

Performance of Random Linear Coding over Multiple Error-Prone Wireless Links

Pablo Garrido, David Gómez, Ramón Agüero, *Member, IEEE* and Luis Muñoz, *Member, IEEE*

Abstract—In this letter we derive an exact formulation for the performance of Random Linear Coding (RLC) when applied over multiple wireless links. We combine this technique with UDP so as to offer a reliable communication service. We extend a previous result, which only considered one single link, to embrace both multiple sources as well as varying quality of wireless links. We establish the number of excess packets that are required to successfully accomplish the communication and, based on the Bianchi model, we calculate the achieved throughput. We also propose a context-aware probabilistic transmission scheme that leads to a relevant performance gain. We use a thorough simulation-based study over the ns-3 framework to assess the validity of the proposed model and to broaden the corresponding analysis.

Index Terms—Random linear coding; wireless mesh networks; simulation

I. INTRODUCTION

In this work we analyze the performance of a scheme that exploits the combination of Random Linear Coding (RLC) and UDP to offer a reliable communication service, used e.g. to transfer a file. We start from the work of Trullols-Cruces *et al.*, who derived in [1] the exact probability to obtain k linearly independent packets after receiving N ($\geq k$), when the sender uses RLC over a Galois Field of size q , $GF(q)$. A single error-free link was considered in that work, and our goal is to broaden the model the authors proposed therein so as to include multiple error-prone wireless links. In this sense, we assume that the content is available in various nodes (as in the case of network caching), which send coded packets to a single receiver. We consider the IEEE 802.11 technology to assess the performance of our scheme and we exploit the well known Bianchi model [2] to derive the corresponding throughput.

The rest of this paper is structured as follows. In Section II we summarize the foundations of the RLC scheme to afterwards formulate the exact decoding probability, the number of excess packets and the corresponding throughput. Section III validates the analytical model by means of an extensive simulation campaign over the ns-3 framework, using IEEE 802.11 as the subagent technology. Finally, Section IV concludes the paper and outlines the future research lines we are already pursuing.

The authors are with the University of Cantabria, Communications Engineering Department, Santander, Spain (email: {pgarrido,dgomez,ramon,luis}@tmat.unican.es)

This work has been supported by the Spanish Government by its funding through the project COSAIF, “Connectivity as a Service: Access for the Internet of the Future” (TEC2012-38754-C02-01).

II. ANALYTICAL RLSC PERFORMANCE

A. Scheme Description

As already mentioned, the proposed scheme combines RLC and the UDP protocol to provide a reliable communication service, which could be used to e.g. transfer a file stored in multiple caches to a single receiver.

Each of the coding entities (implemented as a new layer between IP and UDP) at the source nodes receives the information from the upper layers, keeping it at their transmission buffers. The information is then coded, using blocks of k native packets (p_i), and delivered downwards as coded packets (\tilde{p}_j). These can be expressed as linear combinations of the native ones, $\tilde{p}_j = \sum_{i=0}^{k-1} c_i \cdot p_i$, where the c_i ’s are random coefficients, generated from a finite field $GF(q)$. We can also define the coded vector \vec{c}_j , as $\vec{c}_j = (c_0, c_1, \dots, c_{k-1})$. A source node will continuously send coded packets, until the destination acknowledges the successful decoding and reception of the current block. In that moment, the transmitter entity removes the k native packets from its buffer, starting the procedure with a new block, until the whole file has been completely received.

On the other hand, the decoding entity at the destination node uses two different storage modules: a matrix \mathcal{C} (of dimension $k \times k$) to store the received coded vectors, and a buffer to store up to k coded packets. Upon the reception of a packet \tilde{p}_j , the corresponding coded vector (\vec{c}_j) is appended to \mathcal{C} . Afterwards a rank calculation, $\mathbb{R}(\mathcal{C})$, is used to determine whether this vector is linearly independent of the existing stored vectors. In such case, the packet is stored at the buffer; otherwise, it is silently discarded and the vector is removed from \mathcal{C} . When the destination node has received k innovative packets, $\mathbb{R}(\mathcal{C}) = k$, it solves the corresponding linear equation system (see Section 4.3 in [3]) to decode the original block, which is delivered upwards to the application. Afterwards, an ACK is sent backwards to the source nodes, informing them about the successful decoding of the whole block. Note that whenever the destination receives a packet belonging to an already decoded block, it will immediately send another ACK, so that the sources realize they need to shift to the next block. In this way, we deal with the potential losses of these acknowledgement packets.

B. Exact Decoding Probability

We start from the probability of decoding k packets after receiving N , $N \geq k$, $\xi_q(k, N)$. This was obtained by Trullols-Cruces *et al.* [1], assuming a Galois Field of size q , and is given by

$$\frac{\xi_q(k, N)}{\xi_q^0} = \left[\begin{matrix} N \\ N-k \end{matrix} \right]_q + \sum_{i=1}^{N-k} (-1)^i \binom{N}{i} \left[\begin{matrix} N-i \\ N-k-i \end{matrix} \right]_q \quad (1)$$

where $\left[\begin{matrix} m \\ n \end{matrix} \right]_q$ are the q -binomial (or Gauss) coefficients [1] and ξ_q^0 is the probability of successfully decoding k packets after receiving k , which can be obtained with

$$\xi_q^0 = \xi_q(k, k) = \frac{q^{k^2}}{(q^k - 1)^k} \prod_{j=1}^k \left(1 - \frac{1}{q^j} \right) \quad (2)$$

The above result only accounts for the possible reception of linear combinations of already received packets. It does not consider neither the potential transmission losses or the network topology. We now assume that there are different nodes ($1 \dots M$) transmitting the information, and the links between each of them and the single receiver exhibit different qualities. We assume that each of the sources transmit n_i packets, while $\sum_{i=1}^M n_i = N$.

The probability of receiving j packets when the overall number of transmissions equals N is shown in (3). As can be seen, we use the multinomial distribution, $\mathcal{M}(\mathbf{x}, \bar{\pi})$. \mathbf{x} is a vector of $M+1$ elements, where x_i ($i \leq M$) is the number of packets received over the i^{th} link and x_{M+1} accounts for the $N-j$ lost packets; $\bar{\pi}$ is a vector with the probabilities that a single transmitted packet was received over a particular link ($i = 1 \dots M$) or lost ($i = M+1$).

$$\mathbb{P}_{rx}(j, N) = \sum_{t_2=0}^j \sum_{t_3=0}^{j-t_2} \dots \sum_{t_M=0}^{j-\sum_{q=2}^{M-1} t_q} \mathcal{M}(\mathbf{x}, \bar{\pi}) \quad (3)$$

where

$$\mathbf{x} = \left[j - \sum_{i=2}^M t_i, t_2, t_3, \dots, t_M, N-j \right] \quad (4)$$

$$\pi_i = \begin{cases} \frac{1-FER_i}{M} & 1 \leq i \leq M \\ 1 - \sum_{i=1}^M \pi_i & i = M+1 \end{cases} \quad (5)$$

$$\mathcal{M}(\mathbf{x}, \bar{\pi}) = \frac{N!}{\prod_{i=1}^{M+1} x_i!} \prod_{j=1}^{M+1} \pi_j^{x_j} \quad (6)$$

By combining (1) and (3) we can find the probability of successfully decoding a block of k packets when the overall number of transmitted packets was N , as can be seen below

$$\mathbb{P}_{dc}(k, N) = \sum_{j=k}^N \xi_q(k, j) \cdot \mathbb{P}_{rx}(j, N) \quad (7)$$

C. Throughput

The next step is to obtain the corresponding system throughput, based on the exact probability previously derived. First, we obtain the average number of transmissions that are required so as to decode a block of k packets, $E[N]$. We start by deriving the corresponding probability density function $p_{dc}(k, N)$, which can be computed as $p_{dc}(k, N) = \mathbb{P}_{dc}(k, N) - \mathbb{P}_{dc}(k, N-1)$. Afterwards $E[N]$ can thus be easily calculated as shown below.

Note that $p_{dc}(k, i) = 0$ when $i < k$.

$$E[N] = \sum_{i=k}^{\infty} i \cdot p_{dc}(k, i) \quad (8)$$

As $E[N] \geq k$, we can establish the average excess packet ratio, since it will have a negative impact on the achieved throughput.

$$\epsilon = \frac{E[N] - k}{E[N]} \quad (9)$$

Another aspect that may jeopardise the performance is the transmission of the acknowledgement packet by the receiver when it successfully decodes a block. If we consider that τ is the average delay of a data packet, and τ_{ACK} that corresponding to an ACK, the penalization factor induced by the acknowledgement packets, ϵ_{ACK} , can be estimated as follows

$$\epsilon_{ACK} = \frac{\tau_{ACK}}{E[N] \cdot \tau + \tau_{ACK}} \quad (10)$$

Finally, the overall throughput (S) can be calculated using (11), where S_{\max} is the throughput under saturation conditions, considering ideal channels. We use the Bianchi model [2] to establish the value of S_{\max} as a function of the active stations (i.e. number of sources, M).

$$S = S_{\max} \cdot (1 - \epsilon) \cdot (1 - \epsilon_{ACK}) \quad (11)$$

D. Context-aware probabilistic transmission

As was mentioned earlier, we assume that all the wireless links between the M sources and the destination share the same channel, and that a contention-based MAC protocol (for instance IEEE 802.11) is used. Under these circumstances, assuming saturation conditions, we could state that a *naive* scheme would equally share the transmission opportunities between the nodes (i.e by allowing, on average, the transmission of one packet every M attempts). However, this might be not the best strategy when link qualities are different, since it would be more sensible to allocate more transmission opportunities to those sources having better links. We therefore propose a context-aware probabilistic transmission scheme, by means of which the transmission rate of each source is modulated according to a certain probability, λ_i , which would favor those links with a lower FER (Frame Error Rate). We use the expression shown in (12), where the γ parameter can be seen as an aggressiveness factor: as γ gets higher, more transmission opportunities will be granted to those links exhibiting a better quality.

$$\lambda_i = \frac{1}{FER_i^\gamma} \frac{1}{\sum_{j=1}^M \frac{1}{FER_j^\gamma}} \quad (12)$$

This scheme, where each source transmits with a certain probability, also follows the previous theoretical model. The only modification required is to replace the expression (5) for the probability of a particular transmission π_i by

$$\pi_i = \begin{cases} \lambda_i \cdot (1 - FER_i) & 1 \leq i \leq M \\ 1 - \sum_{i=1}^M \pi_i & i = M + 1 \end{cases} \quad (13)$$

Note that if all wireless channels exhibit the same quality, $\lambda_i = \frac{1}{M}$; $\forall i \leq M$, and (13) simplifies to (5).

III. RESULTS AND DISCUSSION

As was already discussed, our baseline scenario considers multiple sources transmitting the same information to a single destination. Each transmitter splits the information into blocks of k packets, and sends random linear combinations of them until the receiver is able to decode the whole block, confirming its correct reception with the delivery of a backwards acknowledgement (broadcast).

In order to evaluate the accuracy of the proposed theoretical model, we have carried out an extensive simulation campaign over the ns-3 simulator [4], which was modified to integrate the entities required to carry out the coding and decoding procedures. We used IEEE 802.11b links (working at 11 Mbps) and we disabled the MAC retransmission scheme, since it has been shown [5] that it does not provide any benefit to the RLC protocol performance.

The authors in [5] studied the most appropriate values for the block size parameter, k . Based on this analysis, we will fix the block size to $k = 32$ packets. Furthermore, and for the sake of simplicity, we will mostly work with $GF(2)$. In any case, the theoretical model is also valid for different q values (as will be seen later), but the additional gain does not justify the increased computational burden (especially the one required by the decoding procedure).

In a first set of experiments we fixed the number of transmitters to two. The destination shall receive a file of 1000 blocks (each of them having k packets, whose length is 1500 Bytes). Figure 1 shows the cumulative distribution function (cdf) of the number of packets that need to be transmitted (by the two sources) before the destination is able to decode the k original packets of a single block. We use different frame error rates for the two wireless channels (i.e FER_1 and FER_2) and we analyze the results of the two transmission schemes (naive and probabilistic) that were previously presented. For all the different configurations we plot the simulation results (shown by markers) together with the theoretical values. In all cases, there is an almost perfect match, yielding the validity of the proposed model. We can also see that the probabilistic scheme equals the performance that is observed over an ideal channel, while the number of excess packets for the naive transmission scheme increases as the the quality of the wireless links get worse. Finally, the results also show the impact of receiving linear combinations of previous packets, since even in the case of ideal conditions, the number of excess packets is larger than k with a probability of ≈ 0.7 .

Once we have assessed the validity of the theoretical model, Figures 2 and 3 show the performance when modifying the quality of the two wireless links. In the first one (Figure 2), we increase the FER of the two channels, which exhibit the same quality. In this case, the performance of the probabilistic scheme would be symmetric, and we therefore only plot the

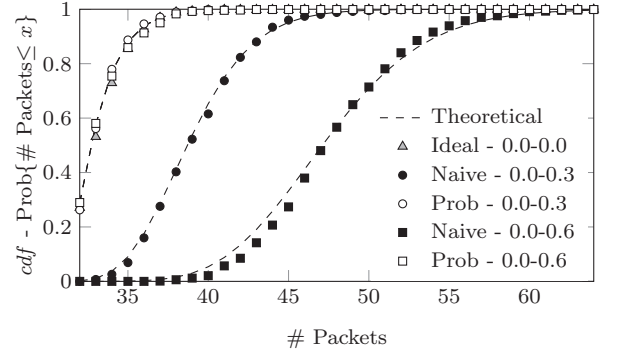
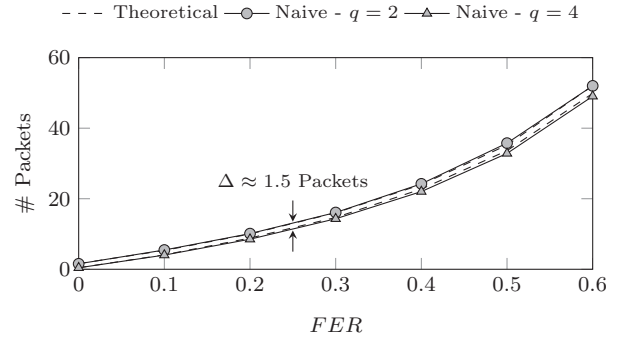
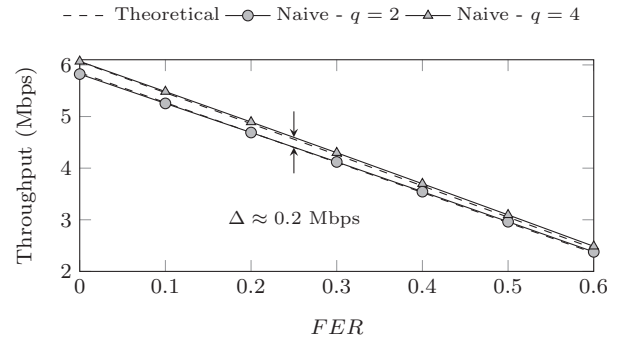


Fig. 1. cdf of the number of packets that are required to decode a block



(a) Excess Packets



(b) Throughput

Fig. 2. Excess packets and throughput, symmetric channels (i.e $FER_1 = FER_2$)

results obtained with the naive model. The figure validates the accuracy of the theoretical model, both for the number of excess packets and the throughput. In addition, we also studied the influence of the parameter q . As can be seen, there is a rather small benefit (in terms of both the number of excess packets and the throughput) when using $GF(2^2)$; studies carried out with higher order fields did not show any additional gain in further increasing q . The observed results also validate the correctness of the theoretical model, which perfectly matches those obtained with the simulator, for the two q values. Afterwards we fixed the FER of one of the channels to 0.3 and we increased the other one from 0 to 0.6 (Figure 3). In this case, we analyze the behavior of the probabilistic transmission scheme, studying the impact of the

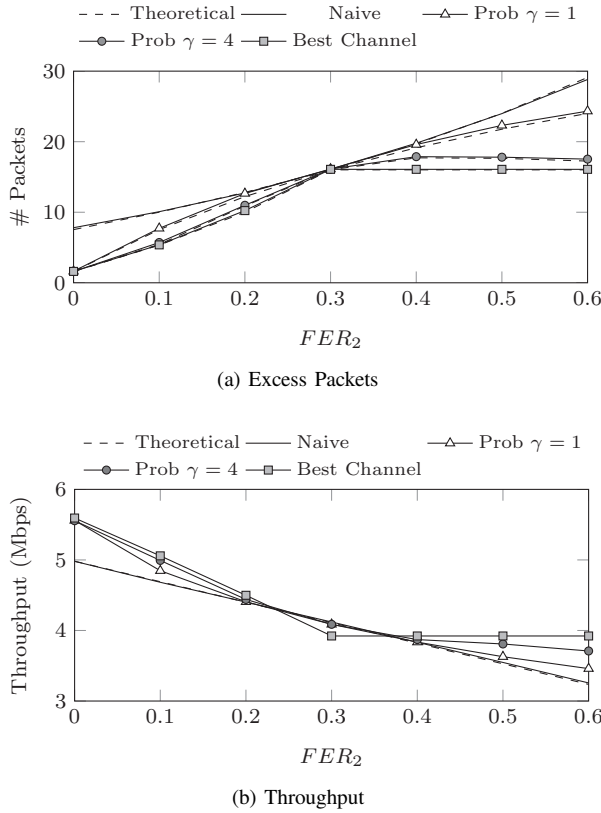


Fig. 3. Excess packets and throughput, asymmetric channels ($FER_1 = 0.3$)

γ parameter, by representing the performance that would have been achieved if only the best channel had been used; we can see that the behavior of $\gamma = 4$ is closer to the one exhibited by this *best channel* configuration. We can again observe that the theoretical model matches the simulation results. Regarding the throughput we can see that the probabilistic scheme, with $\gamma = 4$, obtains almost the same performance as the *best channel* configuration.

In the final group of experiments, we increase the number of sources to 4 and 8, and we carry out a Montecarlo process (comprising 1000 independent runs), randomly selecting the FER of the corresponding wireless channels within the range $0.0 \dots 0.6$. Figure 4 shows the throughput cdf. We have added a new configuration (*Blind*) in which the same link is always used (whatever FER it has). The naive scheme outperforms the *Blind* approach in 60% of the cases when $N = 4$ and in 40% of cases when N equals 8. This proves that the performance of the naive scheme is heavily jeopardized by the larger number of contending stations. On the other hand, the probabilistic scheme throughput is always higher than this baseline case and, moreover, the impact of the γ parameter is more clear. When $\gamma = 4$ the throughput is almost the same as the one that would have been achieved if the best channel had been always used and its influence is stronger for the scenario having more sources.

IV. CONCLUSIONS AND FUTURE WORK

We have proposed a theoretical model to compute the exact probability that a receiver obtains k linearly independent pack-

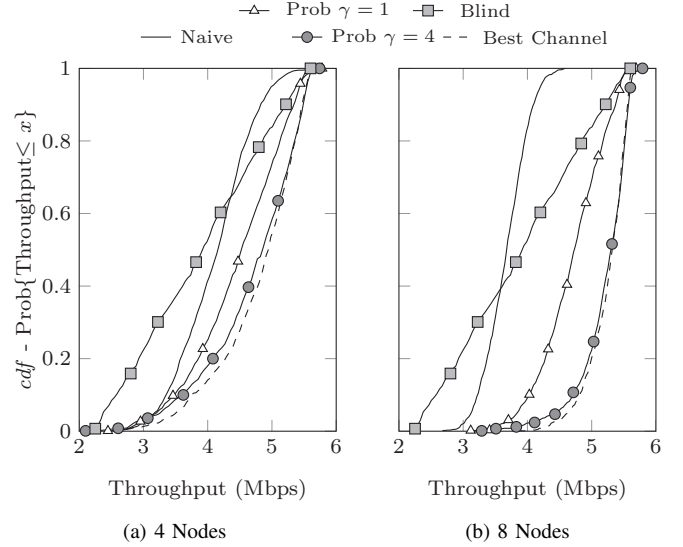


Fig. 4. Throughput cdf for the different transmission configurations

ets when multiple sources send the same content over error-prone wireless links. Based on the average number of excess packets and the Bianchi model we have as well derived an expression to obtain the overall throughput. A *context-aware* probabilistic transmission scheme that clearly outperforms the naive solution has been as well proposed. The validity of the model has been assessed by means of an extensive simulation study carried out over the ns-3 platform.

In our future work we plan to extend the analysis to more complex network topologies (i.e. comprising more than one single wireless link between the sources and the destination), having different characteristics, both in terms of error rate and delay. This study will also embrace the impact of the routing protocols (for instance, their overhead) that would be required in these scenarios. In this paper we have assumed that all sources are synchronised and therefore send the same block; we will study the influence of considering scenarios where each of the transmitters could send different blocks. This will be as well extended so as to consider multiple destinations, using scenarios similar to those studied by Firooz et al. [6].

REFERENCES

- [1] O. Trullols-Cruces, J. Barcelo-Ordinas, and M. Fiore, "Exact decoding probability under random linear network coding," *Communications Letters, IEEE*, vol. 15, no. 1, pp. 67–69, 2011.
- [2] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 3, pp. 535–547, March 2000.
- [3] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, ser. SIGCOMM '07. New York, NY, USA: ACM, 2007, pp. 169–180. [Online]. Available: <http://doi.acm.org/10.1145/1282380.1282400>
- [4] "The ns-3 network simulator," <http://www.nsnam.org/>.
- [5] D. Gómez, E. Rodríguez, R. Agüero, and L. Muñoz, "Reliable Communications over Lossy Wireless Channels by means of the Combination of UDP and Random Linear Coding," in *Computers and Communications (ISCC), IEEE Symposium on*, June 2014.
- [6] M. H. Firooz and S. Roy, "Data dissemination in wireless networks with network coding," *Communications Letters, IEEE*, vol. 17, no. 5, pp. 944–947, May 2013.