

Network-Coded Multigeneration Protocols in Heterogeneous Cellular Networks

Roberto Torre¹[0000–0003–1536–4790], Sreekrishna Pandi¹[0000–0002–5684–8831],
and Frank H. P. Fitzek¹[0000–0001–8469–9573]

Deutsche Telekom Chair of Communication Networks,
roberto.torre@tu-dresden.de,
sreekrishna.pandi@tu-dresden.de,
frank.fitzek@tu-dresden.de,
WWW home page: <https://cn.ifn.et.tu-dresden.de/>

Abstract. In the upcoming era of 5G, the number of devices will increase massively, defining a heterogeneous wireless network. Nodes will be gathered in Mobile Clouds, and communicate between peers to achieve a general benefit. To provide packet resilience, error correction codes will be used. In particular, Random Linear Network Coding is standing out as one of the most successful ones. The interplay between Network Coding and Mobile Clouds creates a mesh network where nodes may receive information from multiple sources. However, RLNC was optimized to provide in-order-delay in D2D communications. RLNC need to adapt to a new heterogeneous mesh network where nodes receive packets from multiple paths. In this paper, we propose a method to improve conventional RLNC protocols by making them be able to manage multiple generations simultaneously. We also identify possible trade-offs between conventional RLNC protocols and our new approach. We conclude that multigeneration protocols have better behavior in terms of throughput and resilience, but the average latency per packet decoded is higher.

Keywords: Random linear network coding, RLNC, multipath, multigeneration, heterogeneous, cellular networks, mobile clouds

1 Introduction

Cisco Technical Report of 2017 [1] reported that the number of devices will massively increase up to three times the global population in 2021. New infrastructure elements, such as femto/pico base stations, fixed/mobile relays, cognitive radios, and distributed antennae are being massively deployed, thus making future 5G cellular systems and networks more heterogeneous [2].

The increase in mobile traffic reported by Cisco made companies look for solutions in order to decrease data traffic. One of the most successful ones is Mobile Clouds [3], a cooperative arrangement of dynamically connected nodes sharing opportunistically resources. These nodes need to be co-located to be able to communicate among themselves. If the number of nodes cooperating is higher

than 3, clients might expect data from multiple paths, leading to a new paradigm of multipath communication among multiple users. Apart from Mobile Clouds, Wireless Mesh [4] or Platoon Communication [5] will benefit from this system.

New error correction techniques such as Random Linear Network Coding (RLNC) [6] are stepping up in the last years due to its good performance in terms of throughput and packet resilience. Furthermore, it has been studied in [7] that the combination of RLNC and Mobile Clouds can lead to high increases in terms of network throughput, packet resilience, and substantially decrease the energy consumption. RLNC protocols are however not optimized for heterogeneous networks and multipath communications [8], but for providing in-order-delay packets in D2D communications. Hence, packets that arrive out of order might be discarded instantly. Conventional RLNC protocols rely on their redundancies not only to recover from packet erasures but also from the jitter of the network. However, in scenarios where jitter is non negligible such as cellular communications [9], mobile clouds or wireless mesh communications, being able to utilize the packets that arrive out of order may increase the decoding probability, and as a consequence, throughput, and packet resilience.

In this paper, we propose a new algorithm to make RLNC protocols be able to manage multiple generations simultaneously. The destination, where the packets should be decoded, will now be organized in standalone **subdecoders**, a packet buffer that is created every time a packet with new generation arrives. We compare our multigeneration approach with the conventional one, identify possible trade-offs, and perform simulations in order to study how our approach is behaving. We confirm that conventional RLNC protocols are abusing of the coding redundancies to cope with packet arrival fluctuations and this would diminish with a multigeneration approach. However, if a packet is lost, the average latency per packet will increase. RLNC protocols will benefit from this because their redundancy algorithms will no longer require focussing on jitter, but only on packet errors.

The remainder of the paper is organized as follows. In section 2 we give a detailed information about the state of the art. In section 3 we introduce the problem RLNC protocols are currently dealing with. Section 4 gathers our solution proposed. In section 5 we compare the conventional mode with our proposed solution by running software simulations. In section 6 the conclusion of our work is presented.

2 Related Work

Heterogeneous Cellular Networks. Cellular networks have suffered massive changes in the last years. New elements such as smartphones, smart cars, sensors, and IoT have recently appeared making the environment inside the cell more heterogeneous [2]. The amount of traffic is also increasing year by year, making the network ideal for short-range communications, platoon communication, wireless mesh, and cooperative networks [10, 11].

Cooperative Mobile Wireless systems. Cooperative communications are nowadays taking the lead in video streaming and data dissemination. Nodes tend to group into Mobile Clouds [12, 13] to increase user energy performance [14] or network throughput. All those technologies exploit the idea of a fully mesh cooperative network [8], where packets can be received from several paths. Hence, the next generation of mobile networks is moving forward a multipath cooperative network.

Random Linear Network Coding. In this emerging heterogeneous networking environment where cellular networks are continuously adapting to new user requirements, it was demonstrated that the use of Network Coding can increase wireless network throughput [15]. A high-performance improvement used to overcome those errors is Random Linear Network Coding (RLNC), which was first introduced in [6]. Some studies also stated that the interplay of Random Linear Network Coding along with different technologies such as Cooperative Networking [16] can substantially increase network throughput and packet resilience in comparison with its predecessors [17], and have created an innovative communication paradigm known as Network-Coded Cooperative (NCC) networks [7, 18, 19]. However, RLNC has been optimized to provide in-order-delay packets in D2D communications. Regarding future multipath communications, RLNC needs to adapt by using tools like the ones proposed in this paper.

3 Problem Description

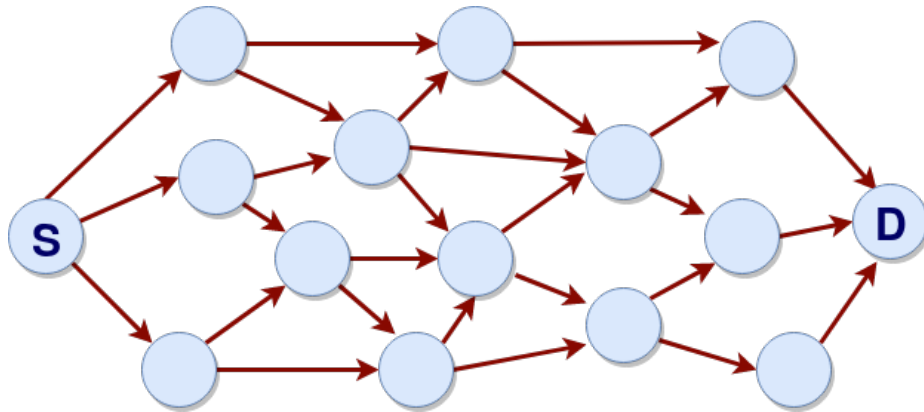


Fig. 1: Example of a mesh network topology with multiple paths. Packets are generated in the Source (S) and must arrive at the destination (D). Multiple paths can be selected for the communication so the propagation delay may vary significantly.

RLNC protocols are optimized for providing in-order-delay in D2D communications. However, when the jitter of the network is high, the performance of

RLNC decreases significantly. This increase in jitter can happen due to the topology in a multipath environment since the packet can arrive from different paths of the network. An example of a multipath mesh network can be observed in figure 1. In [20], the authors study the jitter in Ad Hoc networks using different protocols. They obtain average jitters up to 0.1 msec. In this paper, we will use these values to see how RLNC would react to this jitter.

RLNC is a linear block code, like Reed-Solomon or Hamming codes. This means that the transmitted packets are grouped into blocks, then, new coded packets are generated by creating a linear combination of the packets in the block, and later transmitted through the network. The decoder needs to receive as many linearly independent packets as the size of the block in order to be able to decode the information of the whole block. In figure 2 the difference between RLNC and a conventional protocol is depicted, and how packets inside a block are linearly combined.

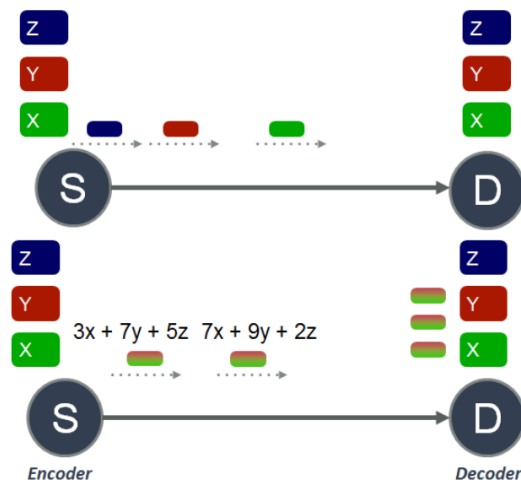
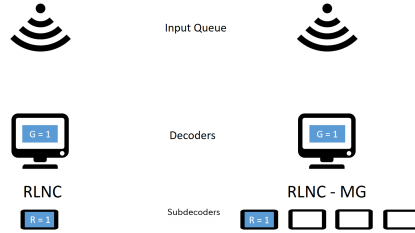
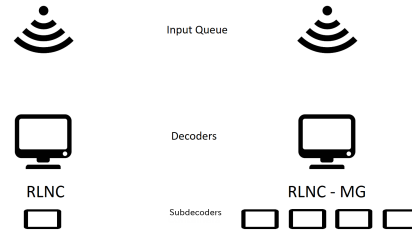


Fig. 2: Difference between a conventional protocol (top) and an RLNC protocol (bottom). Meanwhile in the conventional protocol the packets are sent raw, in the RLNC example the packets are gathered in blocks, and linearly combined.

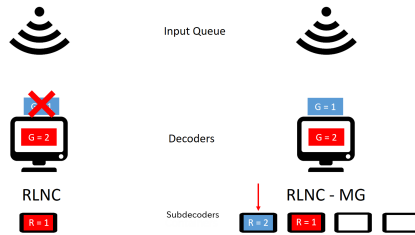
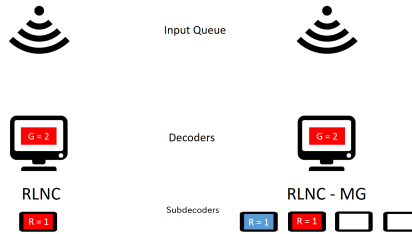
In RLNC, blocks are commonly known as generations. Each decoder in RLNC can handle only one generation. When the first packet of the next generation is received, the decoder understands that the older generation is over, and it will try to flush all possible data and store the new packet. In multipath scenarios with a non negligible jitter exists a high chance of receiving next-generation packets out of position. In the following pictures is depicted a possible scenario, where a protocol that handles multiple generations at the same time provides a better decoding ratio.

Step 1. As an example, an idle system with one decoder is presented. The decoder has no starting generation. We are assuming a wireless input where multiple sources can be delivering packets.



Step 2. The first packet arrives. After receiving it, the systems obtain the generation of the packet, and adjust their internal parameters (Generation, Rank, etc).

Step 3. A second packet arrives with a new generation number. In this case, the left system flushes its memory decoding all possible packets. The right system saves the second packet in a second subdecoder, adjusts its internal parameters, and awaits new incoming packets.



Step 4. A third packet arrives with an old generation number. The system on the left considers this is an old generation packet and discards it. The system on the right keeps it and updates the parameters of the corresponding subdecoder.

4 Multiple Generations in one Coder

The solution we propose in this paper for the problem explained in section 3 consists in making the recoders and decoders be able to handle multiple generations simultaneously. Thus, we add a new concept, the **subdecoder**, a standalone generation buffer which is created every time a packet with a subsequent generation arrives, then this subdecoder is attached to that generation, and when the generation is completed or the system needs to be flushed, the entity is destroyed. The subcoders are not aware of the rest, and they will not communicate among themselves. There will be a central logic that will create, destroy, and give orders to each of the subcoders. The logic of this central brain can be observed

in algorithm 1. This represents the workflow when an event(incoming packet) appears. In this algorithm, *Gen* refers to Generation and *In* to the incoming packet.

```

foreach incoming packet (In) do
  if Gen(In) > Current gen. then
    if current subdecoders = MAX_SUBDECODERS then
      Flush oldest subdecoder;
      Destroy oldest subdecoder;
      Update decoder metadata;
    end
    Create new subdecoder;
    Increase decoder max. generation;
  else if Current gen. <= Gen(In) < Oldest gen. then
    Find Gen(In) subdecoder ;
    Update Gen(In) subdecoder ;
  else
    Discard packet;
  end
  while source packet to decode do
    Decode last packet;
    Forward packet to next layer;
    if Gen. decoded then
      Destroy oldest subdecoder;
      Update decoder metadata;
    end
  end
end

```

Algorithm 1: Decoder workflow for each incoming packet

The workflow of the algorithm is described as follows. When a packet arrives at the decoder, the generation of the incoming packet will be extracted. If it belongs to a previous generation than the oldest one in the decoder, the packet will be dropped. If it belongs to a generation between the oldest and the newest generation of the decoder, it will forward the packet to the corresponding subdecoder. This subdecoder will extract the information, update the subdecoder metadata, and forward the packet in case it needs to be recoded. If the generation of the incoming packet belongs to a later generation than the newest one in the decoder, it will first check if another subdecoder can be created. If not, the subdecoder attached to the oldest generation will be flushed and destroyed, leaving an empty spot to the new generation subdecoder. The metadata information of the decoder will be updated as well. When there is an empty spot, a new subdecoder will be created and attached to the new generation, the incoming packet will be forwarded to the subdecoder created and the decoder metadata will be updated.

With the incorporation of subdecoders inside a decoder, the probability of decoding will increase. However, there are some drawbacks that must be con-

sidered. The more subdecoders there are on the decoder, the longer the decoder would have to wait until it flushes the oldest subdecoder when a generation is not filled in time. This trade-off will be studied in the following section.

5 Simulations

Table 1: Parameter settings of the simulator

Parameter	Symbol	Settings
Packets sent	N	10.000 packets
Generation size	G	100 packets
Maximum subdecoders	C	4 subdecoders
Propagation time	T_p	1 ms
Packet length	ℓ	1500 bytes
Data rate	R	11.76 Mbps
Jitter	j	0..0.128 ms

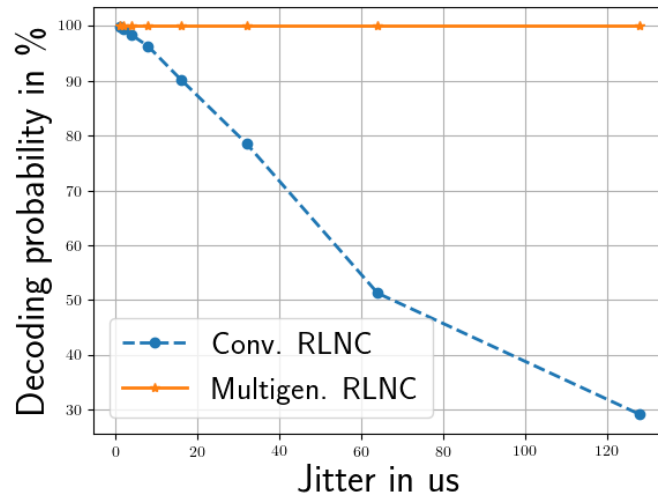


Fig. 3: Decoding probability of the conventional RLNC protocols and multigeneration protocols for the parameters in table 1

We consider a simulator of an LTE network, where a base station is encoding and sending data to a user equipment that acts as a decoder. The values of

the simulator are gathered in table 1. We send a total number of $N = 10.000$ packets, the number of packets each generation will have is $G = 100$. The amount of subdecoders the RLNC multigeneration protocol has is 4. The MTU is 1500 bytes, and the propagation time, data rate, and jitter are taken from [9, 20].

We are assuming the system is aware and adaptive to packet erasures in the channel. Hence, packet losses will only happen due to next generation early packet arrivals. By doing so, we are limiting error losses only to jitter and we can isolate the losses related to network jitter. In figure 3 the decoding probability of both systems is shown. It can be observed that when the jitter is minimal, the conventional RLNC protocols start to have problems decoding the whole message, which would lead to an increase of the redundancies and a more inefficient transmission in the network. On the other hand, the multigeneration protocol does not suffer the jitter fluctuations. If the jitter keeps being increased, the multigeneration approach would start losing packets, but the decoding probability of the conventional RLNC would decrease down to almost zero.

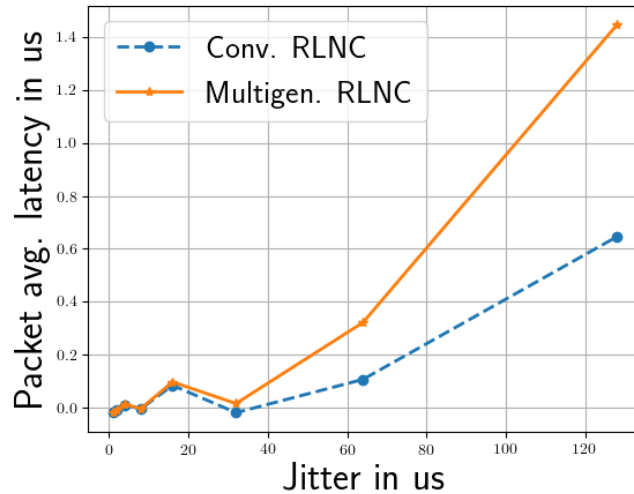


Fig. 4: Extra latency per packet of the conventional RLNC protocols and multigeneration protocols for the parameters in table 1

In figure 4 the average extra latency per packet is depicted. As expected, a conventional RLNC protocol has here a better performance, since all the packets that arrive do not have to wait for packets of previous generations. It can be observed that with a higher jitter, the average latency on each packet increases with a multigeneration protocol, but, on the contrary, there are no losses due to the early arrival of new generation packets. We can conclude that exists a trade-off between these two parameters, and further studies must be done in order to

model the advantages and disadvantages of using multiple generation protocols with different parameters, like the number of subdecoders used, the conditions of the network, the coding ratio, the generation size or the packet size.

6 Conclusion

RLNC has taken the lead of error correction codes in heterogeneous cellular networks in the last years. Nevertheless, RLNC protocols are optimized for D2D communications, but not for multipath, because they can only handle one generation at the same time.

In this paper we propose a novel way of adapting RLNC protocols, so the decoder is able to handle multiple generations simultaneously, making them more suitable for multipath communications. This is done by organizing the decoder in standalone subdecoders, buffers attached to each generation which are destroyed after the generation is decoded. We identify and study a trade-off between the packets decoded and the average latency per packet depending on the network jitter. We confirm that the conventional RLNC protocols have better performance in terms of latency, but the multigeneration approach provides better decoding ratio.

This paper opens up a new approach for RLNC protocols, however, further studies must be done. Studies with a higher complexity must be developed, varying different parameters such as the number of subdecoders used, the conditions of the network, the coding ratio, the generation size or the packet size. Moreover, the possibility of adapting the optimal amount of subdecoders on the fly should also be studied.

7 Acknowledgements

This project has received funding from the European Union's H2020 research and innovation program under grant agreement H2020-MCSA-ITN- 2016-SECRET 722424 [21].

References

1. Cisco: Cisco visual networking index: Forecast and methodology, 2016 to 2021. Technical report, Cisco Technologies (March 28, 2017)
2. Heath, R.W., Kountouris, M.: Modeling heterogeneous network interference. *IEEE Inf. Theory Appl. Workshop* (Feb 2012)
3. Fitzek, F.H., Katz, M.D.: *Mobile Clouds. Exploiting Distributed Resources in Wireless, Mobile and Social Networks*. John Wiley and Sons, Ltd, United Kingdom (2014)
4. Pandi, S., Wunderlich, S., Fitzek, F.H.P.: Reliable low latency wireless mesh networks from myth to reality. In: *2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC)*. (Jan 2018)

5. Jia, D., Lu, K., Wang, J., Zhang, X., Shen, X.: A survey on platoon-based vehicular cyber-physical systems. *IEEE Communications Surveys Tutorials* (2016) 263–284
6. Ho, T., Medard, M., Shi, J., Efros, M., Karger, D.R.: On randomized network coding. *Proc. 41st Annu. Allerton Conf. Commun., Control, Computi* (2003)
7. Renzo, M.D., Iezzi, M., Graziosi, F.: On diversity order and coding gain of multi-source multirelay cooperative wireless networks with binary network coding. *IEEE Trans. Veh. Technol.* (2013) 1138–1157
8. Gheorghiu, S., Toledo, A.L., Rodriguez, P.: Multipath TCP with network coding for wireless mesh networks. In: *2010 IEEE International Conference on Communications*. (May 2010)
9. 3GPP: Physical channels and modulation (September 2015)
10. Pedersen, M.V., Fitzek, F.H.P.: Mobile clouds: The new content distribution platform. *Proceedings of the IEEE* (2012)
11. Laneman, J.N., Tse, D.N.C., Wornell, G.W.: Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Trans. Inf. Theory* (2004) 3062–3080
12. Fitzek, F., Katz, M., Zhang, Q.: Cellular controlled short-range communication for cooperative P2P networking. *Wireless World Research Forum (WWRF)* 17 (2006)
13. Albiero, F., Katz, M., Fitzek, F.H.P.: Energy-efficient cooperative techniques for multimedia services over future wireless networks. In: *2008 IEEE International Conference on Communications*. (May 2008) 2006–2011
14. Albiero, F., Fitzek, F., Katz, M.: Cooperative power saving strategies in wireless networks: an agent-based model. *Symposium on Wireless Communication Systems* (Oct 2007)
15. Ahlswede, R., Cai, N., Li, S.Y.R., Yeung, R.W.: Network information flow. *IEEE Transactions on Information Theory* (July 2000)
16. Rossetto, F., Zorzi, M.: Mixing network coding and cooperation for reliable wireless communications. *IEEE Wireless Comms* (Feb 2011)
17. Militano, L., Condoluci, M., Araniti, G., Molinaro, A., Iera, A., Fitzek, F.H.P.: Wi-fi cooperation or D2D-based multicast content distribution in LTE-A: A comparative analysis. In: *2014 IEEE International Conference on Communications Workshops (ICC)*. (June 2014) 296–301
18. Pandi, S., Arranz, R.T., Nguyen, G.T., Fitzek, F.H.P.: Massive video multicasting in cellular networks using network coded cooperative communication. In: *2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC)*. (Jan 2018)
19. Fitzek, F.H.P., Heide, J., Pedersen, M.V., Katz, M.: Implementation of network coding for social mobile clouds [applications corner]. *IEEE Signal Processing Magazine* (Jan 2013) 159–164
20. Talooki, V.N., Rodriguez, J.: Jitter based comparisons for routing protocols in mobile ad hoc networks. In: *2009 International Conference on Ultra Modern Telecommunications Workshops*. (Oct 2009)
21. Rodriguez, J., et al.: Secret-secure network coding for reduced energy next generation mobile small cells. *ITA Conference* (2017)