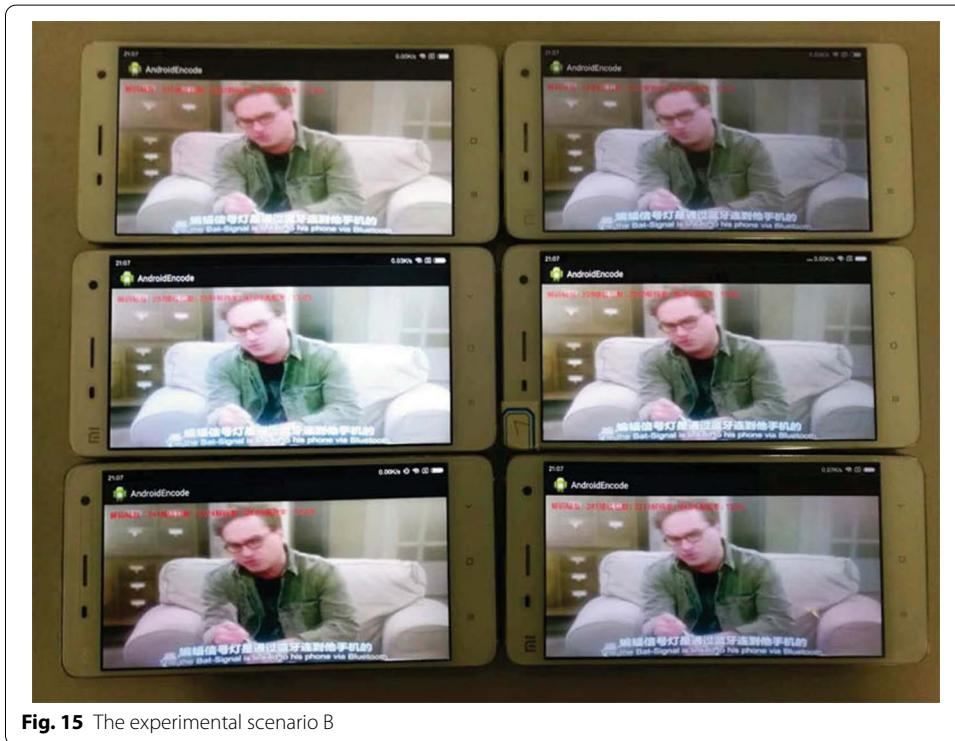
We evaluate the performance of our scheme first and compare it with the performance of traditional schemes based on replication and IDNC. Then, we use the device's successful recovery rate of the video frames as an evaluation criterion. We also evaluate the coding latency of the proposed scheme on different mobile devices. According to Fig. 16, as the transmission distance increases, the SDR of the video frames decreases in all three schemes. The video frame recovery rate in the RLNC-based scheme is higher than that in the other two schemes, which is consistent with our theoretical analysis and simulation results.

The relation between the number of redundant packets and the recovery rate of frames is shown in Fig. 17. Compared with the other two schemes, the SDR of the RLNC-based scheme is higher. With the increase in redundant data packets, the decoding efficiency is gradually improved.





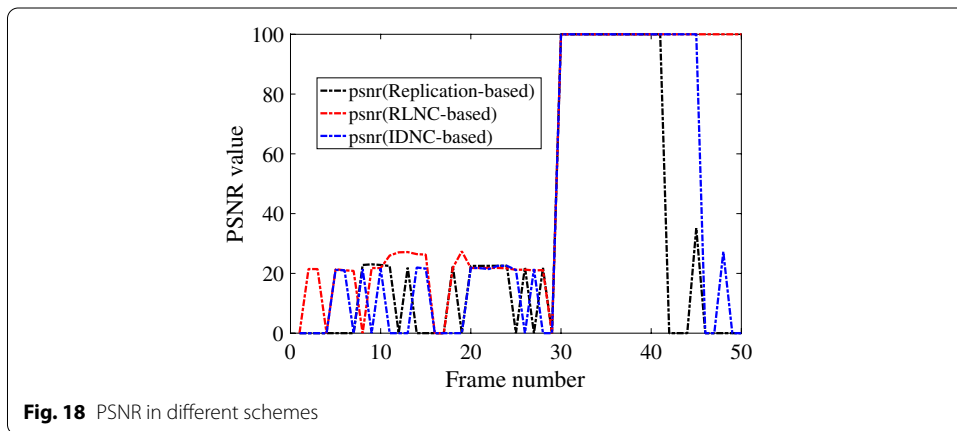
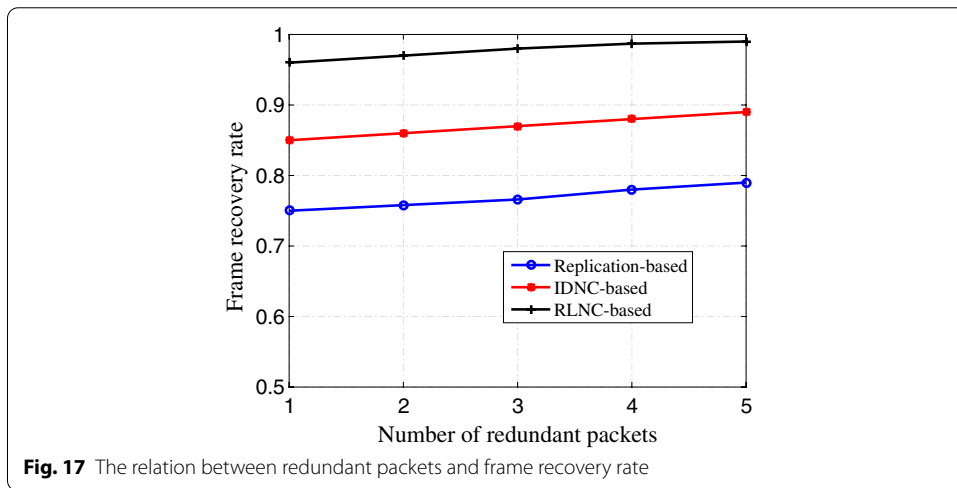
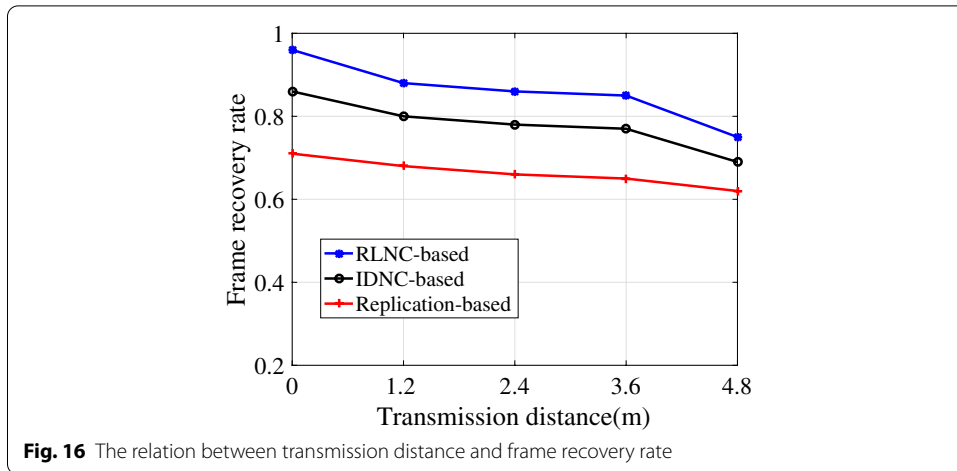
**Fig. 14** The experimental scenario A



**Fig. 15** The experimental scenario B

The peak signal-to-noise ratio (PSNR) is often used to measure the quality of pictures in the video. To evaluate the impact of the proposed scheme on video quality, we randomly selected an H.264 video stream for transmission and compared the video frames



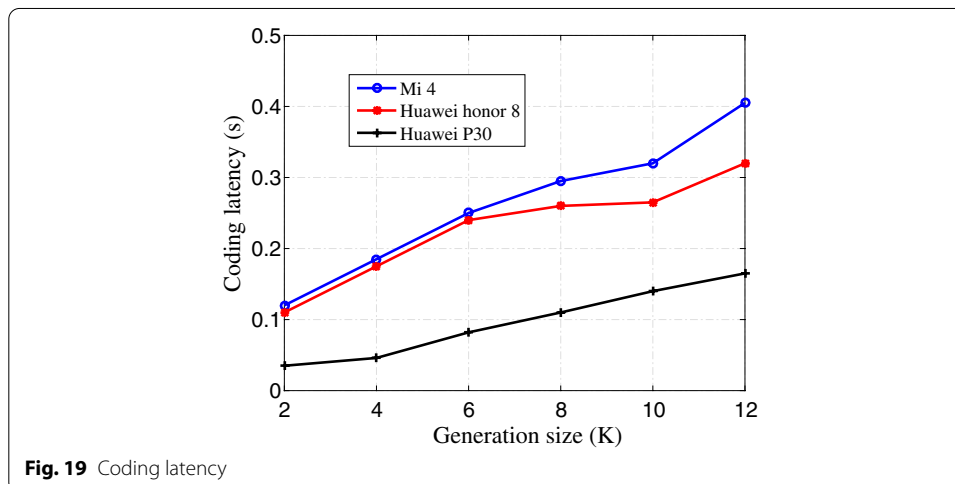


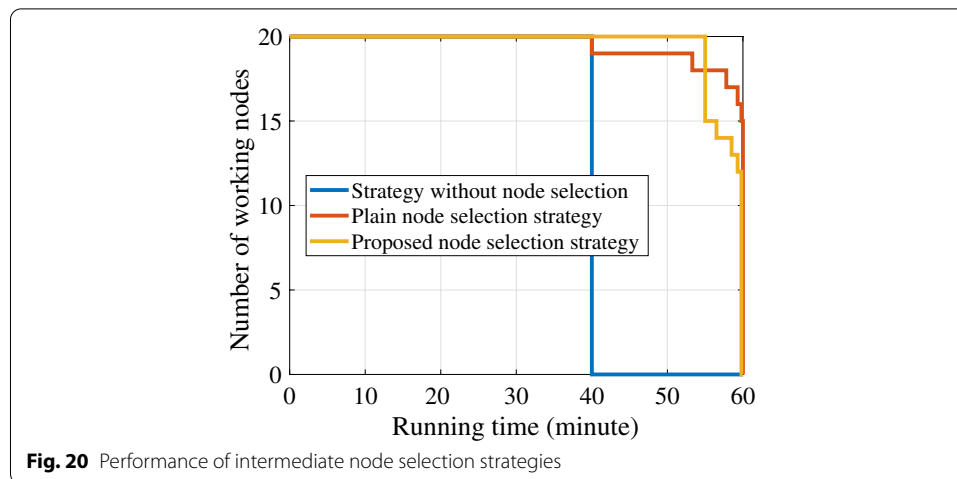
obtained after transmission with the original video stream. At the same time, we also conducted experiments with the replication-based and IDNC-based schemes. Figure 18 shows that the video obtained with the RLNC-based scheme has the best quality. In

Fig. 18, when the value of PSNR is zero, it means that the corresponding frame is missing; when the value of PSNR is 100, it means the device has completely recovered the corresponding frames.

We know that advantages always go with disadvantages. It is very difficult to have a scheme that can increase performance in one area without sacrificing another. Therefore, it is inevitable to cause coding delay after using network coding. Figures 16 and 17 show that we could obtain better performance after using network coding. Then, we need to discuss whether the coding delay makes a negative impact on user experience. According to the previous section, the time complexity of decoding operation in the network-coding-based scheme is  $O(n^3)$ , which is obtained from a theoretical perspective. In this section, we study the computational overhead from a practical perspective. We analyzed the impact of generation size  $k$  on coding latency and evaluated the coding efficiency of the proposed scheme on different mobile devices, which is shown in Fig. 19. In Fig. 19, each value is obtained by averaging ten experimental results. The size of the video we used was 1MB. From Fig. 19, with the increase in the value of  $k$ , the throughput of the encoding in the device gradually decreases. The complexity of the coding increases as the coding dimension becomes higher. According to Fig. 19, the hardware performance of mobile devices determines the coding latency. For example, the bandwidth overhead required to receive a video (1-minute duration, 1920-by-1080 resolution) is about 10MB. According to Fig. 19, the decoding rate after using network coding is far greater than the required receiving rate. Therefore, the coding latency of network coding does not have an apparent adverse effect on the user experience.

In Algorithm 3, we designed a selection strategy for intermediate nodes. The strategy needs to be evaluated in a network with many nodes. However, in our testbed, there are only six nodes. Therefore, we evaluated the performance of the proposed strategy in the simulated network, which is shown in Fig. 10. Figure 20 shows the relation between running time and the number of working nodes. After using the strategy without intermediate node selection, there is only one intermediate node, and it is assigned manually. After 40 minutes, the intermediate node runs out of energy. Because there is no subsequent node working as an intermediate node, all the nodes





in the network stop transmission. In the plain node selection strategy, there will be a new intermediate node after the previous intermediate node exhausts its energy. After 40 minutes, the intermediate nodes gradually leave the network. After using the proposed strategy, the energy overhead of intermediate nodes is balanced so that the video distribution could last for 55 minutes.

## 6 Conclusion

In order to improve the quality of video transmission in a multi-hop D2D communication environment, we propose a network-coding-based video distribution scheme. This scheme can provide additional protection to the critical information of video, which can improve the reliability during transmission. According to the experimental results, our scheme has higher stability and better video quality than the other two traditional schemes. Moreover, through the practical evaluation in our testbed, we observed that the coding delay introduced by network coding does not make a negative impact on the user experience during video playback at the receiving nodes.

### Abbreviations

D2D: Device-to-device; 5G: 5th generation; 3GPP: 3rd-generation partnership project; RLNC: Random linear network coding; P2P: Peer-to-peer; EMSCNC: Extended Multicast Scalable Video Transmission using Classification Scheduling Algorithms and Network Coding; MSVT: Multicast scalable video transmission; NCVCS: Network coding-based video conferencing system; GF: Galois field; GOP: Group of pictures; AVC: Advanced video coding; SDR: Successful decoding rate; IDNC: Instantly decodable network coding; BER: Bit error rate; MAC: Medium access control; PSNR: Peak signal-to-noise ratio; MB: Mega byte.

### Acknowledgements

The authors gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

### Authors' contributions

The authors have contributed jointly to the manuscript. All authors have read and approved the final manuscript.

### Funding

This work was supported by China Postdoctoral Science Foundation (2019M651921).

### Availability of data and material

Not applicable.

### Competing interests

The authors declare that they have no competing interests.

**Author details**<sup>1</sup> Jiangsu Key Laboratory of Big Data Security and Intelligent Processing of NUPT, 9, Wenyuan Road, Nanjing, China.<sup>2</sup> School of Computer, Nanjing University of Posts and Telecommunications, 9, Wenyuan Road, Nanjing 210023, China.<sup>3</sup> School of Computer Science and Engineering, Nanjing University of Science and Technology, 200, Xiaolingwei Road, Nanjing 210094, China.

Received: 4 August 2019 Accepted: 1 December 2020

Published online: 11 December 2020

**References**

1. L. Zhou, R.Q. Hu, Y. Qian, H. Chen, Energy-spectrum efficiency tradeoff for video streaming over mobile ad hoc networks. *IEEE J. Sel. Areas Commun.* **31**(5), 981–991 (2013)
2. C. Concolato, J.F. Le, F. Denoual, F. Mazé, E. Nassor, N. Ouedraogo, J. Taquet, Adaptive streaming of HEVC tiled videos using MPEG-DASH. *IEEE Trans. Circuits Syst. Video Technol.* **28**(8), 1981–1992 (2018)
3. X. Shen, Device-to-device communication in 5g cellular networks. *IEEE Netw.* **29**(2), 2–3 (2015)
4. N. Lee, X. Lin, J.G. Andrews, R.W. Heath, Power control for D2d underlaid cellular networks: modeling, algorithms, and analysis. *IEEE J. Sel. Areas Commun.* **33**(1), 1–13 (2015)
5. S.F. Hasan, 5g communication technology. In: Hasan SF (ed) *Emerging trends in communication networks*. Springer briefs in electrical and computer engineering, pp. 59–69. Springer, Cham (2014)
6. N. Vo, T.Q. Duong, H.D. Tuan, A. Kortun, Optimal video streaming in dense 5g networks with D2d communications. *IEEE Access* **6**, 209–223 (2018)
7. K. Doppler, M. Rinne, C. Wijting, C.B. Ribeiro, K. Hugl, Device-to-device communication as an underlay to LTE-advanced networks. *IEEE Commun. Mag.* **47**(12), (2009)
8. Y. Yan, B. Zhang, C. Li, Network coding aided collaborative real-time scalable video transmission in D2d communications. *IEEE Trans Veh Technol* **67**(7), 6203–6217 (2018)
9. P. Ostovari, J. Wu, Robust wireless transmission of scalable coded videos using two-dimensional network coding. *Comput. Netw.* **95**, 115–126 (2016)
10. L. Zhou, Mobile device-to-device video distribution: theory and application. *ACM Trans. Multimedia Comput. Commun. Appl.* **12**(3), 38–13823 (2016)
11. P. Pahlavani, M. Hundebøll, M.V. Pedersen, D. Lucani, H. Charaf, F.H. Pfitzek, H. Bagheri, M. Katz, Novel concepts for device-to-device communication using network coding. *IEEE Commun. Mag.* **52**(4), 32–39 (2014)
12. J. Yin, Y. Yang, L. Wang, X. Yan, A reliable data transmission scheme based on compressed sensing and network coding for multi-hop-relay wireless sensor networks. *Comput. Electr. Eng.* **56**, 366–384 (2016)
13. E. Datsika, A. Antonopoulos, N. Zorba, C. Verikoukis, Cross-network performance analysis of network coding aided cooperative outband D2d communications. *IEEE Trans. Wireless Commun.* **16**(5), 3176–3188 (2017)
14. Y. Wu, W. Liu, S. Wang, W. Guo, X. Chu, Network coding in device-to-device (D2d) communications underlying cellular networks. In: 2015 IEEE international conference on communications (ICC), pp. 2072–2077 (2015)
15. R. Ahlswede, C. Ning, S.R. Li, R.W. Yeung, Network information flow. *IEEE Trans. Inf. Theory* **46**(4), 1204–1216 (2000)
16. Y. Yang, C. Zhong, Y. Sun, J. Yang, Network coding based reliable disjoint and braided multipath routing for sensor networks. *J. Netw. Comput. Appl.* **33**(4), 422–432 (2010)
17. M. Ploumidis, N. Pappas, V.A. Siris, A. Traganitis, On the performance of network coding and forwarding schemes with different degrees of redundancy for wireless mesh networks. *Comput. Commun.* **72**, 49–62 (2015)
18. M. Kim, K. Park, W.W. Ro, Benefits of using parallelized non-progressive network coding. *J. Netw. Comput. Appl.* **36**(1), 293–305 (2013)
19. S.R. Li, C. Ning, R.W. Yeung, On theory of linear network coding. In: *Proceedings. International Symposium on Information Theory, 2005. ISIT 2005.*, pp. 273–277 (2005)
20. T. Ho, M. Médard, R. Koetter, D.R. Karger, M. Effros, J. Shi, B. Leong, A random linear network coding approach to multicast. *IEEE Trans. Inf. Theory* **52**, 4413–4430 (2006)
21. Z. Lin, Y. Wang, Y. Lin, L. Wu, Z. Chen, Analysis and optimization of rlnc-based cache placement in 5g d2d networks. *IEEE Access* **6**, 65179–65188 (2018)
22. J. Huang, S. Huang, C. Xing, When d2d meets network coding: a tutorial case study. In: *Proceedings of the International Conference on Research in Adaptive and Convergent Systems*, pp. 146–151 (2017). ACM
23. J. Connelly, K. Zeger, Capacity and achievable rate regions for linear network coding over ring alphabets. *IEEE Trans. Inf. Theory* **65**(1), 220–234 (2019)
24. M.S. Karim, S. Sorour, P. Sadeghi, Network coding for video distortion reduction in device-to-device communications. *IEEE Trans. Veh. Technol.* **66**(6), 4898–4913 (2017)
25. K. Nguyen, T. Nguyen, S.C. Cheung, Video streaming with network coding. *J. Signal Process. Syst.* **59**(3), 319–333 (2010)
26. D. Ferreira, R.A. Costa, J. Barros, Real-time network coding for live streaming in hyper-dense WiFi spaces. *IEEE J. Sel. Areas Commun.* **32**(4), 773–781 (2014)
27. O.B. Rhaïem, F.C. Lamia, W. Ajib, Network coding-based approach for efficient video streaming over MANET. *Comput. Netw.* **103**, 84–100 (2016)
28. L. Wang, Y. Yang, W. Zhao, Network coding-based multipath routing for energy efficiency in wireless sensor networks. *EURASIP J. Wireless Commun. Netw.* **2012**(1), 115 (2012)
29. L. Wang, Z. Yang, L. Xu, Y. Yang, NCVCS: network-coding-based video conference system for mobile devices in multicast networks. *Ad Hoc Netw.* **45**, 13–21 (2016)
30. O.B. Rhaïem, L.C. Fourati, W. Ajib, QoS improvement for video streaming over MANET using network-coding. In: 2015 IEEE 82nd vehicular technology conference (VTC2015-Fall), pp. 1–5 (2015)

31. P. Kitsos, G. Theodoridis, O. Koufopavlou, An efficient reconfigurable multiplier architecture for Galois field GF(2<sup>m</sup>). *Microelectron. J.* **34**(10), 975–980 (2003)
32. T. Lehnigk-Emden, N. Wehn, Complexity evaluation of non-binary Galois field LDPC code decoders. In: 2010 6th International Symposium on Turbo Codes Iterative Information Processing, pp. 53–57 (2010)
33. S. Park, D.H. Cho, Random linear network coding based on non-orthogonal multiple access in wireless networks. *IEEE Commun. Lett.* **19**(7), 1273–1276 (2015)
34. P. Vingelmann, M. Pedersen, F. Fitzek, J. Heide, Multimedia distribution using network coding on the iphone platform. In: Proceedings of the 2010 ACM Multimedia Workshop on Mobile Cloud Media Computing, pp. 3–6 (2010). ACM
35. P.S. Stanimirović, M.D. Petković, Gauss-Jordan elimination method for computing outer inverses. *Appl. Math. Comput.* **219**(9), 4667–4679 (2013)
36. N. Bahri, N. Belhadj, M.A.B. Ayed, N. Masmoudi, T. Grandpierre, M. Akil, Real-time H264/AVC high definition video encoder on a multicore DSP TMS320c6678. In: International Conference on Computer Vision and Image Analysis Applications, pp. 1–6 (2015)
37. D.C. Nguyen, T.S. Nguyen, C.C. Chang, H.S. Hsueh, F.R. Hsu, High embedding capacity data hiding algorithm for H.264/AVC video sequences without intraframe distortion drift (2018)
38. I.U. Khan, M.A. Ansari, S.S. Hasan, K. Khan, Evaluation and analysis of rate control methods for H. 264/AVC and MPEG-4 video codec. *Int. J. Electr. Comput. Eng.* (2088-8708) **8**(2) (2018)
39. C.F. Chiasserini, E. Viterbo, C. Casetti, Decoding probability in random linear network coding with packet losses. *IEEE Commun. Lett.* **17**(11), 1–4 (2013)
40. R. Khan, G.K. Kurt, I. Altunba, Decoding failure probability of random network coding systems in fading channels. In: 2015 23rd Signal Processing and Communications Applications Conference (SIU), pp. 2050–2053 (2015)
41. S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, J. Crowcroft, XORs in the air: practical wireless network coding. *IEEE/ACM Trans. Netw.* **16**(3), 497–510 (2008)
42. A. Douik, S. Sorour, T.Y. Al-Naffouri, M.S. Alouini, Instantly decodable network coding: from centralized to device-to-device communications. *IEEE Commun. Surv. Tutorials* **19**(2), 1201–1224 (2017)

### Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)

---