Effect of link failure on QoS parameters under the influence of field and generation size in wireless network

Syed abidhusain[1] & Baswaraj Gadgay[2]

[1] Dept of E&C BLDEA college of engg & technology and Visvesvaraya Technological University of Belagavi -590018

[2]Visvesvaraya Technological University Regional Campus, Kalaburagi, Karnataka, India -585105

[1] abidsyed4u@gmail.com, baswaraj_gadgay@vtu.ac.in [2]

Abstract

Monitoring trails can be used to identify faults in any wireless network connection. The problem with such systems is that it is frequently impossible to differentiate links based on their expected conditions in various network topologies. Random linear network coding (RLNC), delivers compression as needed in each specific network by operating decentralize dly and tolerating network changes or link failures, naturally generalising error exponents. We employ RLNC to investigate the impact of link failure conditions on QoS measures such as latency, congestion, and loss rates. The results of the studies demonstrate that the QoS parameters are not severely impacted even in the event of link failure.

Keywords: Finite Field, random linear network coding (RLNC), link failure.

1. Introduction

The optimised use of system resources is one of the primary concerns when designing any network[18].Ensuring robust and reliable communication is critical to the success of any communication network. A network administrator's main challenge is to promptly identify and fix link faults and node failures[19].The complexity of contemporary gatherings, which necessitate regular QoS guarantees (such as online gaming, video conferencing, etc.), has increased the necessity for accurate and timely network monitoring technologies. This makes it simpler for network engineers and Internet service providers (ISPs) to monitor network performance. The phrase "network tomography" was first used by Vardi [1] to refer to a collection of techniques for determining internal link properties and evaluating connection congestion conditions. Numerous critical network components, including loss rate, delay distribution, bandwidth availability, and origin-destination traffic, must be understood and taken into account.However, only a few of these characteristics, such as origin-destination traffic, may be obtained directly from network hardware (like switches and routers). The link-level loss rate, delay distribution, and available bandwidth are a few examples of other qualities that network devices directly assess. It is frequently impractical to do so in a large network comprising many independent systems due to security challenges and other problems. Amid route finding, buffer overflow from data packet buffering could lead to packet losses. If the network becomes more dynamic, then this problem can degrade.The link-level loss rate, delay distribution, and available bandwidth are a few examples of other qualities that network devices directly assess. It is frequently impractical to do so in a large network comprising many independent systems due to security challenges and other problems. Amid route finding, buffer overflow from data packet buffering could lead to packet losses. If the network becomes more dynamic, then this problem can degrade.The frequency of route discoveries increases as the number of connection failures does. Additionally, the frequency of route discovery has an impact on the QoS performance of the network because each discovery adds significant packet overhead.

A network should be resilient so that it can endure errors in dangerous situations. The focus of robustness is how the network reacts when nodes or links are removed. It is simple to consider the network that is generated by removing the failing links when given a network and a pattern of link failures, which is similar to looking at static solutions. In a static circumstance, the network is unaware of the particular failure pattern. Each node is anticipated to broadcast a function of the observed random processes on its outgoing edges in order for the functions to be independent of the current failure pattern. As a result, unsuccessful connections ought to show the link constant as zero.By integrating numerous data packets into a single transmission entity, network coding is a powerful approach that can handle any challenging condition in communication networks to improve throughput, dependability, and latency. This technique gives improved flexibility in handling various traffic patterns and enables more effective network resource utilisation. The dependability of the network is essential to contemporary communication systems. Random Linear Network Coding (RLNC), one of many methods to increase reliability, has become a viable approach. In this post, we will talk about the idea of a link failure in the context of RLNC, how it affects network performance, and how to prevent and identify such failures. We'll also look at RLNC's benefits and how it can be used in other fields.

In the remaining parts of this work, we first examine the various previous studies in this direction in Section2. In Section 3, we see minimisingthe effect of link failure on QoS using RLNC. Section 4examines monitoring link failuresusing random linear network coding. Finally, we examine the simulation results and discuss them in Section 5. To grasp random linear network coding better, we have briefly discussed finite field theory in the appendix.

2.Related work

A. Background

The work in [2] describes a real-time native packet retrieval process using a fault-tolerant routing architecture that uses random linear network coding. Through the integration of random linear network coding and multi-link routing algorithms, this method improves the conventional method of choosing coding nodes. According to [3], a genetic algorithm based on gene number and fitness cooperative mutation is proposed under the name NF-QGA. The number uses an adaptive adjustment method for the rotation angle. The recommended method offers a quick convergence time and good optimisation potential while resolving concerns with network coding resource optimisation. In a mobile and unstable environment, secure packet transmission using stateless protocols like REST HTTP completely relies on the transmitter side.The work in [4] linked REST HTTP with RLNC to address this issue and reduce unnecessary retransmissions. The technique in [5] combines RLNC with cryptographic operations to improve transmitted communications' effectiveness and dependability while boosting their resistance to attacks. The work in [6] introduces a novel method for locating any congested link inside a network at any time by extending sent packets, which raises the complexity of the procedure. The study in [7] studied a many-to-one stream paradigm for proactive, reliable data transfer systems with requirements for resilience. The method in [8] can be successfully applied to the graph theory application in studying link failure cases.The Network Coding-based Fault-Tolerance Mechanism (NCFM) offers a greedy grouping mechanism to divide the topology into simpler logical units. It then uses random linear network coding to build a spanning tree that offers linearly independent coding options. The numerical results demonstrate that this transmission method outperforms existing ones in reducing the risk of packet loss, the resource redundancy rate, and the average delay while boosting the effective throughput rate. The work in [10] has developed formulas for a few QoS parameters, such as transmission rate, to analyse quantum networks with defective links. The research in [11–15] focuses on fault-tolerant concerns while investigating network structure, node placement, and routing methods.

B. Addition

Initially, it was thought that random linear network coding couldn't be employed for network monitoring applications. We are motivated to further investigate the impact of link failure on QoS parameters by the work in [6]. Multicast trees are used in this method to locate blocked connections. when the shortest route between the source and the destination is thought to have been blocked for an unforeseen reason. Traffic is guided along the next alternate and shortest route. The QoS metrics in a link failure scenario are contrasted with the values in a link failure case-free scenario. The results indicated that there isn't much of a difference. This demonstrates RLNC's capacity to survive any sudden changes in the network.

3.Minimizing the effect of link failure on QoS using RLNC

A.RLNC function

A key component of any communication network's effectiveness is ensuring strong and dependable communication. Any network administrator's main challenge is quickly identifying and fixing link faults. However, RLNC can be used to monitor the reception of encoded packets at the receiver side in order to detect link failure conditions. The framing points help to explain how RLNC operates to the benefit of any network[16][17]:

- Packet Encoding: At the source node, data packets are encoded using RLNC. These coded packets are redundant because they contain data from many original packets.
- Transmission: Coded packets are sent over the network, maybe including a failing connection.
- Reception:The destination node recieves both coded and any directly transmitted packets.
- Decoding : The original data is decoded by the destination node using the coded packets it has just received. Some coded packets may be lost or corrupted if a link failure occurs.The destination node keeps track of the decoding procedure to detect link failures. It can infer a link failure status if it has trouble decoding the data because of malformed or missing packets.
- Link Failure Detection: The destination node monitors the decoding process. If it encounters difficulties while decoding the data, such as missing or corrupted packets, it can infer a link failure condition.

B. Recovery Strategies

Once a link failure is detected, various recovery strategies can be applied:

- **Re-transmission:** The destination node can request re-transmission of missing packets from the source or intermediate nodes.
- **Use of Redundant Packets**: The destination node can use the redundancy introduced by RLNC to recover the lost information without requiring re-transmission.
- **Route Switchover**: In cases where multiple routes are available, the network can automatically switch to an alternative path, bypassing the failed link.

C. Benefits

- **Efficiency:** RLNC can detect link failures without the need for continuous monitoring or complex signalling protocols.
- **Reduced Latency**: The detection and recovery process can be swift, minimising communication disruptions.
- **Improved Reliability**: The redundancy introduced by RLNC enhances data recovery capabilities, reducing the impact of link failures.
- **Network Load Distribution**: Load can be evenly distributed across multiple links, reducing the risk of congestion and failures.
- **Dynamic Adaptation**: RLNC can adapt to changing network conditions, making it suitable for dynamic environments.

D. Challenges and Considerations

While RLNC is a powerful tool for link failure detection, it's essential to consider some challenges:

- **Overhead**: Encoding and decoding introduce computational overhead, which may affect network performance.
- **Implementation Complexity**: Implementing RLNC may require specialised hardware or software components.
- **Compatibility**: Compatibility issues may arise when integrating RLNC into existing network infrastructure.
- **Security**: Adequate security measures must be in place to prevent malicious tampering with coded packets.

4.Monitoring Link Failure Using Random Linear Network Coding

Any wireless network that experiences link failure faces an intriguing problem that requires a fix without violating the QoS standards. To what extent is the network configuration still guaranteed to be successful under a link failure circumstance? is the question we are trying to answer in this work. As was already said, network coding is a method that can make a network more resistant to non-ergodic link failures. According to analysis, network coding techniques can be applied to a particular connection failure pattern without needing to be customised. We concentrate on connection failure scenarios in networks, presuming that a link is either fully operational or has been removed from the network.We use binary vectors to examine each component of a connection failure pattern. A broken link's database position is one if it breaks; otherwise, it is zero. If the supporting links associated with the failure pattern have been removed, a network is considered to be solvable under the link failure condition. The investigation of the solvability of a particular failure pattern, which is a straightforward process, is no longer as interesting as finding common solutions for classes of failure patterns.

To carry on the conversation, we use the idea outlined in [6]. Assume that all nodes in the network, involves in the network coding process, except the source and destination nodes.Let D_l be the signal on an outgoing link $l \in E$, such that :

$$D_l = \sum_{j\in E=v} aD_j, v = o(l), l \in E \quad (1)$$

$$T_{ek}(d) = T^i(d): e_k \in T^i(d) s.t. o(e_k) = s \quad (2)$$

where o(l) and d(l) are respectively, the source and destination nodes of link $T_{ek}(d)$. The addition and multiplication operations in Eq. (1) ,are performed under finite field $F(2^q)$ (for more information, see [1]).$T_{ek}(d)$ is the collection of all links from s to d. The ith link between s and d is represented by Ti(d).Assume that source s has K outgoing connections $\{e_1, e_2,.., e_K\}$ and the symbol k is sent through the outgoing connections ek, where { k = 1, 2,.., K}. If the source delivers a symbol such as $F(2^q)$ across T(d), then destination will recieve :

$$D[n] = a \prod_{l\in P^i(d)} c_l = ab_i(G) \quad (3)$$

where $c_l \in F_2^q$ is the coefficient of the link $T^i(d)$ and $b_i(G) \epsilon F_2^q$ is the product of the LNC coefficients of all links on the ith link $P^i(d)$, from source to destination.If s sends the symbol $a_k[n]$, in time slot n through the k-th outbound link ek, {where k = 1,..., K}. Then it travels along all the links $T^i(d) \epsilon T_{ek}$. Because of the linear nature of network coding, the following information is superimposed on the data that the destination receives, in time slot n:

$$D[n] = \sum_{k=1}^{M} m_k [i] = \sum_{p^i} \prod_{l\in P^i(d)} c_l \quad (4)$$

where $$a'^{T}[n] = \left[a_k[n] \ldots a_k[n], a^{T}[n] = a'^{T}_{k}[n]\right]_{k}^{K} = 1$$

$$b^{T}(G) = \left[\prod_{l \in P^{i}(d)} c\right]_{[i=1]}^{N} = [b_i(G)]_{[i=1]}^{N} \qquad (5)$$

We call b(G), the total network coding vector of the graph G. If M symbols that constitute a packet are sent in M time slots, then the destination receives:

$$D_{Mx1} = A_{MxN} b(G)_{Nx1} \qquad (6)$$

where A is a MxN matrix whose nth row is a T[n], the training symbols sent in time slot n. By construction, it is evident that for { k = 1, 2,..., K}, the columns of A that correspond to Tek are equal. Packets are significantly delayed and presumed to have been lost at the destination when a connection is overloaded to an unreasonable degree. The vector (G) given in (9) is connected to the full network coding vector of the graph Gl, denoted by (Gl), using the edge deleted subgraph Gl (Vl, El) as a model for the network with link l in a congested state (for a definition of an edge deleted subgraph from earlier, see [8]). It is clear that even if the congested links are busy, packets moving over them won't be affected, as observed in [6].

i.e. $$\mathbf{b}(G) = \begin{cases} b_i(G) if l \notin T^{i}(d) \\ 0 otherwise \end{cases} \qquad (7)$$

where l is the clogged link and i(G) is the ith entry of (G). Assume that link l,is congested and source ‘s’, transmits k[n] over link T_{ek} in time slot n. The destination node recieves:

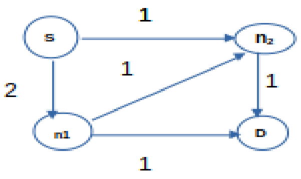
$$D^{l}[n] = \sum_{k=1}^{N} a_k [n]. \sum_{P^{i}(d) \in P_{ek}} b\,(G_l) \qquad (8)$$

throughout period n. When link l is congested and probes are being sent in M consecutive time slots, the vector form equation shown below may be constructed using (7) and (8):

$$D_{Mx1}^{l} = A_{MxN} b(G_l)_{Nx1} \qquad (9)$$

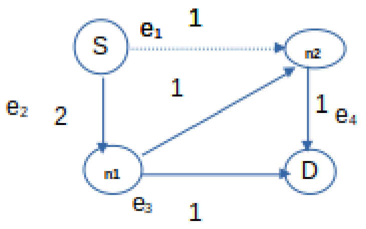
where D^{l} is the vector of symbols seen at the destination. Equations (6 and 9) may be compared to demonstrate how, for the given matrix A, the received symbols may fluctuate in response to network congestion. We shall demonstrate next that this occurs if (G) and A meet specific requirements, providing the opportunity to pin point the congested link. We then give an example to serve as illustration

Example: Consider the network shown in fig1,having four nodes and four links.To reach from source s to D,a packet has to traverse through any one of the fallowing paths:1.$s \rightarrow n_2 \rightarrow D$or 2.$s \rightarrow n_1 \rightarrow D$ or 3. $s \rightarrow n_1 \rightarrow n_2 \rightarrow D$



TableI:Cost matrix for node s

source	cost	via	Dest
s	2	-	n_1
s	1	-	n_2
s	2	n_2	D



TableII:Cost matrix for node s when link e1 is broken

source	cost	via	Dest
s	2	-	n_1
s	3	n_1	n_2
s	3	n_1	D

Fig1: Dashed links represent intermediate nodes.Ge1:link e1 is congested or failured,dash arrow represents link removed fom network

In case of link failure,say e1,the packets are rerouted via any one of the paths as illustrated in TableI&II.Obviously route described in TableII,takes a longer path as the cost of reaching the destination increases.This applies to source destination pairs.

5.Simulation results and discussion

A.Packet buffering

A packet buffer is memory that has been designated for the storage of packets that are being transported across a network or that have already been received. The quantity of packets saved as a function of generation size as they pass through various nodes is depicted in Fig. 2 below. In the event of a link breakdown, more packets will be delayed as generation size increases.

Fi

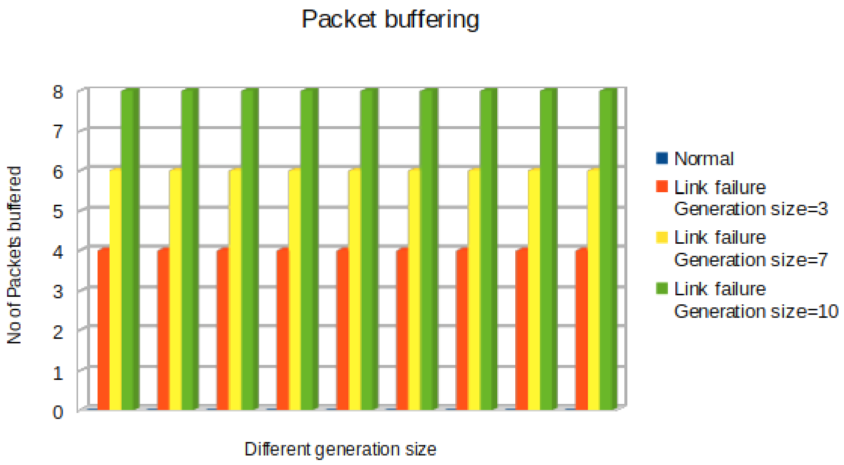


Fig2:Packet buffering

B. Delay

Figs. 3a and 3b illustrate the effect of link failure under different field sizes and generation sizes on delay. Link failure did not have a considerable effect. We have designed the network in fault-tolerant mode. But as such, large files incur a large delay when compared to small files.

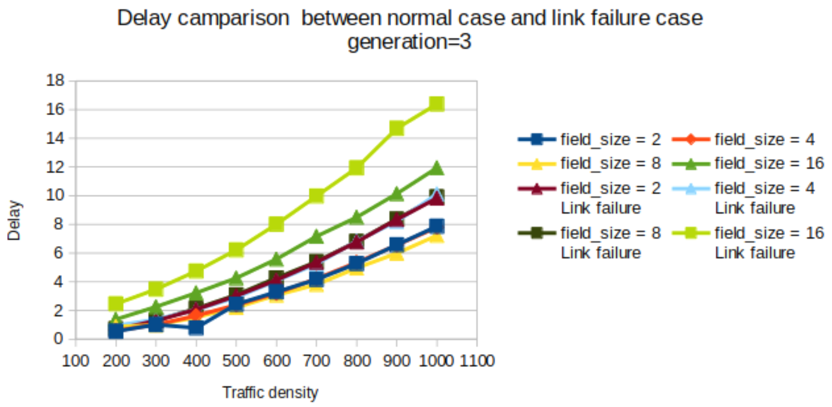


Fig3a:Traffic density verses Delay for different field size generation=3

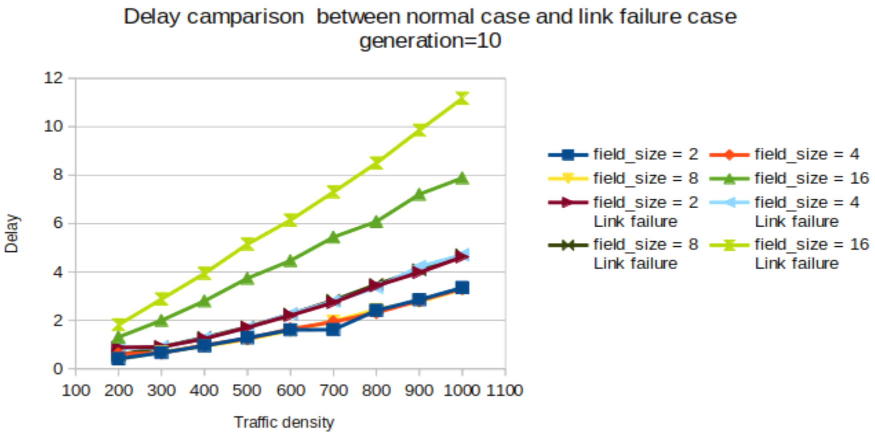


Fig3b:Traffic density verses Delay for different field size generation=10

Conclusion

Detecting link failure conditions using random linear network coding is a valuable technique that enhances the reliability and resilience of communication networks. By introducing redundancy and efficient coding, RLNC allows for swift detection and recovery from link failures thus managing important QoS parameters like delay and buffer, reducing data loss and service interruptions. However, network administrators should carefully consider the associated challenges and plan for an effective implementation to harness the full potential of RLNC in their networks.

References

[1]E. Horowitz, “Modular arithmetic and finite field theory: A tutorial,” in Proceedings of the second ACM symposium on Symbolic and algebraic manipulation, 1971, pp. 188–194

[2]Peng, Y., Song, Q., Yu, Y., & Wang, F. (2014). Fault-tolerant routing mechanism based on network coding in wireless mesh networks. *Journal of Network and Computer Applications*, *37*, 259-272. https://doi.org/10.1016/j.jnca.2013.02.015

[3]Liu, T.; Sun, Q.; Zhou, H.; Wei, Q. Optimization of Network Coding Resources Based on Improved Quantum Genetic Algorithm.*Photonics***2021**,*8*, 502. https://doi.org/10.3390/photonics8110502

[4]C. V. Phung, J. Dizdarevic, F. Carpio and A. Jukan, "Enhancing REST HTTP with Random Linear Network Coding in Dynamic Edge Computing Environments", 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 2019, pp. 435-440, doi: 10.23919/MIPRO.2019.8756782.

[5]Reem Melki, Hassan N. Noura, Ali Chehab,"Efficient and secure multi-homed systems based on binary random linear network coding",Reem Melki, Hassan N. Computers & Electrical Engineering,Volume 87,2020,106774,ISSN 0045-7906,https://doi.org/10.1016/j.

[6]Mohammad H. Firooz, Student Member, IEEE, Sumit Roy, Fellow, IEEE,"Link failure monitoring via network coding",Local Computer Networks (LCN), 2010 35th Conference IEEE .**DOI:**10.1109/LCN.2010.5735682

[7]Peng, Y.; Wang, X.; Guo, L.; Wang, Y.; Deng, Q. An Efficient Network Coding-Based Fault-Tolerant Mechanism in WBAN for Smart Healthcare Monitoring Systems.*Appl. Sci.***2017**,*7*, 817. https://doi.org/10.3390/app7080817

[8] J. Clark and D. Holton, A first look at graph theory. World Scientific Publishing Company, 1991.

[9]C. Lydick, "Tutorial: Network coding using opnet and matlab," Thech-nical report, May 2009.

[10]Masahito Hayashi,Seunghoan Song,"Quantum state transmission over partially corrupted quantum information network",Phys. Rev. Research**2**, 033079 – Published 15 July 2020 by American physical society

[11]Qiu, T.; Zhao, A.; Xia, F.; Si, W.; Wu, D. ROSE: Robustness Strategy for Scale-Free Wireless Sensor Networks.IEEE/ACM Trans. Netw. 2017, 1–16. [CrossRef]

[12]Chen, X.; Kim, Y.-A.; Wang, B.; Wei, W.; Shi, Z.; Song, Y. Fault-tolerant monitor placement for out-of-band wireless sensor network monitoring. Ad Hoc Netw. 2012, 10, 62–74. [CrossRef]

[13]Boukerche, A.; NelemPazzi, R.W.; Araujo, R.B. Fault-tolerant wireless sensor network routing protocols for the supervision of context-aware physical environments. J. Parallel Distrib. Comput. 2006, 66, 586–599.[CrossRef]

[14]Ning, Z.; Liu, L.; Xia, F.; Jedari, B.; Lee, I.; Zhang, W. CAIS: A Copy Adjustable Incentive Scheme in Community-based Socially-Aware Networking. IEEE Trans. Veh. Technol. 2017, 66, 3406–3419. [CrossRef]

[15]Qiu, T.; Chen, N.; Li, K.; Qiao, D.; Fu, Z. Heterogeneous ad hoc networks: Architectures, advances and challenges. Ad Hoc Netw. 2017, 55, 143–152. [CrossRef]

[16] Ho, Tracey, et al.," Random Linear Network Coding: A Survey". IEEE Communications Surveys & Tutorials 16.3 (2014): 1535–1559

[17]Fitzek, Frank H.P., and Marcos D. Katz. "Cooperative Communications and Networking. Cambridge University Press, 2009".

[18]J.D. Mallapur et al.(2010), "Fuzzy Based Bandwidth Management for Wireless Multimedia Networks",January 2010 International Conference on Recent Trends in Business Administration and Information Processing", Trivandrum, Kerala, India Communications in Computer and Information Science 70:81-90,DOI:10.1007/978-3-642-12214-9_15

[19]Shaikh,syed and Agusbal (2016),"An algorithm for sensor node failure detection in WSNs", International Conference on Electrical, Electronics, and Optimization Techniques(ICEEOT-2016) DOI:10.1109/ICEEOT.2016.7754912