Reliability of All-to-all Broadcast with Network Coding

Li Ma*, Zihuai Lin*, Zijie Zhang†, Guoqiang Mao* and Branka Vucetic*
*School of Electrical and Information Engineering, University of Sydney, Australia
†National ICT Australia (NICTA), Australia
Email: {li.m,zihuai.lin,guoqiang.mao,branka.vucetic}@sydney.edu.au , zijie.zhang@nicta.com.au

***Abstract*—Wireless communication is notoriously lossy due to channel fading, interference and multi-path effects. This work investigates the reliability of all-to-all broadcast in lossy wireless networks where the *reliability* is measured by the probability that every node in the network receives or decodes the native packet of every other node. To improve the reliability, a novel network coding scheme, namely *random neighbour network coding* (RNNC) scheme is proposed, which is capable of adaptively generating encoding packets according to the packets received from lossy wireless channels. The network reliability is analysed theoretically and the optimal RNNC scheme that maximises the reliability of a given network is obtained. The theoretical analysis is validated using simulations and it is shown that RNNC can improve the network reliability significantly.**

***Index Terms*—Network coding, All-to-all, Broadcast, Reliability**

I. Introduction

All-to-all broadcast, where every node has a packet to transmit to every other node, is a key mechanism in wireless communications. Due to the lossy nature of wireless communications, the packet transmitted from a source node may not be able to reach its destinations in one transmission. Therefore, multiple transmissions may be required to achieve a desired reliability, where *reliability* is defined as the probability that every node receives or decodes the native packets of every other node in the network.

On the other hand, the lossy and broadcast nature of wireless communications makes network coding [1] suitable for all-to-all broadcasting. There is some research in the literature studying the expected number of transmissions for information reaching its intended destinations in the one-to-all broadcast; but reliability after each transmission coupled with the all-to-all broadcast is less understood.

In this paper, we propose a novel random neighbour network coding scheme for all-to-all broadcast in lossy wireless networks. Specifically, the major contributions of this work are highlighted as follows.

- A *random neighbor network coding* is proposed where each node randomly chooses 1) whether or not to perform coding according to a tuning parameter, which is described in detail in Section III; and 2) with which packet to perform coding on-the-fly according to the packets that it has received and decoded.
- Theoretical analysis is conducted to characterise the reliability of networks employing the proposed RNNC scheme.
- The tuning parameter that maximises the reliability of all-to-all broadcast in a network with given link quality is derived. It is concluded that it may not provide optimal results for a node to take every opportunity to perform coding.
- It is shown that network reliability is improved significantly by utilising the proposed RNNC scheme. In addition, the reliability performance of RNNC is compared with random linear network coding under $GF(2)$.

The rest of the paper is organised as follows. Section II reviews related work. Section III introduces the system model. Theoretical analysis and optimisation techniques are given in Sections IV and V respectively. Then, the numerical results are given in Section VI. Finally, Section VII concludes the paper and proposes future work.

II. Related work

Network coding has been widely used in lossy networks to reduce the number of retransmissions while maintaining a certain reliability in the literature [2]–[6].

Packet retransmission scheme based on the use of network coding for wireless broadcast is proposed in [3]. An XOR coding is employed there to combine the lost packets for different receivers. Then at each individual receiver, the lost packets can be recovered with the knowledge of previously received packets. In this case, multiple receivers may recover their lost packets in one transmission. Therefore the total number of retransmissions can be reduced. In this scheme, the source node requires the knowledge of the lost packets from each receiver. However, in broadcast scenario, feedback is expensive and inefficient in terms of energy and bandwidth consumption. In this paper, we focus on the study of network coding based broadcast scheme without feedback.

In [4], the authors consider a two stage broadcast scheme where every node broadcasts native packets in the first stage and XOR coded packets are transmitted in the second stage. They provide the optimal numbers of packets to be XORed in the second stage to minimise the expected number of transmissions with given success probabilities of every link. However, a network applying this coding method does not always outperform the corresponding non-coded network. In comparison, a network applying RNNC can achieve *at least* the same performance as that of the corresponding non-coded network.

In [5], the reliability gain is characterised analytically, where network coding is compared with traditional error control protocols, such as ARQ and FEC. The considered systems have tree topologies where each multicast tree has equal number of children. The expected numbers of transmissions by the source node under different error control protocols are computed. It is concluded that the reliability gain made by network coding increases logarithmically with respect to the number of receivers in a multicast group compared with a simple ARQ scheme [5].

Ref. [6] studies networks with up to four nodes where random linear network coding is applied at the source node. The delay distribution, i.e. the probability of successful decoding at individual delay, considered in [6], is the same as the reliability distribution considered in this paper. The broadcast message is determined prior to the first broadcast in [6]. In contrast, in this work the broadcast message is determined adaptively at a source node depending on its received packets, which varies over time. Additionally, the algorithms developed in this work are applicable to a network with an arbitrary number of nodes.

III. System model

The model of interest is all-to-all broadcast in a network with n nodes. It is assumed that time is slotted. In each time slot, one source node broadcasts a single packet to all other nodes in the network while every other node listens. All nodes in the network broadcast in a round-robin manner and a successful transmission is not acknowledged. A *round*, denoted by R, is defined as a sequence of time slots during which every node broadcasts once. At the beginning (time 0), each node has one packet to broadcast. Define the packet that a node N_k ($k \in \{1, 2, \ldots, n\}$) has at time zero as its native packet, which is denoted by X_k.

Further, it is assumed the probability that a packet transmitted from N_j reaches N_i successfully in one time slot is p_{ji}, where $p_{ji} \in (0, 1]$. Then, the p_{ji}s ($i, j \in \{1, 2, \ldots, n\}$) for every pair of node is written in a matrix form, known as the probabilistic connectivity matrix [7]. Since this work focuses on the impact of network coding on the reliability, it is assumed that probabilistic connectivity matrix is known.

Due to the lossy nature of wireless communications, the packet broadcast from a source node may not be able to reach all destinations in one time slot. Therefore, retransmission is required. Conventionally, a source node only re-broadcasts its native packet. With the use of network coding, a source node may broadcast a coded packet.

A large number of existing network coding schemes are *fixed* network coding schemes, where network coding is performed on all or certain predetermined packets [5], [6]. Since practical wireless networks contain lossy channels, the packets that a node received are random and the predetermined packets may not be available, hence *random* network coding schemes receive increasingly interest recently [4].

The network coding scheme considered in this work is *random neighbour network coding* (RNNC) scheme. The following describes in detail the encoding and decoding rules of the RNNC scheme.

Encoding

Denote the collection of native packets that N_j possesses as $\mathcal{D}_j$. It includes the native packet of itself, the native packets of other nodes received directly and the native packets decoded from coded packets. If $\mathcal{D}_j = \{X_j\}$, then N_j broadcasts X_j. If $\mathcal{D}_j \setminus \{X_j\} \neq \emptyset$, then N_j does not employ network coding and broadcasts the native packet with probability $1 - \omega$; and with probability ω, N_j randomly selects a packet from $\mathcal{D}_j \setminus \{X_j\}$ with equal probabilities and performs bitwise XOR between the selected native packet and its native packet X_j.

The parameter ω is introduced to overcome the cases when network coding may have negative impact on reliability. For example, when a node receives a number of XORed packets, it may not able to decode them due to lack of native packets, as described in [4]. Therefore, ω is used to allow a node to choose to broadcast its native packet with a certain probability even if there are other packets available for coding. For example, if $\mathcal{D}_2 = \{X_1, X_2, X_3\}$, then N_2 broadcasts X_2, $X_1 \oplus X_2$ and $X_2 \oplus X_3$ with probabilities $1-\omega$, $\omega/2$ and $\omega/2$ respectively. Moreover, the optimal value of ω that maximises the reliability will be studied in Section V.

Decoding

Using the proposed coding algorithm, the successful decoding of a coded packet only requires that one of the two native packets forming the coded packet has already been successfully received or decoded. For example, X_z can be decoded from packets $X_z \oplus X_k$ and X_k by performing $(X_z \oplus X_k) \oplus X_k$, where $z, k \in \{1, 2..., n\}$ and $y \neq k$. It is worth to note that X_k can either be received directly or be decoded from another coded packet. After X_z and X_k are decoded, they can be used to decode other coded packets. The decoding process continues until no more XORed packets can be decoded.

A *buffer* is used at each node to store the packets including its native packet, the packets it received from other nodes and the decoded packets. Duplicate packets are dropped. Moreover, if every native packet forming a coded packet already exists in the buffer, the coded packet is dropped. In this way, the demand on the size of the buffer can be minimised.

IV. Theoretical analysis

In each time slot, a source node broadcasts a randomly generated packet. Furthermore, the packet may be received by different destination nodes with different probabilities. Therefore, it is challenging to track the packets that each node receives and stores after each transmission.

Denote the set of packets that N_k has at the end of the time slot t as $\boldsymbol{v_k}$(t). Further, denote the packets stored at all nodes in the network at the end of time slot t as $\boldsymbol{V}(t) = [\boldsymbol{v_1}(t); \boldsymbol{v_2}(t); \ldots \boldsymbol{v_n}(t)]$. Let random process $\mathcal{Z}(t)$ represent the packets that are individually stored at every node in the network at the end of time slot t, which includes the packets received and stored in all previous time slots, from 0 to t.

Using the definition of $\mathcal{Z}(t)$, it can be shown that:

$$\Pr\Big(\mathcal{Z}(t+1)=\boldsymbol{V}(t+1)\Big|\mathcal{Z}(1)=\boldsymbol{V}(1),\ldots,\mathcal{Z}(t)=\boldsymbol{V}(t)\Big)$$
$$=\Pr\Big(\mathcal{Z}(t+1)=\boldsymbol{V}(t+1)\Big|\mathcal{Z}(t)=\boldsymbol{V}(t)\Big). \quad (1)$$

It is clear that this random process is memoryless and the packets stored in buffers of all nodes in time slot $t+1$ depends only on the packets in time slot t, but not on those before time slot t. Therefore, the random process $\mathcal{Z}(t)$ can be modelled by a Markov chain.

The method to establish and reduce states of the network is introduced in Subsection IV-A, followed by the calculation of the transition matrices in Subsection IV-B. Finally, the reliability of a network is given in Subsection IV-C.

A. States

Denoted by matrix $\boldsymbol{V_a}$ the a^{th} *state of a network*, where the state means the status of stored packets at all nodes, and $a \in \{1, 2, \ldots, L\}$. L is the total number of states, which will be discussed in Remark 1. Further, let the k^{th} row of $\boldsymbol{V_a}$, denote by $\boldsymbol{v_{ka}}$, be the packets that node N_k has when the network is in state $\boldsymbol{V_a}$.

There are two categories of packets in the network. The first category is the native packet of each source node, while the second category is the XORed packet of a pair of native packets. Each packet is assigned with a unique index. Specifically, the native packet X_k is assigned with index k; and an XORed packet, say $X_z \oplus X_k$ (it is assumed $1 \le z < k \le n$ without losing generality) is assigned with index $\mu_{z,k} \triangleq nz - z^2/2 - z/2 + k$. Then, the total number of distinct packets is $n + \binom{n}{2} = (n^2+n)/2$. Consequently, $\boldsymbol{v_{ka}}$ can be represented by a row vector which is composed of $(n^2+n)/2$ elements where each element represents a packet. More specifically, the first n elements of $\boldsymbol{v_{ka}}$ represent native packets of n nodes respectively, and the following elements represent XORed packets. In node N_k, the possession of a packet is represented by assigning the corresponding element in $\boldsymbol{v_{ka}}$ to one; otherwise, it is set to zero. For example, in a network with three nodes, if node N_1 has packets X_1 and $X_2 \oplus X_3$, then $\boldsymbol{v_{1a}} = [1, 0, 0, 0, 0, 1]$.

Absorbing state: There exists an absorbing state representing the event that all nodes in the network have successfully received or decoded native packets from every other node. When this occurs, the status of encoded packets no longer matters. This state is represented by $\boldsymbol{V_L}$, in which the first n elements in every row are one and every other element is zero, as is shown in Fig. 1.

States reduction: To reduce the complexity of analysis, two methods are introduced to reduce the number of states. Firstly, the number of states can be reduced by taking decoding process into account, i.e., the states including the XORed packets whose corresponding native packets have already been received or decoded, are merged.

Secondly, the number of states can be further reduced by discarding invalid states. The invalid states are the states that can never be entered. For example, when neither N_z nor N_k has both X_z and X_k, it is not possible for a third node N_θ where $\theta \in \{1, 2, \ldots, n\} \setminus \{z, k\}$ to have encoded $X_z \oplus X_k$. Therefore the associated state is invalid.

Remark 1. Before reduction, the total number of states for a network with n nodes is given by $2^{(n^3+n^2-2n)/2}$, where the base 2 represents the procession or not of a packet at a node.

The state reduction can significantly reduce the number of states. For example, after states reduction, the total number of states for networks with three and four nodes are 103 and 30519 respectively (outputs from Matlab when states are constructed by a program following the simple rules described above), compared with 32768 and 68719476736 respectively before state reduction.

B. Transition matrices

At the end of each time slot, the packet broadcast from a source node may be received by destination nodes with different probabilities resulting in update of status of packets stored in some nodes. This can be reflected by a transition in states of the Markov chain. In illustration of this, Fig. 1 shows some states in the Markov chain.

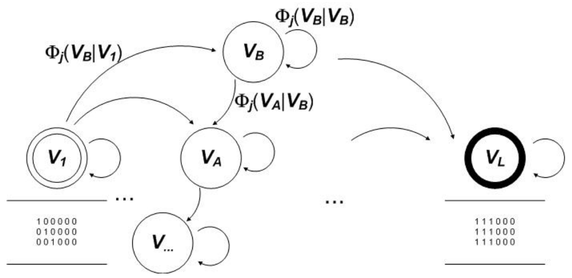

Fig. 1. An illustration of the Markov chain of a network when $n = 3$, where $\boldsymbol{V_1}$ represents initial state, $\boldsymbol{V_L}$ represents the absorbing state and $\Phi_j(\boldsymbol{V_A}|\boldsymbol{V_B})$ is the transition probability that the state of the network change from state $\boldsymbol{V_B}$ to state $\boldsymbol{V_A}$ when N_j broadcasts a packet.

Denote $\mathcal{M}_j$ as the transition matrix when N_j broadcasts. It is an $L \times L$ matrix, defined as:

$$\mathcal{M}_j = \begin{bmatrix} \Phi_j(\boldsymbol{V_1}|\boldsymbol{V_1}) & \cdots & \Phi_j(\boldsymbol{V_L}|\boldsymbol{V_1}) \\ \vdots & \ddots & \vdots \\ \Phi_j(\boldsymbol{V_1}|\boldsymbol{V_L}) & \cdots & \Phi_j(\boldsymbol{V_L}|\boldsymbol{V_L}) \end{bmatrix} \quad (2)$$

where the entry $\Phi_j(\boldsymbol{V_A}|\boldsymbol{V_B})$ is the probability that the state of the network changes from $\boldsymbol{V_B}$ to $\boldsymbol{V_A}$ in the next time slot. It can be calculated by the following procedures.

The first step is to compare $\boldsymbol{v_{jA}}$ and $\boldsymbol{v_{jB}}$, which are the packets stored in the transmitting node N_i. If $\boldsymbol{v_{jA}} \neq \boldsymbol{v_{jB}}$, then the transition probability $\Phi_j(\boldsymbol{V_A}|\boldsymbol{V_B}) = 0$. This is because, the transmitting node N_j only broadcasts packets it already has and will not lead to a variation of its own packets.

Next, we consider the case that $\boldsymbol{v_{jA}} = \boldsymbol{v_{jB}}$. Recall that the native packets that N_j has are $\mathcal{D}_j$. Further, denote the cardinality of $\mathcal{D}_j$ as m_j. Then the source node N_j may broadcast m_j different types of packets, among which $m_j - 1$ are XORed packets and one is native packet. Denote π_{jh}, where $h \in \mathcal{D}_j \setminus \{j\}$ as the event that N_j broadcasts an XOR coded packet $X_j \oplus X_h$; and denote π_{jj}, as the event that N_j broadcasts its native packet X_j. There are:

$$\Pr(\pi_{jh}) = \frac{\omega}{m_j - 1},$$
$$\Pr(\pi_{jj}) = 1 - \omega. \quad (3)$$

Further, denote $\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jj})$ and $\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jh})$ as the conditional probabilities that the state of the network transforms from $\boldsymbol{V_B}$ to $\boldsymbol{V_A}$ conditioned on the events that the source node broadcasts packets X_j and $X_j \oplus X_h$ respectively. Then according to the total probability theory, there is:

$$\Phi_j(\boldsymbol{V_A}|\boldsymbol{V_B}) = \Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jj})\Pr(\pi_{jj}) + \sum_{h\in\mathcal{D}_j\setminus\{j\}} \Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jh})\Pr(\pi_{jh})$$
$$= \Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jj})(1-\omega) + \sum_{h\in\mathcal{D}_j\setminus\{j\}} \Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jh})\frac{\omega}{m_j-1}. \quad (4)$$

The conditional probability $\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jh})$, is the product of n probabilities of mutually independent events, denoted by $P^h_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$, i.e. the probabilities that the status of packets in the buffer of node N_i changes from $\boldsymbol{v_{iB}}$ to $\boldsymbol{v_{iA}}$ for all $i \in \{1,2\ldots,n\}$ when node N_j transmits $X_j \oplus X_h$. Similarly, the conditional probability $\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jj})$ is the product of n probabilities $P^h_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$, where $P^h_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$ is the probability of N_i changing from $\boldsymbol{v_{iB}}$ to $\boldsymbol{v_{iA}}$ when N_j broadcasts X_j.

Then, the probabilities $\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jj})$ and $\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jh})$ can be calculated by:

$$\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jj}) = \prod_{i\in\{1,2,\ldots,n\}} P^j_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}}),$$
$$\Pr(\boldsymbol{V_A}|\boldsymbol{V_B},\pi_{jh}) = \prod_{i\in\{1,2,\ldots,n\}} P^h_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}}). \quad (5)$$

The probabilities $P^j_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$ and $P^h_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$ depend on whether or not the reception of the packet from N_j leads to the status of packets stored in N_i change from $\boldsymbol{v_{iB}}$ to $\boldsymbol{v_{iA}}$. These probabilities can be either value from the set $\{0,1,p_{ji},1-p_{ji}\}$, and can be obtained by comparing $\boldsymbol{v_{iB}}$ and $\boldsymbol{v_{iA}}$, which will be discussed in details in Algorithm 1 for obtaining $P^j_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$ and Algorithm 2 for $P^h_{ji}(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}})$ respectively.

In the algorithms, denote by $\boldsymbol{v_{iA}}\{\lambda\}$ (resp. $\boldsymbol{v_{iB}}\{\lambda\}$) the λ^{th} element of $\boldsymbol{v_{iA}}$ (resp. $\boldsymbol{v_{iB}}$), and we say that $\boldsymbol{v_{iA}} = \boldsymbol{v_{iB}}$ if $\boldsymbol{v_{iA}}\{\lambda\} = \boldsymbol{v_{iB}}\{\lambda\}$ for all $\lambda \in \{1,2,\ldots,(n^2+n)/2\}$.

Further, In Algorithm 1 denote by $\mathcal{H}_{ji}$ the collection of indices of the native packets that node N_i is able to decode upon receiving packet X_j when the packets stored in N_i is $\boldsymbol{v_{iB}}$. The collection $\mathcal{H}_{ji}$ can be obtained recursively by adding in the corresponding indices of native packets (e.g., γ and k) of an XORed packet $(X_\gamma \oplus X_k)$, if the following two conditions are both satisfied: 1) N_i has the XORed packet $(X_\gamma \oplus X_k)$; and 2) $\mathcal{H}_{ji}$ has the index of one of the native packets (either γ or k). Denote by $\mathcal{G}_{ji}$ the collection of every index of the XORed packet that can be decoded at node N_i when the packet X_j is received.

Similarly, in Algorithm 2, denote by $\mathcal{H}_{jhi}$ the collection of indices of the native packets that node N_i is able to decode upon receiving packet $X_j \oplus X_h$ when the packets stored in N_i is $\boldsymbol{v_{iB}}$. It can be obtained from a recursive method similar to that for obtaining $\mathcal{H}_{ji}$. The iteration begins with $\mathcal{H}_{ji} = \{j,h\}$ when N_i either has X_j or X_h, i.e., the packet $X_j \oplus X_h$ can be decoded at N_i. Moreover, $\mathcal{H}_{ji} = \emptyset$ when N_i has neither X_j nor X_h, viz. the packet $X_j \oplus X_h$ cannot be decoded at N_i. Denote by $\mathcal{G}_{jhi}$ the collection of indices of all XORed packets that can be decoded at node N_i upon receiving packet $X_j \oplus X_h$.

Algorithm 1 when N_j transmits X_j to N_i

```
if v_iB {j} = 1 then
    if v_iB = v_iA then P^j_ji(v_iA | v_iB) = 1;
    else P^j_ji(v_iA | v_iB) = 0 ;
    end if
else
    if v_iB {μ_{j,λ}} = 1 for any λ ∈ N \ {i, j} then
        if v_iA{x} = 1 and v_iA{y} = 0 for all x ∈ H_ji, y ∈ G_ji;
and v_iB{λ} = v_iA{λ} for all λ ∈ {1, 2, ... (n^2+n)/2}\{H_ji, G_ji}
then P^j_ji(v_iA | v_iB) = p_ji ;
        else if v_iB = v_iA then P^j_ji(v_iA | v_iB) = 1 − p_ji;
        else P^j_ji(v_iA | v_iB) = 0;
        end if
    else
        if v_iA{j} = 1 and v_iB{λ} = v_iA{λ} for all λ ∈
{1, 2, ..., (n^2 + n)/2} \ {i} then P^j_ji(v_iA | v_iB) = p_ji ;
        else if v_iB = v_iA then P^j_ji(v_iA | v_iB) = 1 − p_ji ;
        else P^j_ji(v_iA | v_iB) = 0 .
        end if
    end if
end if
```

Algorithm 2 when N_j transmit $X_j \oplus X_h$ to N_i for all $h \in \mathcal{D}_i$

```
if v_iB {μ_{j,h}} = 1 or ( v_iB {j} = 1 and v_iB {h} = 1) then
    if v_iB = v_iA then P^h_ji(v_iA | v_iB) = 1;
    else P^h_ji(v_iA | v_iB) = 0 ;
    end if
else if v_iB{j} = 0 and v_iB {h} = 0 then
    if v_iA{μ_{j,h}} = 1 and v_iB {λ} = v_iA{λ}, for all λ ∈
{1, 2, ..., (n^2 + n)/2} \ {μ_{j,λ}} then P^h_ji(v_iA | v_iB) = p_ji;
    else if v_iB = v_iA then P^h_ji(v_iA | v_iB) = 1 − p_ji;
    else P^h_ji(v_iA | v_iB) = 0;
    end if
else
    if v_iA{x} = 1 and v_iA{y} = 0 for all x ∈ H_jhi, y ∈
G_jhi; and v_iB {λ} = v_iA{λ} for all λ ∈ {1, 2, ..., (n^2 + n)/2} \
{H_jhi, G_jhi} then P^h_ji(v_iA | v_iB) = p_ji;
    else if v_iB = v_iA then P^h_ji(v_iA | v_iB) = 1 − p_ji;
    else P^h_ji(v_iA | v_iB) = 0;
    end if
end if
```

Remark 2 (Explanation of Algorithms). In Algorithm 1, if the node N_i already has packet X_j (viz. $\boldsymbol{v_{iB}}\{j\} = 1$), the status of packets in the buffer of node N_i does not change regardless of whether or not N_i receives X_j from N_j in this transmission.

In the case that N_i does not have X_j but has $X_j \oplus X_k$, the reception of X_j leads to every native packets in set $\mathcal{H}_{ji}$ decoded from XOR coded packets whose indices are included in $\mathcal{G}_{ji}$. Recall that in $\boldsymbol{v_{ka}}$, the possession of a packet is represented by setting its index to 1, and 0 otherwise. Additionally, the indices of XORed packets that are decoded are set to 0. Therefore, in the variation of status from $\boldsymbol{v_{iB}}$ to

$\boldsymbol{v_{iA}}$, if x^{th} bits for all $x \in \mathcal{H}_{ji}$ changes from 0 to 1 and y^{th} bits for all $y \in \mathcal{G}_{ji}$ changes from 1 to 0 while other bits stay the same, then X_j reaches N_i successfully and the corresponding probability is equal to p_{ji}. On the other hand, if $\boldsymbol{v_{iB}} = \boldsymbol{v_{iA}}$, then no packet is received by N_i in this time slot and the corresponding probability is $1 - p_{ji}$.

Lastly, in the case that N_i does not have X_j nor any coded packet consisting X_j, the reception of X_j only changes the j^{th} element of $\boldsymbol{v_{iB}}$ from 0 to 1. That is because no packet can be decoded upon receiving packet X_j. If $\boldsymbol{v_{iB}}\{j\}$=0 and $\boldsymbol{v_{iA}}\{j\}$=1, then the packet X_j reaches N_i successfully in this time slot, which happens with probability p_{ji}. On the other hand, the probability for the status of packets stored in N_i to stay the same is equal to $1 - p_{ji}$.

Algorithm 2 applies the similar rules.

After obtaining every $P^j_{ji}\big(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}}\big)$ and $P^h_{ji}\big(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}}\big)$. The entry $\Phi_j(\boldsymbol{V_A}|\boldsymbol{V_B})$ is readily obtained, which is:

$$\begin{aligned}\Phi_j(\boldsymbol{V_A}\big|\boldsymbol{V_B}) &= \prod_{i\in\{1,2,\ldots,n\}} P^j_{ji}\big(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}}\big)(1-\omega) \\ &+ \sum_{h\in\mathcal{D}_j\backslash\{j\}} \prod_{i\in\{1,2,\ldots,n\}} P^h_{ji}\big(\boldsymbol{v_{iA}}|\boldsymbol{v_{iB}}\big)\frac{\omega}{m_j-1}. \quad (6)\end{aligned}$$

By applying the same method, the transition matrix $\mathcal{M}_j$ is obtained. Similarly, the transition matrices $\mathcal{M}_j$ for every $j \in \{1, 2, \ldots, n\}$ can be calculated. Then, we define $\mathcal{M} \triangleq \prod_{j=1}^{n} \mathcal{M}_j$ as the transition matrix for a round.

C. Probability vector and the reliability

The *probability vector* indicates the probabilities of the network at every possible state. It is a row vector of size $1\times L$, where the l^{th} element is the probability that the network is at state $\boldsymbol{V_l}$. Denote the probability vector at the end of each round R as $S(R)$. In the initial probability vector, denoted by $S(0)$, the element of the initial state is of probability one and all other elements are zero, where the *initial state* is that every node only has its native packet, denoted by $\boldsymbol{V_1}$. According to Markov theory, the probability vector at the end of round R is equal to:

$$\begin{aligned}S(R) &= S(R-1)\prod_{j=1}^{n}\mathcal{M}_j \\ &= S(0)\mathcal{M}^R. \quad (7)\end{aligned}$$

Finally, we calculate the reliability at the end of round R, denoted by $\psi(R)$. It is the probability that the network falls into the absorbing state by the end of round R.

$$\psi(R) = S(R)\{L\}, \quad (8)$$

where the absorbing state is indicated by the L^{th} bit in the probability vector.

V. Optimisation

In order to maximise the reliability in a given network employing the RNNC scheme, the tuning parameter ω is optimised in this section. The value range of ω is $[0, 1]$, where 0 represents the case that a node does not perform coding and 1 represents the case that a node always performs coding. Therefore, this parameter determines the impact of coding on reliability. By tuning ω, we are tuning the reliability of a network ranging from non-coding to certainly coding.

A. Optimise the reliability at an individual round

The network reliability $\psi(R)$ at round R, given in eq. (8), is a function of variables ω, p_{ji} and $1 - p_{ji}$. In the case that the probabilistic connectivity matrix is given, the expression $\psi(R)$ can be reduced to a single-variable polynomial by substituting the values of the given probabilistic connectivity matrix. Then, the optimisation becomes a constrained nonlinear optimisation problem. Detailed methods will be given in Section VI.

B. Optimise the expected round to absorb

Recall that there is one absorbing state in the Markov chain. Rearrange the transition matrix into the corresponding canonical form [8]:

$$\mathcal{M} = \begin{pmatrix} Q & Y \\ 0 & 1 \end{pmatrix}, \quad (9)$$

where Q is the transitions among all transient states, and Y is the transitions from transient states to the absorbing state. Then, the fundamental matrix is $N = (I - Q)^{-1}$, where the $(\mathrm{A}, \mathrm{B})^{\mathrm{th}}$ entry describes the expected number of rounds to reach a transient state B starting from a transient state A. Finally, the expected number of rounds to reach the *absorbing state*, i.e., all nodes receives or decodes the native packets of all other nodes, can be calculated by $E = N\boldsymbol{c}$, where $\boldsymbol{c}$ is a column vector of size $L \times 1$, whose entries are all ones. Therefore, the expected number of rounds to absorbing state $\boldsymbol{V_L}$ (assume it is the last state) from initial state $\boldsymbol{V_1}$ (assume it is the first state) is the entry $E\{1\} \triangleq E_{exp}$, which is a function of ω, p_{ji} and $1 - p_{ji}$. Substitute the values of the given probabilistic connectivity matrix, then E_{exp} becomes a single variable polynomial of ω. Finally, the minimum E_{exp} and the corresponding ω can readily be found by solving a constrained nonlinear optimisation problem.

VI. Numerical results

This section provides numerical evaluation of the analytical expressions of reliability derived in the previous sections. The analytical results are validated using simulations, which are conducted by Matlab. Further, the examples of the optimisation are given. Then the reliability gain brought by the proposed RNNC scheme is examined in networks with various parameters.

The probabilistic connectivity matrices indicating the channel conditions can be arbitrary. In this section, they are generated randomly from $(0, 1]$ for numerical results.

The RNNC scheme is applied into networks with 3 and 4 nodes. The analytical results of the network reliability derived by the methods introduced in Section IV, are shown in Fig. 2. Additionally, the simulation results under the same network configuration are plotted for comparison. As can be seen in Fig. 2, the theoretical results match the simulation results tightly, which validates the theoretical analysis.

Consider a network consisting of three nodes, the reliability of the network at the end of round $R = 4$ can be calculated by $\psi(4)$ given by eq. (8). Then, substituting the corresponding entries of the probabilistic connectivity matrix into $\psi(4)$, the expression for the network reliability is simplified into a

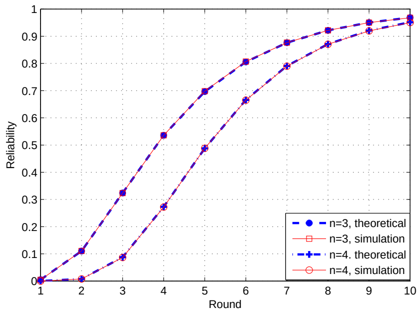

Fig. 2. The comparison between theoretical and simulation results of the network reliability applying random neighbour network coding when n=3 and 4. The probabilistic connectivity matrix for $n = 4$ is [1, 0.2, 0.3, 0.6; 0.4, 1, 0.5, 0.3; 0.6, 0.7, 1, 0.2; 0.3, 0.4, 0.5, 1], and for $n = 3$ is [1, 0.2, 0.3; 0.4, 1, 0.5; 0.6, 0.7, 1]; $\omega = 0.6$.

polynomial of a single variable ω, which is (rounded to four decimal places): $\psi(4) = -0.0003\omega^6 + 0.0083\omega^5 - 0.0468\omega^4 + 0.1362\omega^3 - 0.4154\omega^2 + 0.1285\omega + 0.5432$, where $\omega \in [0, 1]$.

Then, differentiation is calculated, $\frac{d(\psi(4))}{d\omega} = -0.0018\omega^5 + 0.0415\omega^4 - 0.1872\omega^3 + 0.4086\omega^2 - 0.8308\omega + 0.1285$. Let $\frac{d(\psi(4))}{d\omega} = 0$, there exists a solution for ω, subject to the constraint that $\omega \in [0, 1]$, which also gives a maximum value for $\psi(4)$, where $\psi(4) = 0.5537$, the corresponding $\omega = 0.8325$. In Fig. 3 (a), the reliability at the end of round $R = 4$ is plotted against ω. Further, Fig. 3 (a) shows that there exists a minimum value $\psi(4)_{min}$ when $\omega = 0$. Since $\omega = 0$ corresponds to case that all nodes transmit their original packets only, it concludes that the networks using the proposed coding scheme always outperform the non-coded network when $\omega > 0$.

Using the methods given in Section V-B, the optimisation is conducted on the expected number of rounds to reach the absorbing state. The minimum expected number of rounds is 4.7147 and the corresponding $\omega = 0.8460$. The expected number of rounds against ω is shown in Fig. 3 (b).

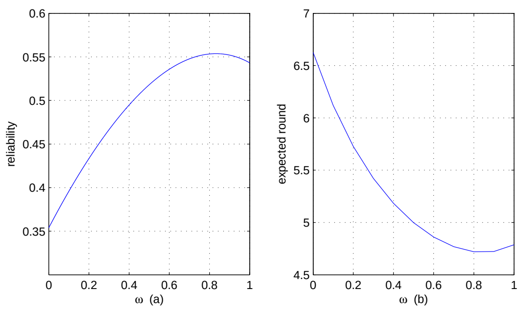

Fig. 3. The reliability of a three-node network at the fourth round (a) and the expected round to achieve reliability of 100 percent (b) when ω varies form zero to one.

The RNNC is applied to networks with different configurations. The reliability benefit made by RNNC is shown Fig. 4. The tuning parameters are set to the optimal values that maximise the reliability at the fourth round, which are 0.8435 and 0.9650 respectively. The reliability of non-coded networks and networks applying the random linear network coding scheme where the finite field is $GF(2)$ are plotted for comparison. The reason to choose $GF(2)$ is that it is fair to compare with RNNC in terms of computational complexity.

It is shown in Fig. 4 that networks applying the proposed RNNC scheme outperform other networks. Moreover, in some cases, this gain in reliability is considerable. For example, under the second network setting, the reliability using RNNC at round $R = 10$ is 0.7685, which is an improvement of 272.34 percent compared with the reliability in the non-coded network, which is 0.2064.

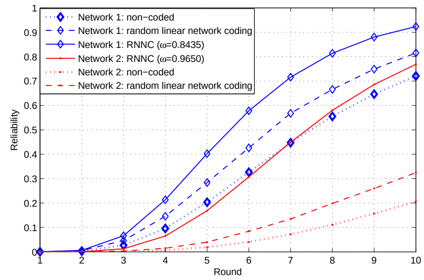

Fig. 4. The reliability comparison of networks with four nodes. The probabilistic connectivity matrices for the first and the second network settings are [1,0.3,0.5,0.4; 0.6,1,0.2,0.6;0.7,0.3,1,0.4;0.5,0.3,0.2,1] and [1,0.1,0.5,0.4; 0.6,1,0.2,0.6;0.7,0.3,1,0.1;0.1,0.3,0.2,1] respectively. The tuning parameters are set to be the optimal value that maximises the reliability at the end of Round 4, which are 0.8435 and 0.9650 respectively.

VII. Conclusion and future work

This paper proposes a random neighbour network coding scheme allowing a source node to perform network coding on-the-fly according to its received packets. The reliability of networks applying the proposed coding scheme is analysed and the improvement in network reliability has been shown. Further, the optimal random neighbour network coding scheme in a network with given probabilistic connectivity matrix is obtained to maximise the reliability at a given round or to minimise the expected time to reach the absorbing state. It is concluded that it may not provide optimal results for a node to take every opportunity to perform coding.

The tuning parameter ω is kept the same for every node in this work. In the future, each node can be assigned with an individual tuning parameter, ω_j where $j \in \{1, 2, ..., n\}$ so that each node can choose its own tuning parameter according to the channel condition to achieve a higher reliability in packet transmission in the network.

References

[1] R. Ahlswede, C. Ning, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[2] S. Katti, H. Rahul, H. Wenjun, D. Katabi, M. Medard, and J. Crowcroft, "Xors in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497–510, 2008.

[3] N. Dong, T. Tuan, N. Thinh, and B. Bose, "Wireless broadcast using network coding," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 914–925, 2009.

[4] W. Zhe and M. Hassan, "Blind xor: Low-overhead loss recovery for vehicular safety communications," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 35–45, 2012.

[5] M. Ghaderi, D. Towsley, and J. Kurose, "Reliability gain of network coding in lossy wireless networks," in *Proceedings of IEEE INFOCOM*, 2008, pp. 2171–2179.

[6] M. Nistor, D. E. Lucani, T. T. V. Vinhoza, R. A. Costa, and J. Barros, "On the delay distribution of random linear network coding," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 5, pp. 1084–1093, 2011.

[7] S. Dasgupta and G. Mao, "On the quality of wireless network connectivity," in *Proceedings of IEEE Globecom*, pp. 518–523, 2012.

[8] S. W. J., *Probability, Markov chains, queues, and simulation : the mathematical basis of performance modeling*. Princeton, N.J. : Princeton University Press, 2009.