


RESEARCH

Open Access



Probabilistic cooperative coded forwarding for broadcast transmissions in industrial mobile edge communications

Wenjun Huang^{*} , Xu Li and Yanan Liang

^{*}Correspondence:
16111027@bjtu.edu.cn

School of Electronic
and Information Engineering,
Beijing Jiaotong University,
Beijing, People's Republic
of China

Abstract

Mobile edge computing (MEC) is considered as a key enabler for the industrial internet of things (IIoT) to cope with the ever-increasing communication and computing demands of nodes. In consideration of the limited resource of the IIoT nodes, it is necessary to design cost-effective multi-hop data transmission schemes for mobile edge IIoTs. However, most of the traditional schemes have to spend enormous cost to meet the reliability requirement, which cannot support timely information processing of MEC-based IIoT. In this article, the probabilistic cooperative coded forwarding (PCCF) scheme for multi-hop data transmission in mobile edge IIoTs is proposed to address the above problem. First, the data packets are encoded at a source IIoT node using the systematic sparse network coding (SSNC) mechanism, then the source broadcasts the coded packets to its one-hop neighbors. To minimize the required number of redundant coded packets, the sparsity of coded packets is optimized. Second, the nodes which received the packets will become volunteer relay nodes and forward the coded packets using the cooperative coded forwarding (CCF) mechanism. The volunteer relays first forward the received coded packets with a forwarding probability, and then re-encode a pair of received coded packets and broadcast the re-encoded packet with a re-encoding probability. To guarantee the broadcast performance while minimizing the transmission number at relay nodes, the feasible forwarding and re-encoding probability are provided. Third, the receiver nodes will try to decode the received coded and re-encoded packets and recover data packets without sending acknowledgments. Finally, through a series of experiments, we verify the accuracy of analytical approximations and also find out the optimal sparsity of coded packets and the existence of minimum transmission numbers. These provide insights for further optimization of multi-hop data transmission in mobile edge IIoTs.

Keywords: Industrial internet of things, Mobile edge computing, Probabilistic forwarding, Network coding, Wireless communication

1 Introduction

With the rapid development of fifth-generation (5G) mobile communication technologies, the industrial internet of things (IIoT) [1] become increasingly intelligent and flexible in recent years. One of the key enabling technologies of IIoT is mobile

edge computing (MEC), which integrates the cloud computing capabilities on the mobile IIoT nodes in the edge networks to accelerate information processing [2, 3]. The numerous connected communication nodes in IIoT can therefore collect industrial data and efficiently process them in real-time or near real-time, which give birth to the cloud-edge hybrid computing method [4]. The IIoT nodes can also process industrial data among the neighboring nodes in a self-organizing manner, rather than transmitting them to the cloud center, which meets strict timeliness requirements [5]. While the ever-growing data on the wireless links between the cloud center and the edge network are offloaded, the substantial stress of data sharing lies on the edge network and put forward requirements to highly efficient multi-hop data transmission techniques [6].

As a promising technology for efficient multi-hop transmission, network coding (NC) helps to improve the wireless multi-cast and broadcast performance in distributed mobile ad hoc networks (MANET) by coded forwarding [7, 8]. In [7], the data packets are encoded at the source node using random linear network coding (RLNC) and forwarded by flooding, and the receiver nodes which receive a sufficient number of coded packets can decode the data packets without sending feedback for re-transmissions. To reduce the cost of flooding, the coded packets can also be forwarded in a deterministic [8] or probabilistic [9] manner, where the relay nodes only forward the coded packets to some selected neighbors or with a certain probability. In recent years, the application of network coding have been extended to MEC in 5G networks [10, 11]. In [10], the mobile edge nodes may encode their delay-sensitive tasks or compute-and-upload the delay-tolerant tasks and send to the cloud center, which helps to offload the task with flexible computation load and uploading time. In [11], the relay nodes helps to encode and forward the packets to the next-hop receivers rather than directly forwarding them, where the relay-coded packets help the receivers to recover lost packets, and the transmission number at the relays can be reduced. The advantages of network coding meet the requirement of efficient multi-hop data transmission in mobile edge networks.

The main challenges of deploying network coding in mobile edge IIoT mainly lies on the constraints of battery capacity and computational resource of the IIoT nodes, and the dynamic and intermittent wireless links. First, the limited communication and computation capabilities of IIoT nodes mean that the design of network coding mechanism should be sufficiently simple, time-efficient and adaptive for dynamic environments. The RLNC mechanism, though simple to be implement, turns out to be inefficient due to the “all-or-nothing” effect, where the receivers have not receive sufficient number of linear irrelevant packets cannot start decoding, which incurs large delay of recovering data packets and cannot meet the QoS requirements of mobile edge IIoT [12]. Second, the dynamic and intermittent links will also jeopardize the performance of multi-hop transmissions. While the flooding mechanism will quickly exhaust the resource of mobile edge IIoT, the neighbor information maintenance of the deterministic forwarding mechanism for selecting relay nodes also invites huge overhead especially in dynamic environments, where the information of neighbors may not be timely maintained. The probabilistic forwarding mechanism, which is easy to be realized as the nodes only forward packets with a certain probability and save the maintenance cost. However, probabilistic forwarding may prune some necessary links and terminates the multi-hop

transmission unexpectedly, and so the forwarding probability should be appropriately selected.

In this article, we focus on combine the advantages of network coding and probabilistic forwarding mechanism in MEC-enabled IIoT scenarios. To this end, we propose the probabilistic cooperative coded forwarding (PCCF) scheme by jointly optimizing the encoding procedures at the source nodes and the forwarding process at the multi-hop relay IIoT nodes. First, the source node adopts a systematic encoding framework and generates sparse coded packets for a generation of data packets and broadcast them to its one-hop neighbor nodes, which eliminates the “all-or-nothing” effect of RLNC as the receiver nodes can start decoding more quickly. The mechanism is termed as systematic sparse network coding (SSNC) and the sparsity of the coding vector of a coded packet is optimized to achieve higher decoding probability. Second, the nodes which received the coded packets from a source will become volunteer relay nodes. These nodes will first forward the coded packets with a certain probability, and also re-encode the received coded packets to enhance the performance of probabilistic forwarding. This coded-and-forward mechanism is called cooperative coded forwarding (CCF). To simplify the operations at relay nodes using CCF, we provide an approximation model based on the percolation theory to appropriately determine the forwarding probabilities in different mobile edge IIoT scenarios with various packet erasure rate, the number of one-hop neighbors and the delivery ratio requirements. To our best knowledge, it is the first analytical model that is capable to predict the multi-hop broadcast performance for mobile edge IIoTs. Third, the receiver nodes will decode the received coded and re-encoded packets and try to recover the data packets from the same generation. Once it received a systematic coded packet, a receiver node can instantly decode the corresponding data packet and helps to decode the other coded packets.

The rest of this article is organized as follows. In Sect. 2, the related works of network coding and multi-hop data transmission schemes are introduced. The method design of the proposed PCCF scheme is presented in Sect. 3. The experiment results of PCCF are provided in Sect. 4. Finally, the conclusion is drawn in Sect. 5.

2 Related works

The cooperative data transmission is one of the fundamental functions of mobile edge networks, which can be considered as a broadcasting problem. The most basic flooding mechanism forces all the network nodes to directly forward the newly received packets to the others, which will quickly exhaust the channel resource and the received packets at destination nodes may be highly likely to be redundant [7].

To alleviate the unacceptable cost of flooding in mobile edge networks, some transmissions need to be pruned. One feasible method is probabilistic forwarding [9], where the nodes will only forward the newly received packet with a predetermined probability. As the broadcast transmissions of nodes are pruned without distinction, some necessary transmission may also be eliminated. Another method is deterministic forwarding [13], where the nodes will select a few neighbors for rebroadcasting according to their knowledge of network topology. To optimize the efficiency of deterministic forwarding, the nodes should perform sufficient maintenance on the topology information they have, which consumes a lot of channel resources. Besides,

the packet erasure effect due to link outage will also affect the multi-hop transmissions. To improve the resilience to failures and minimize the cost, forwarding mechanisms should be more reliable and resource-efficient.

Network coding allows the source and intermediate nodes to encode the packets before transmission rather than doing nothing. This encoding method brings the connection between different data packets, which enables the receiver nodes to start decoding after receiving sufficient coded packets without sending feedback for retransmission. To this end, the source and relay nodes should dynamically determine the number of coded packets and also their forwarding strategy to ensure that the receiver nodes have a sufficient number of coded packets [8]. There are mainly two types of network coding scheme, which characteristics as described as follow.

One is the most commonly discussed RLNC scheme, the encoded packets are generated as random linear combinations of the original data packets, and the coding coefficients are selected from a predetermined finite field. The decoding performance of RLNC is related to the rank of the receiving coding matrix [14], which represents the linear irrelevance among the received coded packets. In order to keep the generated RLNC packets linear irrelevant, the size of the finite field of RLNC should be sufficiently large. As a result, the encoding and decoding procedure of RLNC is computationally complex.

Another scheme is exclusive-or (XOR) based network coding, which is also known as COPE [15]. The receiver nodes will broadcast the information on whether the packets are received or not, like sending collective feedback. An intermediate node will listen and discover the “coding opportunity” and deterministically forward the coded packets to desire receivers [16]. In the XOR-based NC scheme, the packets are encoded with bitwise XOR operation, therefore it can also be considered as a variant of RLNC using the binary finite field. Though XOR-based NC is much simpler than RLNC schemes operating in large finite fields, the broadcasting of receiving status in COPE will lead to resource consumption just like the topology maintenance in deterministic forwarding.

To decrease the complexity of RLNC, the source nodes can increase the sparsity of coded packets [17, 18], which means that the coding coefficients are more likely to be zero. The increasing sparsity of coded packets will reduce the computational cost of both the encoding and decoding processes. The coded packets can also be generated in a systematic manner [19, 20], where the data packets will be first embedded into the coded packets one by one. These systematic packets can be instantly decoded and the recovered data packet can accelerate the decoding process of nodes. To the best of our knowledge, the rank distribution of sparse network coding has no closed-form expressions, which is usually provided in the approximation versions from empirical simulation results. Moreover, in a systematic network coding framework, the analysis of decoding probability should jointly consider the relationship between the received systematic and sparse coded packets.

Despite the aforementioned existing works, various technical challenges associated with the design of network coding scheme for cost-efficient multi-hop transmission in MEC-enabled IIoT, including how to determined the forwarding methods and parameters, the number of coded packets, and the sparsity of coding vectors in order to optimize the decoding performance at the receiver nodes.

3 Methods

3.1 Network model

We consider a MEC-enabled IIoT consisting of a base station (BS) equipped with a MEC server and N randomly distributed IIoT nodes in the edge network. The MEC server has direct connection to the data center at the core cloud. The IIoT nodes in the mobile edge IIoT are assumed to be located inside a square area $[0, B]^2$ on the ground where B is the edge of the square, and the BS is located at the origin of the area. We can assume that the MEC server at the BS has higher computation performance than the IIoT edge nodes. The network model is displayed in Fig. 1.

We assume a multi-hop mobile edge network scenario, where the BS can only cover a fraction of nodes. Hence the IIoT nodes which have received the packets from the BS should forward them to the rest of the nodes in the network. For the convenience of analysis, the coverage radii of the BS and the IIoT nodes are assumed to be the same and denoted as $\delta < B$, and any two nodes with distance closer than δ is assumed to have a direct link with a common packet erasure probability $\rho \in (0, 1)$. This kind of network can be modeled as a Gilbert disk graph [21], one of the most studied random geometric graphs (RGG) in the literature. Let $\mathcal{N}(v)$ denote the one-hop neighbor of node v . The cardinality of $\mathcal{N}(v)$ is referred as the degree of node v , i.e. $d_v \triangleq |\mathcal{N}(v)|$, which follows a Poisson distribution of mean $\varphi \triangleq \mathbb{E}(d_v) = \lambda\pi\delta^2$.

3.2 Scheme design

The probabilistic cooperative coded forwarding scheme consists of two network-coding based mechanisms. First, the BS deploys the systematic sparse network coding (SSNC) mechanism, in which it will organize k data packets as a packet generation $\mathbf{x} = \{x_i | i = 1, \dots, k\}$ and encoded them into $n \geq k$ source-coded packets (SCPs). It broadcasts the SCPs to the IIoT nodes within its coverage. Second, an IIoT node that has received the SCPs will perform cooperative coded forwarding (CCF) as a volunteer relay

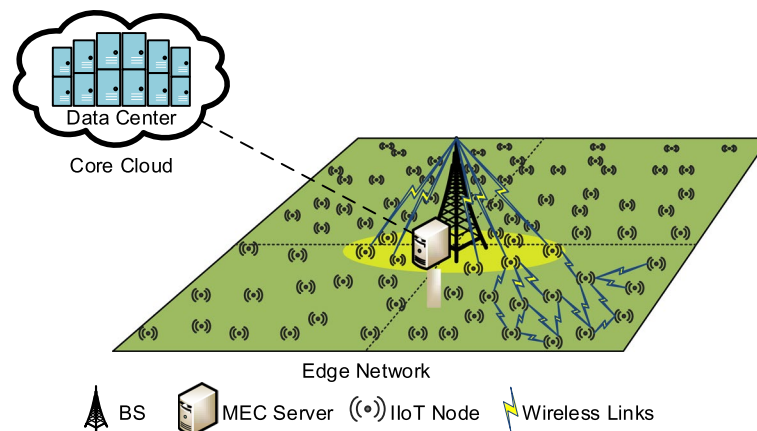


Fig. 1 Architecture of MEC-enabled industrial communication network. The figure displays network model. We consider a MEC-enabled IIoT consisting of a base station (BS) equipped with a MEC server and N randomly distributed IIoT nodes in the edge network. The MEC server has direct connections with the core cloud with data centers. The IIoT nodes in the edge network are assumed to be located inside a square $[0, B]^2$ on the ground where B is the edge of the square, and the BS is located at the origin of the edge network square

node, where it first directly forward the SCPs in a probabilistic manner, and then re-encode the received SCPs into some relay-coded packets (RCPs) and broadcast the RCPs to its one-hop neighbors. In general, SSNC is applied at the source nodes and CCF is adopted at the relay nodes. To accelerate the encoding and decoding process at the BS and IIoT nodes respectively, both the network coding mechanisms should be designed with low computational complexity and high efficiency. The diagram of SSNC and CCF is presented in Fig. 2, and the detailed procedures of these two mechanisms is illustrated as follows.

3.2.1 Systematic sparse network coding (SSNC)

In the encoding process of the j th SCP y_j , $j \in \{1, \dots, n\}$, the source node will first select some data packets in the same data packet generation. Next the source selects a coding coefficient $c_{j,i}$ for a selected data packet x_i in a Galois field F_q of size q . The source then generates the coded packet y_j by performing addition and multiplication in F_q , and the encoding result is given as

$$y_j = \mathbf{c}_j \mathbf{x} = \sum_{i=1}^k c_{j,i} x_i \tag{1}$$

To indicate the coding coefficients corresponding to the carried data packets, the j th SCP should also contain a coding vector $\mathbf{c}_j = \{c_{j,i} | i = 1, \dots, k\}$. Let $\mathcal{P}_j = \{i | c_{j,i} = 1\}$ denote the indices of data packets that y_j carries. Let $d_j = |\mathcal{P}_j|$ denote the cardinality of

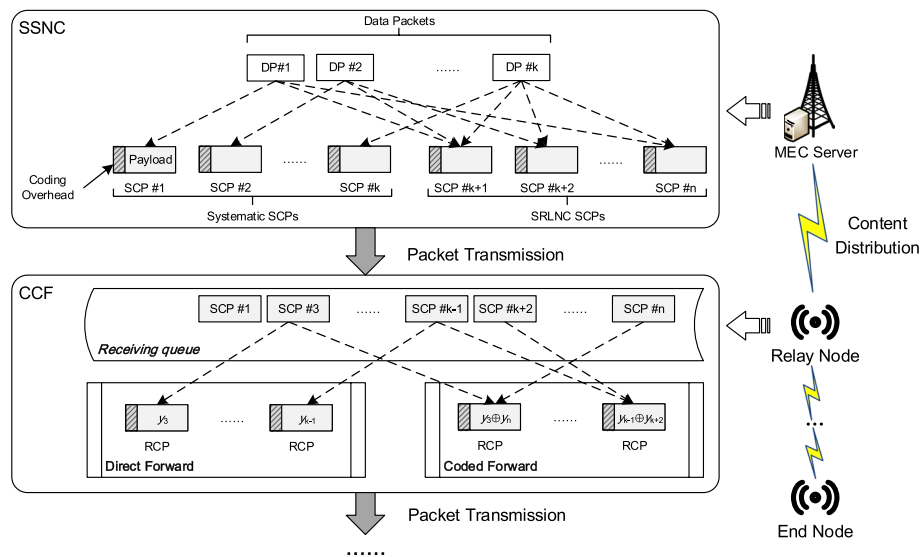


Fig. 2 Diagram of the network-coding based mechanisms of PCCF. The figure displays the scheme of systematic sparse network coding (SSNC) mechanism and cooperative coded forwarding (CCF). The probabilistic cooperative coded forwarding scheme consists of two network-coding based mechanisms. The BS deploys the systematic sparse network coding (SSNC) mechanism, in which it will organize k data packets as a packet generation $\mathbf{x} = \{x_i | i = 1, \dots, k\}$ and encoded them into $n \geq k$ source coded packets (SCPs). It will broadcast the SCPs to the IIoT nodes within its coverage. An IIoT node which has receive the SCPs will perform cooperative coded forwarding (CCF), where it first directly forward the SCPs in a probabilistic manner, and then encode the received SCPs into some relay coded packets (RCPs) and broadcast them to its one-hop neighbors. In general, SSNC is a source-based network coding mechanism and CCF is related to the relay nodes

\mathcal{P}_j , which can be referred as the degree of coded packet y_j and also the weight of its coding vector \mathbf{c}_j .

The selection of coding coefficients will determine the complexity of the encoding and decoding process as well as the successful decoding probability. The measures may include minimizing the degree of coded packets so as to increase the sparsity of coding vectors. To this end, SSNC uses a systematic network coding framework and the encoding operation can be separated into two phases:

- In the first phase of encoding, the BS encodes a generation of k data packets in a systematic manner, which means the j th will only contain the information of the j th data packet, i.e. $c_{j,i} = \mathbf{1}(j = i)$ and $\mathcal{P}_j = \{j\}$, $j = 1, \dots, k$. Therefore, the first k coded packets can be termed as systematic SCPs with degree 1. They are instantly decodable, which help the receiver nodes to accelerate their decoding process.
- In the second phase of encoding, a number of r randomly generated SCPs are produced, i.e., the coding coefficients of \mathbf{c}_j , $j = k + 2, \dots, n$ are randomly selected in F_q . In this article, we adopt the binary finite field $F_2 = \{0, 1\}$, and the coding coefficients should either be one or zero. Let $p_0 \in (q^{-1}, 1)$ denote the probability that a coding coefficient is zero, which is equivalent to the sparsity of coding vectors [14]. As p_0 increases, the expected degree of SCPs will decrease, which means the number of encoding operation will also decrease. However, a larger p_0 makes the coding vectors of randomly generated SCPs too sparse to contribute new data packets during the decoding process. Henceforth, p_0 may have its optimal value and should be designed accordingly.

An IIoT node can initiate the decoding process when it receive a systematic SCP from its neighbors. In the receiving packet queue of the node, there may be systematic and randomly generated SCPs. The node will first decode all the received systematic SCPs into data packets. Next, it will try to decode the randomly generated SCPs with sparse coding vectors. An SCP is said to be available for decoding if its coding vector contains only one new and nonzero coefficient with respect to the decoded data packets. The decoding process will be iteratively performed until there are no available SCPs to be decoded.

An example of the encoding and decoding process of SSNC is provided as follow. Suppose the source encode five data packets x_i , $i = 1, \dots, 5$ in two five systematic SCPs and three randomly generated SCPs denoted as $x_1 \oplus x_2, x_2 \oplus x_4$ and $x_3 \oplus x_5$ with a common low degree 2. One receiver have receive only two out of five systematic SCPs containing x_1 and x_3 . It also receives all the three randomly generated SCPs. Therefore it can first decoded out x_1 and x_3 , and then decode x_2 by performing $x_1 \oplus (x_1 \oplus x_2) = x_2$. Similarly, x_4 and x_5 can be subsequently decoded.

3.2.2 Cooperative coded forwarding (CCF)

The operation of cooperative coded forwarding is described as follows. First, an IIoT node will either forward or ignore a newly received SCP with probability $\omega \in (0, 1)$ or $1 - \omega$, respectively. Second, once a node has received more than two SCPs, it will start the re-encoding and forwarding procedure. The node first chooses two different SCPs y_a and y_b from its received SCP queue. It directly encodes them into a relay coded packet

(RCP) $z_{a,b} = y_a \oplus y_b$ with probability $\xi \in (0, 1)$ using the exclusive or (XOR) method. Since the SCPs are encoded in $GF(2)$, the XOR operation is equivalent to the addition process in SSNC. Even if a node have received neither y_a nor y_b , the reception of $z_{a,b}$ can still be considered as an source generated SCP and will not interfere the decoding process of SSNC.

3.3 Performance model

In this subsection, we focus on modeling the performance of PCCF in a mobile edge IIoT, which includes the broadcast performance of CCF and the decoding performance of SSNC. On the one hand, the successful receiving probability of SCPs η under CCF mechanism is determined by the decisive parameters including the forwarding probability of SCPs ω , the coded forwarding probability of RCPs ξ and the packet erasure probability ρ . These parameters will eventually affect the expected number of relaying neighbors φ . However, it turns out that it is quite difficult to select ω and ξ because of the dominance of complicated boundary effects [22]. To our best knowledge, the closed-form expression of the relationship between η and φ is not given in the literature. Therefore we try to provide the approximated model of η under different ω and ρ according to the experimental results, and then analyze the effect of RCPs on the successful receiving probability of SCPs with different value of ξ . On the other hand, as the SCPs generated by SSNC are forwarded with the CCF mechanism, the successful decoding probability of SCP is jointly affected by the sparsity of coding vectors p_0 and the successful receiving probability of SCPs and RCPs at the receivers. The existing analysis of sparse network coding mainly focus on the optimization of the sparsity, while in our proposed PCCF scheme the reception of systematic SCPs and random SCPs has different impacts in the decoding process of receivers. Thus we separately analyze the receiving numbers of these two types of SCPs and model the decoding process as a Markov chain.

The parameters of PCCF considered in this article are summarized in Table 1.

3.3.1 Successful receiving probability of SCPs

In this subsection, we will first analyze the relationship between the forwarding probability ω and the occurrence of continuum percolation, which is the necessary condition for reliable multi-hop transmission. Then we will turn to the coded forwarding probability ξ and analyze its effect on the successful decoding probability at the receiver nodes.

The percolation condition of probabilistic forwarding under probability ω We start to model the broadcast performance of probabilistic forwarding from the simplest case of an ideal flooding scenario without channel outage, i.e. $\omega = 1, \rho = 0$. We build the model according to the percolation theory [23]. In this scenario, all the nodes are effective relay nodes. Denote the set of nodes that can connect to the origin W_0 containing the origin 0 . Let $\eta = |W_0|/N$ denote the fraction of network nodes that are able to connect to the origin. where $|W_0|$ is the cardinality of W_0 . The notation η is also the probability that a network node can successfully receive a typical packet.

As φ increases, η also increase as there are more nodes in W_0 . Define $\eta(\varphi)$ as a function of φ . Kumar et al. [9] introduces the concept of “near-broadcast” when the value of η is close to 1. Given $\epsilon \in (0, 1)$, let φ_ϵ denote the minimum value of φ that a near-broadcast could happen, which is given as

Table 1 Summary of parameters

| Notation | Meaning |
|--------------------|---|
| λ | Node density |
| δ | One-hop coverage radius |
| φ | Number of one-hop neighbors / Node degree |
| ρ | Packet erasure probability |
| ω | Forwarding probability of SCPs |
| ω_ρ | Equivalent forwarding probability of SCPs under packet erasure probability ρ |
| ξ | Coded-and-forwarding probability of arbitrary SCP pairs |
| η | Successful delivery probability |
| η_r | Successful delivery probability of a network with volunteer relay nodes only |
| η_s | Coverage probability of silent nodes by relay nodes |
| W_0 | The set of nodes that can connect to the origin $\mathbf{0}$ |
| ϵ | Fraction of nodes that are not covered in a near-broadcast |
| φ_ϵ | Minimum value of φ that a near-broadcast could happen |
| a_c | Critical mean node degree for continuum percolation |
| x_i | i th data packet |
| y_j | j th source coded packet (SCP) |
| c_{ji} | the coding coefficient of the i th data packet in the j th SCP |
| z | Relay coded packet (RCP) |
| G_j | Connected forwarding graph of y_j |
| V_j | Nodes that have received the y_j |
| E_j | Links that have successfully forwarded y_j |
| $\mathcal{N}(v)$ | One-hop neighbors of node v |
| r | Number of decoded SCPs |
| c | Number of received RCPs |

$$\varphi_\epsilon = \inf \{ \varphi | \mathbb{E}[\eta(\varphi)] > 1 - \epsilon \} \tag{2}$$

The goal of this subsection is to determine the value of φ_ϵ with given N and ϵ , so as to appropriately set the parameters of ω and ξ . However, it turns out that it is quite difficult to find because of the dominance of complicated boundary effects [22], and the closed-form expression of $\eta(\varphi)$ is not given in the literature. Thus we turn to the simulation results of $\eta(\varphi)$, which is provided in [24] by setting $N = 100$, and finally approximate it by a Weibull distribution given as

$$\eta(\varphi) \simeq 1 - e^{-(2\varphi/3\pi)^\pi} \tag{3}$$

Next we consider the case that only a fraction of nodes will forward the packets after successful reception under perfect channels, i.e. $\omega < 1$ and $\rho = 0$. The volunteer relay nodes provide coverage of the silent nodes in the rest of the network, which can be modeled as a site percolation [25]. To analyze the value of connected probability η in such case, we first remove the silent nodes and calculate the connected probability of the volunteer relay nodes, which is denoted as η_r . The value of η_r can be calculated by substitute the mean degree of volunteer relay nodes $\varphi_r = \omega\varphi$ into Eq. (3). As a result, we have $\omega \geq \varphi_\epsilon / \lambda\pi\delta^2$. Then we randomly scatter the rest silent nodes back to the scenario. Let η_s denote the probability that a randomly placed silent node can be covered by a relay node

in the connected component W_0 , which can be approximated as $\eta_s \simeq \eta_r$ when η_r is large enough. And the connected probability of the whole network should be

$$\eta_\omega = \omega\eta_r + (1 - \omega)\eta_s \simeq \eta_r = \eta(\omega\varphi) \tag{4}$$

Finally, we turn to a more practical case where wireless links between nodes are intermittent with $\rho > 0$. This can be considered as a bond percolation [25] and is equivalent to a site percolation where the relay nodes proactively forward the SCPs with probability $\omega_\rho = \omega\rho$. Thus we can substitute the direct forward probability ω by ω_ρ in Eq. (4) to obtain the result of successful receiving probability.

The effect of coded forwarding under probability ξ Let $G_j = \langle V_j, E_j \rangle$ denote the connected forwarding graph of the j th SCP y_j , where V_j is the set of nodes that has received y_j , and E_j contains the links that have successfully delivered y_j . Consider two different forwarding graph G_a and G_b , and a common node $v \in V_a \cap V_b$. Denote the nodes covered by v in the two graphs as $\mathcal{N}_a(v) = \mathcal{N}(v) \cap V_a$ and $\mathcal{N}_b(v) = \mathcal{N}(v) \cap V_b$ respectively. Suppose v is going to encode y_a and y_b into a RCP $z_{a,b} = y_a \oplus y_b$ and forward it to its one-hop neighbors. A one-hop neighbor node $u \in \mathcal{N}(v)$ of v can receive $z_{a,b}$ with probability $1 - \rho$ due to the packet erasure effect. If u has already received y_a and y_b , i.e. $u \in \mathcal{N}_a(v) \cap \mathcal{N}_b(v)$, the receiving of $z_{a,b}$ become redundant. If u has only received y_a , the receiving of $z_{a,b}$ helps it to decode y_b , and the node u is added to V_b . The link $e(u, v)$ will also be involved in E_b . Similarly, the receiving of y_b and $z_{a,b}$ will add u into V_a . If u have received neither y_a nor y_b , the receiving of $z_{a,b}$ will not change G_a or G_b . The effect of this coded forwarding action on the broadcast performance can therefore be regarded as the modification of the two corresponding forwarding graphs G_a and G_b , which is shown in Fig. 3.

To quantify the modification effect on the forwarding graphs, we analyze the relationship between the coded forwarding probability ξ and the number of decoded SCPs at an IIoT node. Let r denote the number of decoded SCPs at a node. Let c denote the number of received RCPs at a node. The decoding process of RCPs can be modeled as a Markov chain consisting of a series of state $s_n(r)$. Denote the single step transition probability from $s_n(r)$ to $s_n(r + i)$ as $p_n(r, r + i)$, whose expression is given as

$$p_n(r, r + i) = \begin{cases} \frac{\binom{r}{1} \binom{n-r}{1}}{\binom{n}{2}} = \frac{2r(n-r)}{n(n-1)}, & i = 1, r = 1, 2, \dots, n-1 \\ 1 - \frac{2r(n-r)}{n(n-1)}, & i = 0, r = 1, 2, \dots, n \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Let $\mathbf{P}_{n \times n} = \{p_n(i, j) | i, j = 1, 2, \dots, n\}$ denote the transition probability matrix. Let $\mathbf{w}^{(i)} = \{w^{(i)}(n, r) | r = 0, 1, \dots, n\}$ denote the probability distribution function (pdf) of r after i steps. Since r is a binomial random variable of parameter n and η , i.e. $r \sim \text{Bin}(n, \eta)$, the pdf of r before decoding RCPs can be written as

$$w^{(0)}(n, r) = \binom{n}{r} \eta^r (1 - \eta)^{n-r} \tag{6}$$

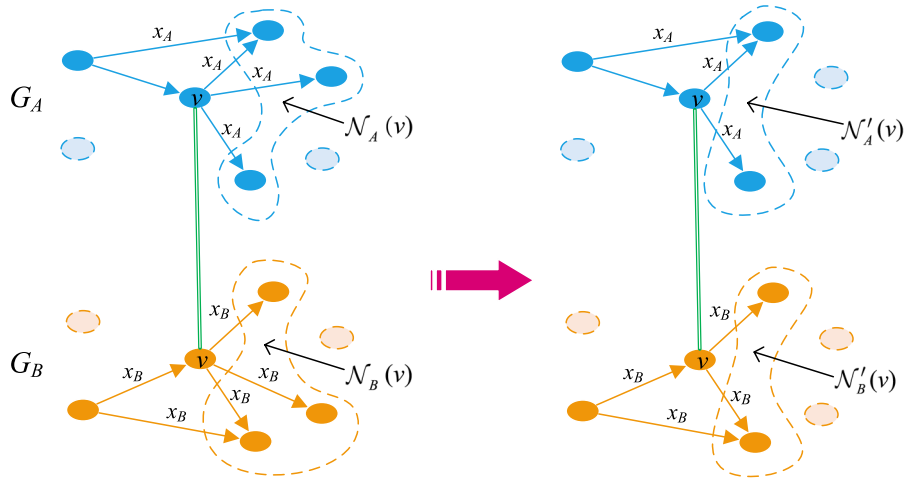


Fig. 3 The modification effect of cooperative coded forwarding on percolation graphs. Let $G_j = \langle V_j, E_j \rangle$ denote the connected forwarding graph of the j th SCP y_j , where V_j is the set of nodes that has received y_j , and E_j contains the links that have successfully delivered y_j . Consider two different forwarding graph G_a and G_b , and a common node $v \in V_a \cap V_b$. Denote the nodes covered by v in the two graphs as $\mathcal{N}_a(v) = \mathcal{N}(v) \cap V_a$ and $\mathcal{N}_b(v) = \mathcal{N}(v) \cap V_b$, respectively. Suppose v is going to encode y_a and y_b into a RCP $z_{a,b} = y_a \oplus y_b$ and forward it to its one-hop neighbors. A one-hop neighbor node $u \in \mathcal{N}(v)$ of v can receive $z_{a,b}$ with probability $1 - \rho$ due to the packet erasure effect. If u has already received y_a and y_b , i.e. $u \in \mathcal{N}_a(v) \cap \mathcal{N}_b(v)$, the receiving of $z_{a,b}$ become redundant. If u has only received y_a , the receiving of $z_{a,b}$ helps it to decode y_b , and the node u is added to V_b . The link $e(u, v)$ will also be involved in E_b . Similarly, the receiving of y_b and $z_{a,b}$ will add u into V_a . If u have received neither y_a nor y_b , the receiving of $z_{a,b}$ will not change G_a or G_b . The effect of this coded forwarding action on the broadcast performance can therefore be regarded as the modification of the two corresponding forwarding graph G_a and G_b

After decoding a number of c RCPs, the pdf of r is written as

$$\mathbf{w}^{(c)} = \mathbf{w}^{(0)} \cdot \mathbf{P}_{n \times n}^c \tag{7}$$

Therefore we obtain the expectation of r , which is given as

$$\bar{r}(n, c) \triangleq \mathbb{E}[r(n, c)] = \sum_{r=0}^n w^{(c)}(n, r) \cdot r \tag{8}$$

The number of RCPs generated at a node is related to the number of decoded SCPs r and the coded forwarding probability ξ . Whenever the nodes have decoded new SCPs from the received RCPs, it will continue to produce some new RCPs afterward. This iterative process can be asymptotically analyzed and finally approaches the relationship between ξ and the mean number of decoded SCPs. Define the mean number of RCPs generated at a node that has received r out of n SCPs as $C(r, \xi) = \binom{r}{2} \cdot \xi = \frac{\xi \cdot r \cdot (r-1)}{2}$. Let r_i denote the number of decoded SCPs after the i th iteration, and let c_i denote the number of RCPs participated in the decoding process in the i th iteration. Clearly, we have $r_0 = r$, and the expression of c_i is written as

$$c_i = \begin{cases} C(r_{i-1}, \xi), & i = 1 \\ C(r_{i-1}, \xi) - C(r_{i-2}, \xi), & i > 1 \end{cases} \tag{9}$$

When $c_i < 1$, i.e. $(r_{i-1} - 1)r_{i-1} - (r_{i-2} - 1)r_{i-2} < 2\xi^{-1}$, the value of r_i may not increase and so the iteration terminates.

3.3.2 Successfully decoding probability of data packets

Consider that each IIoT node receives an arbitrary SCP with the same successful receiving probability η_ξ . Denote the number of received systematic and random SCPs at a typical IIoT node as R_s and R_r , respectively. R_s is binomial random variable with parameters k and η_ξ , i.e. $R_s \sim \text{Bin}(k, \eta_\xi)$. Similarly, we have $R_r \sim \text{Bin}(n - k, \eta_\xi)$.

Let τ denote the number of decoded data packets from the randomly generated SCPs, thus the expectation of successful decoding probability p can be written as

$$\mathbb{E}[p] = \mathbb{E}\left[\frac{R_s + \tau}{k}\right] = \mathbb{E}\left[\frac{R_s}{k}\right] + E\left[\frac{\tau}{k}\right] = \eta_\xi + \frac{\mathbb{E}[\tau]}{k} \tag{10}$$

where the expectation of τ is derived as follows.

Suppose that an IIoT node has decoded $m < k$ data packets, which forms the collection of decoded data packets \mathcal{D} of size $|\mathcal{D}| = m$. Consider a randomly generated SCP of degree $d \in [1, m]$, it can be successfully decoded by this IIoT node only when one out of d data packets it carried have not been decoded. If all the data packets carried by this SCP have been already decoded, this SCP becomes redundant and should be ignored in future decoding process. The probability of these two conditions are written as

$$p_{\text{dec}}(m, d) = \frac{\binom{m}{d-1} \binom{k-m}{1}}{\binom{k}{d}} = \frac{m!(k-d)!(k-m)d}{(m-d+1)!k!} \tag{11}$$

$$p_{\text{ign}}(m, d) = \frac{\binom{m}{d}}{\binom{k}{d}} = \frac{m!(k-d)!}{k!(m-d)!} \tag{12}$$

Since the degree of a randomly generated SCP d is a binomial function of parameter k and $1 - p_0$, the expected successful decoding probability and ignored probability of an arbitrary random SCP are written as

$$\begin{aligned} p_{\text{dec}}(m) &= \sum_{d=1}^m \mathbb{P}(d_j = d) \cdot p_{\text{dec}}(m, d) \\ &= \sum_{d=1}^m \frac{m!(k-m)}{(d-1)!(m-d+1)!} p_0^{k-d} (1-p_0)^d \end{aligned} \tag{13}$$

$$\begin{aligned} p_{\text{ign}}(m) &= \sum_{d=1}^m \mathbb{P}(d_j = d) \cdot p_{\text{ign}}(m, d) \\ &= \sum_{d=1}^m \frac{m!}{d!(m-d)!} p_0^{k-d} (1-p_0)^d \end{aligned} \tag{14}$$

Let S_{m_s, m_r} denote the decoding state of an IIoT node, where m_s is the number of decoded data packets and m_r is the number of randomly generated SCPs that are not processed. An IIoT node at state S_{m_s, m_r} will try to decode the unprocessed random SCPs when $m_s < k$ and $m_r \geq 1$. If there is at least one decodable SCP, the node will switch to state S_{m_s+1, m_r-1} . Otherwise, the node will stay at the current state. Regarding the ignored packets, the transition probabilities can be written as

$$P(S_{m_s+1, m_r-1} | S_{m_s, m_r}) \triangleq 1 - \sum_{m_i=0}^{m_r} \mathbb{P}(m_i = m | S_{m_s, m_r}) \cdot (1 - p_{\text{dec}}(m_s))^{m_r - m} \tag{15}$$

$$P(S_{m_s, m_r} | S_{m_s, m_r}) \triangleq 1 - P(S_{m_s+1, m_r-1} | S_{m_s, m_r}) \tag{16}$$

where $\mathbb{P}(m_i = m | S_{m_s, m_r}) = \binom{m_r}{m_i} p_{\text{ign}}^{m_i}(m_s) (1 - p_{\text{ign}}(m_s))^{m_r - m_i}$, number m_i is a binomial stochastic variable of parameter m_r and $p_{\text{ign}}(m_s)$.

As R_s and R_r are two independent binomial random variables, an IIoT node will start its decoding process at state $S(R_s, R_r)$ with probability

$$P_{\text{start}}(S_{R_s, R_r}) = \mathbb{P}[m_s = R_s] \cdot \mathbb{P}[m_r = R_r] = \binom{k}{R_s} \binom{n-k}{R_r} \eta_{\xi}^{R_s+R_r} (1 - \eta_{\xi})^{n-R_s-R_r} \tag{17}$$

and continue decoding until either all the R_r random SCPs have been processed or the all the k data packets have been decoded. Thus the maximum number of τ should be $\tau_{\text{max}} = \min(k - R_s, R_r)$. The steady state probability of τ th state $S(R_s + \tau, R_r - \tau)$ can be written as

$$P(S_{R_s+\tau, R_r+\tau}) = \begin{cases} P(S_{R_s+\tau, R_r+\tau} | S_{R_s+\tau, R_r+\tau}), & \tau = 0 \\ \left(\prod_{t=0}^{\tau-1} P(S_{R_s+t+1, R_r+t+1} | S_{R_s+t, R_r+t}) \right) \cdot P(S_{R_s+\tau, R_r+\tau} | S_{R_s+\tau, R_r+\tau}), & \tau \in [1, \tau_{\text{max}}] \end{cases} \tag{18}$$

Finally, the expectation of τ can be written as

$$\mathbb{E}[\tau] = \sum_{R_s=0}^k \sum_{R_r=1}^{n-k} P_{\text{start}}(S_{R_s, R_r}) \cdot \sum_{\tau=0}^{\tau_{\text{max}}} \tau \cdot P(S_{R_s+\tau, R_r+\tau}) \tag{19}$$

Substitute Eq. (19) into Eq. (10) we can have the mean successful decode probability at an IIoT node of a data packet generation of size k .

3.3.3 Total transmission number

According to the design of PCCE, the total transmission number of a data packet generation is given as

$$T = N \cdot \left(n\eta \cdot \omega + \frac{n\eta_{\xi}(n\eta_{\xi} - 1)}{2} \cdot \xi \right) \tag{20}$$

which will increase as η and ξ increases.

4 The experiment

In this section, we perform analytical and simulation experiments to analyze the performance of the proposed PCCF scheme. The analytical and numerical simulation results are provided to show the validity of our approximations and the feasibility of PCCF.

4.1 Successfully receiving probability of SCPs

We first compare the simulation and approximated results of η under $N = 100$ and $\varphi \in [0, 10]$ as shown in Fig. 4. We can see from Fig. 4 that as the mean number of one-hop neighbors increases, the percentage of nodes that can receive a specific SCP also increases. The approximated results can predict the simulation result of η with small difference, which validates the accuracy of the approximation.

We also notice that the value of ϵ should be sufficiently small, as the value of φ_ϵ needs to be greater than the critical mean node degree a_c for continuum percolation [21], which is crucial for effective broadcast transmission. The range of such critical mean node degree is provided in [21] as $4.508 < a_c < 4.515$, and a more precise result is given in [24] as $a_c \approx 4.512$. From Fig. 4 we can see that after φ is greater than a_c , η will increase more slowly. To achieve a high value of η such as 0.8 or 0.9, the corresponding value of φ should be greater than 5.5 and 6.5, respectively.

Next we compare the simulation and analytical results of the relationships between c and $\bar{r}(n, c)$ under $n = 150$ and $\eta \in \{0.8, 0.85, 0.9, 0.95\}$ in Fig. 5, which shows that our analytical model can predict the number of decoded SCPs under different number of received RCPs with high accuracy.

Since the value of $\bar{r}(n, c)$ may not be an integer, we denote $r_i = \lceil \bar{r}(n, c_{i-1}) \rceil$ for the simplicity of notation, where $\lceil \cdot \rceil$ is the ceiling function. When $c_i < 1$, i.e. $(r_{i-1} - 1)r_{i-1} - (r_{i-2} - 1)r_{i-2} < 2\xi^{-1}$, the value of r_i may not increase and so the iteration terminates. An demonstration of the numerical analysis for the iterative decoding is presented in Fig. 6.

Finally, we can quantify the modification effect on the forwarding graphs based on the relationships between ξ and the successful receiving probability of SCPs $\eta_\xi = \bar{r}/n$ at a IIoT node, which is shown in Fig. 7. We observe that the larger the value of n is, the

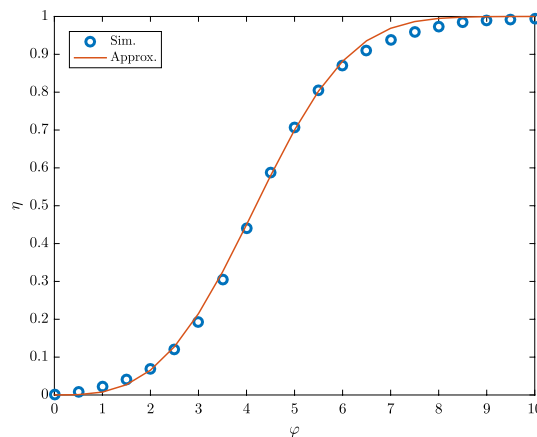


Fig. 4 Simulation and approximated results of $\eta(\varphi)$, $N = 100$. The figure shows the simulation and approximated results of $\eta(\varphi)$, $N = 100$ and validates the accuracy of the approximation

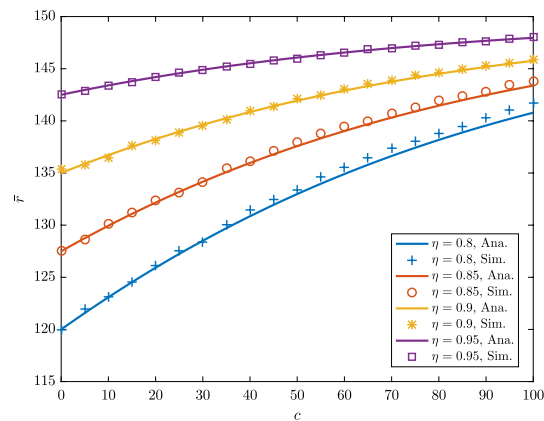


Fig. 5 Simulation and analytical results of the relationships between c and $\bar{r}(n, c)$. The figure shows simulation and analytical results of the relationships between c and $\bar{r}(n, c)$ under $n = 150$ and $\eta \in \{0.8, 0.85, 0.9, 0.95\}$

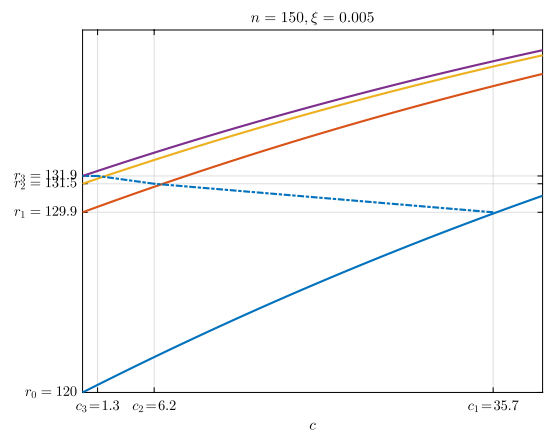


Fig. 6 Demonstration of the asymptotic analysis of r_i and c_i . The figure presents demonstration of the asymptotic analysis of r_i and c_i under $n = 150, \eta = 0.8$ and $\xi = 0.005$

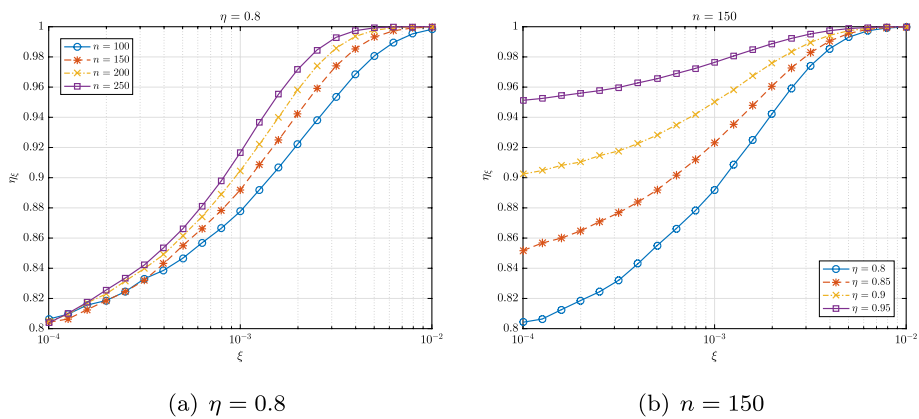


Fig. 7 Relationship between ξ and η_ξ . The figure presents the relationship between ξ and η_ξ : **a** $\eta = 0.8$ and $n \in \{100, 150, 200, 250\}$; **b** $n = 150$ and $\eta \in \{0.8, 0.85, 0.9, 0.95\}$. We can quantify the modification effect on the forwarding graphs through the relationships between ξ and the successful receiving probability of SCPs $\eta_\xi = \bar{r}/n$ at a IIoT node. We observe that the larger the value of n is, the faster η_ξ increases as ξ grows. Also the contribution of the increasing of ξ is much higher with a smaller η

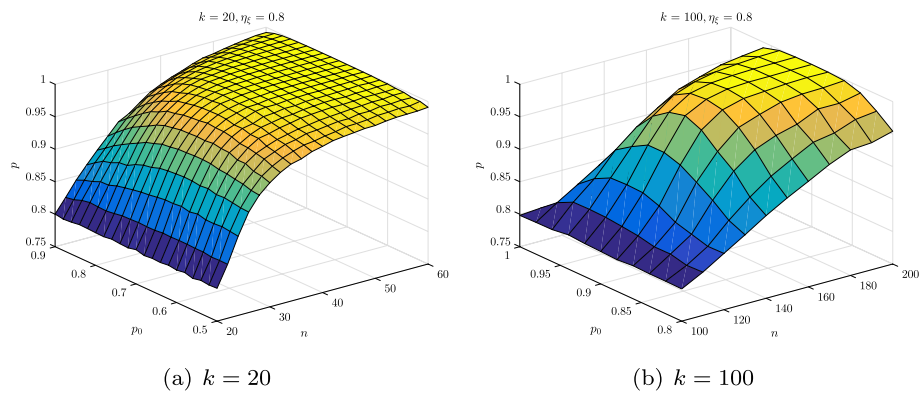


Fig. 8 Simulation results of successful decoding probability p under $\eta_\xi = 0.8$. The figure presents simulation results of successful decoding probability p under $\eta_\xi = 0.8$. We can observe that the optimal sparsity $p_{0,opt}$ for SSNC is approximately 0.7 when $k = 20$, and 0.9 when $k = 100$. This means that the optimal sparsity should be increased as the size of data packet generation increases

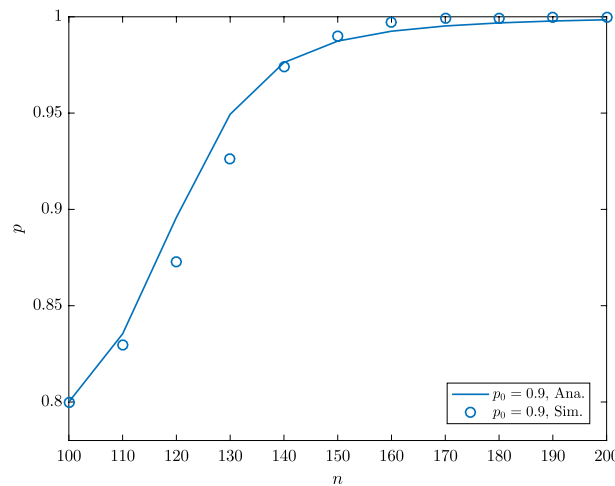


Fig. 9 Comparison of analytical and simulation results of successful decoding probability. The figure shows comparison of analytical and simulation results of successful decoding probability under $k = 100, \eta_\xi = 0.8$ and $p_0 = p_{0,opt} = 0.9$. It demonstrates the analytical and simulation results of p , which validates the accuracy of analytical model

faster η_ξ increases as ξ grows. Also the contribution of the increasing of ξ is much higher with a smaller η . The reason is that in with higher η , the nodes are able to decode more systematic SCPs before receiving RCPs, which decrease the opportunity of modifying forwarding graphs.

4.2 Successful decoding probability of data packets

The simulation results of successful decoding probability p are shown in Fig. 8, where we can observe that the optimal sparsity $p_{0,opt}$ for SSNC is approximately 0.7 when $k = 20$, and 0.9 when $k = 100$. This means that the optimal sparsity should be increased as the size of data packet generation increases.

Figure 9 demonstrates the analytical and simulation results of p under $k = 100$ and $p_0 = p_{0,opt} = 0.9$, which validates the accuracy of the analytical model.

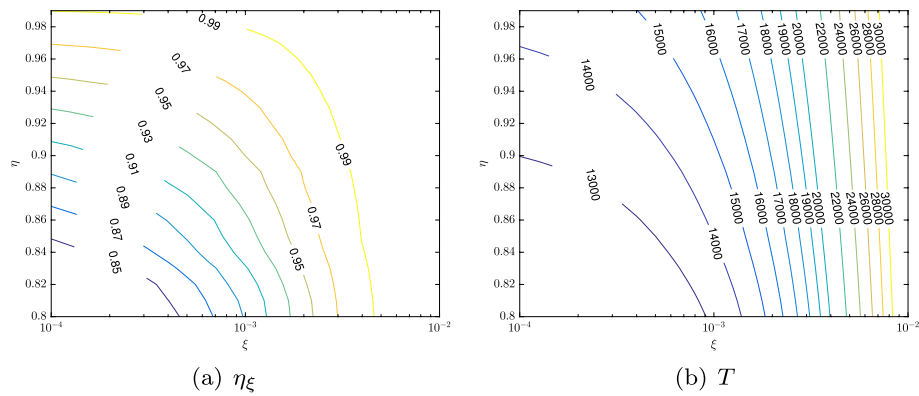


Fig. 10 Comparison of the contour maps of η_ξ and T under $n = 150$. The figure shows comparison of the contour maps of η_ξ and T under $n = 150$. We can see that in order to achieve the same value of η_ξ with different transmission number, which indicates the existence of the smallest transmission number and the corresponding optimal values of ω and ξ to fulfill the near-broadcast requirement of mobile edge networks

4.3 Transmission number

The results of the successful receiving probability η_ξ and the transmission number T are given in Fig. 10a, b, respectively. We can see that in order to achieve the same value of η_ξ , there exists a smallest transmission number and the corresponding optimal values of ω and ξ to fulfill the near-broadcast requirement of mobile edge networks. According to the above analysis, we can optimize ω and ξ under a required delivery ratio and maximum transmission number of mobile edge networks.

5 Conclusion

In this article, the network coding technique is adopted in the proposed PCCF scheme to realize efficient broadcasting in MEC-enabled IIoT networks. To minimize the transmission number at the data source, the systematic network coding framework is adopted and the sparsity of randomly generated coded packets is optimized by simulation experiments. To increase the efficiency of re-encoding process at the relay nodes, we model the impact of direct and coded forwarding probability on the receiving performance at the IIoT nodes, which gives fundamental results for further optimization. Simulation and approximation results carried out by experiments show that the proposed PCCF scheme can effectively minimize the required transmission number of both the source and relay nodes. Future researches include the design of cooperative coded transmission among multiple data sources and the performance analysis under hierarchical network settings.

Abbreviations

| | |
|------|--|
| MEC | Mobile edge computing |
| 5G | Fifth generation |
| IIoT | Industrial internet of things |
| UAV | Unmanned aerial vehicle |
| BS | Base station |
| D2D | Device-to-device |
| NC | Network coding |
| RLNC | Random linear network coding |
| GF | Galois field |
| PCCF | Probabilistic cooperative coded forwarding |
| SSNC | Systematic sparse network coding |
| CCF | Cooperative coded forwarding |
| SCP | Source-coded packet |
| RCP | Relay-coded packet |

Acknowledgements

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

Authors' contributions

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

Funding

This work is supported by the Fundamental Research Funds for the Central Universities under Grant 2021JBZ102.

Availability of data and materials

Data sharing not applicable to this article during the current study.

Declarations**Competing interests**

The authors declare that they have no competing interests.

Received: 4 January 2022 Accepted: 29 May 2022

Published online: 13 June 2022

References

- H. Gao, X. Qin, R.J.D. Barroso, W. Hussain, Y. Xu, Y. Yin, Collaborative learning-based industrial IoT API recommendation for software-defined devices: the implicit knowledge discovery perspective. *IEEE Trans. Emerg. Top. Comput. Intell.* **6**(1), 66–76 (2022)
- P. Corcoran, S.K. Datta, Mobile-edge computing and the internet of things for consumers: extending cloud computing and services to the edge of the network. *IEEE Consum. Electron. Mag.* **5**(4), 73–74 (2016). <https://doi.org/10.1109/MCE.2016.2590099>
- Y. Liu, M. Peng, G. Shou, Y. Chen, S. Chen, Toward edge intelligence: multiaccess edge computing for 5G and internet of things. *IEEE Internet Things J.* **7**(8), 6722–6747 (2020). <https://doi.org/10.1109/JIOT.2020.3004500>
- H. Gao, W. Huang, Y. Duan, The cloud-edge-based dynamic reconfiguration to service workflow for mobile e-commerce environments: a QoS prediction perspective. *ACM Trans. Internet Technol.* **21**(1), 1–23 (2021)
- H. Gao, Y. Zhang, H. Miao, R.J.D. Barroso, X. Yang, SDTIOA: modeling the timed privacy requirements of IoT service composition: a user interaction perspective for automatic transformation from BPEL to timed automata. *Mob. Netw. Appl.* **26**(6), 2272–2297 (2021)
- H. Gao, C. Liu, Y. Yin, Y. Xu, Y. Li, A hybrid approach to trust node assessment and management for VANETs cooperative data communication: historical interaction perspective. *IEEE Trans. Intell. Transp. Syst.* 1–10 (2021)
- H. Song, L. Liu, B. Shang, S. Pudlewski, E.S. Bentley, Enhanced flooding-based routing protocol for swarm UAV networks: random network coding meets clustering, in *IEEE INFOCOM 2021—IEEE Conference on Computer Communications* (IEEE, Vancouver) (2021). <https://doi.org/10.1109/INFOCOM42981.2021.9488721>
- N. Papanikos, E. Papapetrou, Deterministic broadcasting and random linear network coding in mobile ad hoc networks. *IEEE/ACM Trans. Netw.* **25**(3), 1540–1554 (2017). <https://doi.org/10.1109/TNET.2016.2641680>
- B.R.V. Kumar, N. Kashyap, Probabilistic forwarding of coded packets on networks. *IEEE/ACM Trans. Netw.* (2020). <https://doi.org/10.1109/TNET.2020.3031467>
- L. Shi, K. Cai, Z. Mei, Linear network coded computation in mobile edge computing, in *2019 IEEE Global Communications Conference (GLOBECOM)* (IEEE, Waikoloa, 2019), pp. 1–6. <https://doi.org/10.1109/GLOBECOM38437.2019.9014018>
- W. He, Y. Su, X. Xu, Z. Luo, L. Huang, X. Du, Cooperative content caching for mobile edge computing with network coding. *IEEE Access* **7**, 67695–67707 (2019). <https://doi.org/10.1109/ACCESS.2019.2917977>
- Y. Yin, Z. Cao, Y. Xu, H. Gao, R. Li, Z. Mai, QoS prediction for service recommendation with features learning in mobile edge computing environment. *IEEE Trans. Cogn. Commun. Netw.* **6**(4), 1136–1145 (2020)
- L. Li, R. Ramjee, M. Buddhikot, S. Miller, Network coding-based broadcast in mobile ad-hoc networks, in *IEEE INFOCOM 2007—26th IEEE International Conference on Computer Communications* (IEEE, Anchorage, 2007), pp. 1739–1747. <https://doi.org/10.1109/INFCOM.2007.203>
- W. Chen, F. Lu, Y. Dong, The rank distribution of sparse random linear network coding. *IEEE Access* **7**, 43806–43819 (2019). <https://doi.org/10.1109/ACCESS.2019.2907005>
- S. Katti, H. Rahul, H. Wenjun, D. Katabi, M. Medard, J. Crowcroft, XORs in the air: practical wireless network coding. *IEEE/ACM Trans. Netw.* **16**(3), 497–510 (2008). <https://doi.org/10.1109/TNET.2008.923722>
- G. Yue, K. Yang, S. Zhao, H.V. Poor, Design of network coding for wireless broadcast and multicast with optimal decoders. *IEEE Trans. Wirel. Commun.* **17**(10), 6944–6957 (2018). <https://doi.org/10.1109/TWC.2018.2864996>
- H. Sehat, P. Pahlavani, An analytical model for rank distribution in sparse network coding. *IEEE Commun. Lett.* **23**(4), 556–559 (2019). <https://doi.org/10.1109/LCOMM.2019.2896626>
- W. Chen, F. Lu, Y. Dong, Improved expression for rank distribution of sparse random linear network coding. *IEEE Commun. Lett.* **25**(5), 1472–1476 (2021). <https://doi.org/10.1109/LCOMM.2020.3041845>
- B. Shrader, N.M. Jones, Systematic wireless network coding, in *MILCOM 2009—2009 IEEE Military Communications Conference* (IEEE, Boston, 2009), pp. 1–7. <https://doi.org/10.1109/MILCOM.2009.5380081>

20. A.L. Jones, I. Chatzigeorgiou, A. Tassi, Binary systematic network coding for progressive packet decoding, in *2015 IEEE International Conference on Communications (ICC)* (IEEE, London, 2015), pp. 4499–4504. <https://doi.org/10.1109/ICC.2015.7249031>
21. M. Haenggi, *Stochastic Geometry for Wireless Networks* (Cambridge University Press, Cambridge, 2012)
22. M.D. Penrose, Onk-connectivity for a geometric random graph. *Random Struct. Algorithms* **15**(2), 145–164 (1999). [https://doi.org/10.1002/\(SICI\)1098-2418\(199909\)15:2<145::AID-RSA2>3.0.CO;2-G](https://doi.org/10.1002/(SICI)1098-2418(199909)15:2<145::AID-RSA2>3.0.CO;2-G)
23. W. Gang, Z. XiaoRong, C. HaiTao, L. LiZhi, Research on the probabilistic broadcasting algorithms of mobile ad hoc network based on percolation theory, in *2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)* (IEEE, Jilin, 2011), pp. 330–333. <https://doi.org/10.1109/MEC.2011.6025468>
24. X. Gu, H. Feng, Connectivity analysis for a wireless sensor network based on percolation theory, in *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, vol. 5 (IEEE, Taiyuan, 2010). <https://doi.org/10.1109/ICCASM.2010.5619163>
25. V. Raman, I. Gupta, Performance tradeoffs among percolation-based broadcast protocols in wireless sensor networks, in *2009 29th IEEE International Conference on Distributed Computing Systems Workshops* (IEEE, Montreal, 2009), pp. 158–165. <https://doi.org/10.1109/ICDCSW.2009.76>

Publisher's Note

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

Submit your manuscript to a SpringerOpen[®] 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)
