



**HAL**  
open science

# A Probabilistic Retransmission Protocol on a Relaying Network

Faton Maliqi, Francesca Bassi, Pierre Duhamel, Ilir Limani

► **To cite this version:**

Faton Maliqi, Francesca Bassi, Pierre Duhamel, Ilir Limani. A Probabilistic Retransmission Protocol on a Relaying Network. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Oct 2017, Montreal, Canada. hal-01699365

**HAL Id: hal-01699365**

**<https://centralesupelec.hal.science/hal-01699365v1>**

Submitted on 2 Feb 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Probabilistic Retransmission Protocol on a Relaying Network

Faton Maliqi<sup>†‡</sup>, Francesca Bassi<sup>†§</sup>, Pierre Duhamel<sup>†</sup>, Ilir Limani<sup>‡</sup>

<sup>†</sup> L2S – CNRS-CentraleSupélec-Université Paris-Sud, Gif-sur-Yvette, France

<sup>§</sup>ESME-Sudria, Ivry-sur-Seine, France

<sup>‡</sup> University of Prishtina, Faculty of Electrical and Computer Engineering, Prishtina, Kosovo

**Abstract**—Relaying protocols are usually deterministic, i.e. they commute from one action to the next one in a predefined manner, depending on the success of the transmission. This paper demonstrates the advantages that probabilistic protocols (i.e. the next action to be undertaken depends on some probability) may have in this context. The whole study is performed on a simple source-relay-destination network, the relay working in demodulate and forward mode, in order to obtain insights on the interaction between relaying and HARQ. So far, probabilistic protocols have been mainly proposed for higher layers of communication systems, but are applied here to physical and MAC layers. Since it contains only two parameters, we demonstrate that our probabilistic protocol can easily be tuned for best performance using a Finite State Markov Chain (FSMC), and that it brings improvement over deterministic protocols in the same scenario.

## I. INTRODUCTION

Diversity helps to mitigate the effects of fading, interference, pathloss, etc. of wireless channels, and can be obtained via several means. Cooperative diversity [1] uses Relays as alternative nodes for retransmitting the same signal to the destination, while time diversity can be obtained via an Automatic Repeat reQuest (ARQ) protocol. This ARQ protocol can also be combined with channel coding, which is known as Hybrid ARQ (HARQ) [2]. This paper follows these lines, and discusses the interaction between Relaying and HARQ. Usually, this interaction has been studied using deterministic protocols, like in [3], and in our analysis we focus on probabilistic protocols.

This interaction has also been studied from various perspectives, like in [4] (energy efficiency) or like in [5] (information theory). In most of these studies, the Relay is considered to work either on the Decode-and-Forward (DCF) mode, which is complex to implement and requires computational resources at the Relay, or on the Amplify-and-Forward (AF) mode, where both the useful signal and the noise are amplified. In contrast, in our analysis we consider the Relay to work on the Demodulate-and-Forward (DMF) mode [6], which corresponds to a very simple implementation.

So far, probabilistic protocols, where the action to be performed next (which node will retransmit the information) is chosen with a given probability, have been mainly proposed for higher layers of communication systems, like in [7] or in [8]. In contrast, this paper studies a probabilistic protocol implemented at physical and MAC layer. With this probabilistic

protocol, the actions from the nodes depend on random events, and the retransmitting node is chosen probabilistically. This protocol gives a good insight on the performance analysis, as it contains very few parameters (only 2) that can be optimized for best performance. Note that these parameters can be computed to mimic the behavior of a given deterministic protocol (not shown in the paper due to lack of room), and that the result of the optimization can only improve over this one. In this paper, the performance of our optimized probabilistic algorithm is checked against results of the literature [3].

Section II introduces the probabilistic protocol and its analysis using Finite State Markov Chains (FSMC), which allows performance evaluation and optimization. Section III provides numerical results, where the simulations are used to validate the theoretical predictions. Finally, Section IV draws the conclusions.

## II. THE PROBABILISTIC PROTOCOL

### A. Definition of the probabilistic protocol

Consider a cooperative network with one Source (S), one Relay (R) and one Destination (D), as depicted in Figure 1. The data is transmitted in information Packet Data Units (PDUs). Each PDU has a fixed length, and contains the information bits to which a Cyclic Redundancy Check (CRC) sequence [9] is appended. The PDU is channel encoded, and the coded packet is transmitted multiple times according to the HARQ technique. On each received packet, D responds with control messages which are heard by both S and R via an ideal feed-back channel. We consider Type II HARQ, where D exploits all copies of the coded packet received so far to decode the current PDU. More specifically, S and R always transmit the same coded packet, and D combines all received copies by Chase Combining [10]. The Relay works on the Demodulate-and-Forward (DMF) mode, chosen for its very simple implementation which does not require heavy computations at R.

At the beginning of the protocol a new PDU is broadcast by S for the first time, and received by both D and R. D decodes the packet, uses the CRC sequence to detect possible errors and sends a control message (ACK for positive and NACK for negative outcome), which is received by both S and R. If D broadcasts the ACK message S proceeds with the transmission of a new information PDU. If D broadcasts

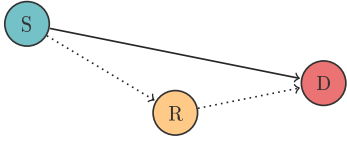


Fig. 1. S-R-D network

the NACK message the protocol enters the retransmission phase. The retransmission phase is characterized by two random variables,  $V_S$  and  $V_R$ , whose value determine if the corresponding nodes will be transmitting or not. The retransmission phase is governed by the following rules: **(i)** The first retransmission after any (re-)transmission by S is performed by R; **(ii)** if R is retransmitting, the next action is determined by drawing  $(V_R, V_S)$ : for  $V_R = 1$ , R is allowed to retransmit in the next time slot (probability  $(1 - \gamma)$ ); for  $(V_R, V_S) = (0, 1)$ , S is allowed to retransmit in the next time-slot (probability  $\gamma(1 - \beta)$ ); for  $(V_R, V_S) = (0, 0)$ , neither R or S are allowed to retransmit in the next time slot (probability  $\gamma \cdot \beta$ ); **(iii)** if neither R and S are allowed to retransmit, the current PDU is lost, and S transmits a new PDU for the first time. Variable  $V_R$  is Bernoulli distributed, with parameter  $\gamma$ . Variable  $(V_S|V_R = 0)$  is Bernoulli distributed, with parameter  $\beta$ , and variable  $(V_S|V_R = 1) = 0$  with probability 1.

The Source-Destination (S-D), Source-Relay (S-R) and Relay-Destination (R-D) channels are assumed to be block-fading channels impaired by Gaussian noise of variance  $N_0$ . For simplicity, all transmitting nodes are assumed to use the same energy per symbol,  $E_s$ , the same channel code and the same modulation scheme. Let  $d_{XY}$  denote the distance between nodes X and Y. The path-loss factor between nodes X and Y is  $l(d_{XY}) = d_{XY}^{-\alpha}$ , with  $\alpha$  the path-loss exponent. As a result, the channels S-D and S-R in time-slot  $t$  are modeled as:

$$\mathbf{y}_{SD,t} = \sqrt{E_s l(d_{SD})} \mathbf{h}_{SD,t} \mathbf{x}_t + \mathbf{w}_{SD,t} \quad (1)$$

$$\mathbf{y}_{SR,t} = \sqrt{E_s l(d_{SR})} \mathbf{h}_{SR,t} \mathbf{x}_t + \mathbf{w}_{SR,t} \quad (2)$$

where  $\sqrt{E_s} \mathbf{x}_t$  is the vector of modulated symbols with average energy  $E_s$ ;  $\mathbf{w}_{XY,t} \sim \mathcal{CN}(0, N_0)$  is the vector of white complex noise;  $\mathbf{h}_{XY,t} \sim \mathcal{CN}(0, 1)$  is the vector of Rayleigh fading coefficients. The channel R-D for time-slot  $t$  reads:

$$\mathbf{y}_{RD,t} = \sqrt{E_s l(d_{RD})} \mathbf{h}_{RD,t} \tilde{\mathbf{x}}_t + \mathbf{w}_{RD,t} \quad (3)$$

where  $\tilde{\mathbf{x}}_t$  is the vector of modulated symbols that R obtains performing hard decision on  $\mathbf{y}_{SR,t}$ .

### B. Probabilistic protocol analysis using Markov Chains

This section describes the probabilistic protocol, in a compact manner allowing an easy procedure for tuning parameters  $\gamma$  and  $\beta$  to maximize given performance criteria.

The probabilistic protocol can be described using a countable-state Markov Chain (MC). The state of the MC determines the action (R or S transmitting) that is going to take place during the current time-slot; the outcome of the action (ACK or NACK control message), along with the drawing of the random variables  $V_R$  and  $V_S$ , determines the transition to

the next state. The transition probability from State  $i$  to State  $j$  does not depend on time, but only on the pair  $(i, j)$ , to ensure the Markov property.

In the case of Type II decoding the probability to observe ACK or NACK depends on the number and order of copies received by D since the beginning of the transmission of the current PDU. This information needs to be included in the definition of the states, to preserve the Markov property. Moreover, due to the probabilistic nature of the protocol, the total number of transmissions can be infinite. This would result in an infinite number of states in the Markov Chain.

In order to obtain a more compact, albeit approximated, representation of the probabilistic protocol, we turn to a variation of the decoder which allows to represent the protocol using a finite number of states. The associated Finite State MC (FSMC) will be used for parameter optimization. Before entering the details of the analysis, note that the actual implementation of the probabilistic protocol at transmitter is very simple, and that the algorithm at destination is a classical Type II HARQ decoding algorithm. Only the intermediate steps to obtain a tractable FSMC model may look slightly involved.

### C. The probabilistic protocol with limited buffer

Consider the probabilistic protocol and add the constraint that the decoder can process at most  $C_{\max}$  received packets simultaneously. The probabilistic protocol at the transmitter does not change. At the destination, the received copies are stored in a buffer. If the buffer is filled to capacity  $C_{\max}$  and a new copy is received, this new copy overwrites one item in the buffer. We impose the rule that a packet from S can overwrite packets from S only, and a packet from R can overwrite packets from R only. The packet to be replaced is chosen among the candidates as the one with the worst quality. The quality of a received packet is defined by a measure of the belief that hard demodulation of the sequence would produce the correct sequence of coded bits. The details are omitted here due to lack of space.

The probabilistic protocol with limited buffer at D can be described with a FSMC. The definition of the states is given in Figure 2 for  $C_{\max} = 4$ . The states are defined according to the values of **(i)** the parameter CO (Copy Order), representing the order with which the buffer has been filled before the current time-slot; **(ii)** the control parameters  $(V_R, V_S)$  regulating the next access to the channel; **(iii)** the latest control message  $W$  issued by D. Both States 0 and 1 correspond to the first transmission of a new PDU; they allow to distinguish a first transmission after the last PDU has been ACK-ed (State 0) or after a drop (State 1). This allows an easy computation of the PDU error rate. Each state is associated with one of the following actions: ST (Source Transmits), RRT (Relay Retransmits) or SRT (Source Retransmits). The reason for including the parameter CO in the definition of the state is related to the DMF mode of the relay. Since R retransmits the most recently received message without checking for its integrity, the average probability of ACK at D depends not only on the number of copies from R, but also on how often

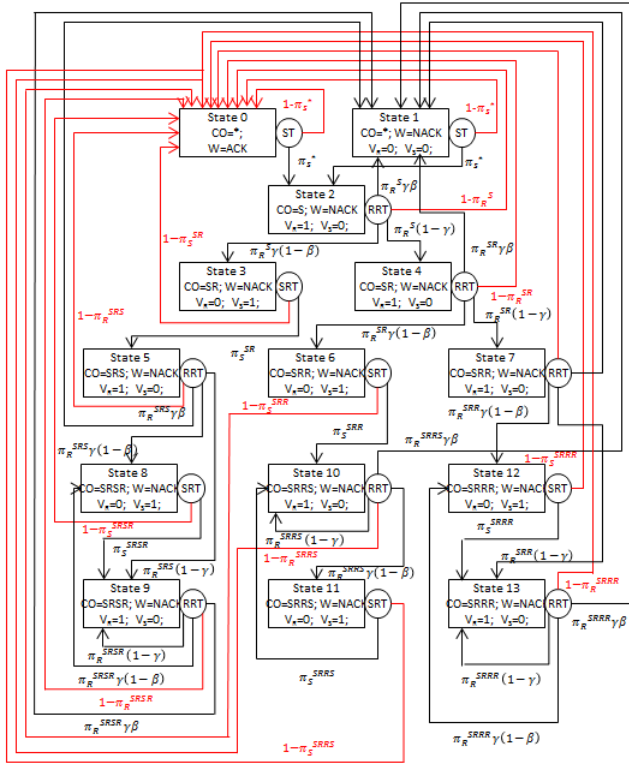


Fig. 2. FSMC representing the probabilistic protocol with limited buffer  $C_{\max} = 4$

the message at R has been regenerated. The number of states increases with the size of the buffer  $C_{\max}$ . For size  $C_{\max} = 4$  the number of states is 14, for  $C_{\max} = 5$  the number of states is 23 and for  $C_{\max} = 6$  the number of states is 37.

The state transitions of the FSMC are depicted on Figure 2, along with their probabilities. Let  $(1 - \pi_S^{\text{co}})$  be the probability of successful decoding at D when S is transmitting, and the buffer has configuration  $\text{CO}=\text{co}$ . Define  $\pi_R^{\text{co}}$  similarly.  $P$ , the transition matrix of the FSMC with states defined as in Figure 2 has the following form:

$$P = \begin{pmatrix} (1 - \pi_S^*) & 0 & \pi_S^* & \cdots & \cdots \\ (1 - \pi_S^*) & 0 & \pi_S^* & \cdots & \cdots \\ (1 - \pi_R^S) & \gamma \cdot \beta \pi_R^S & \gamma(1-\beta) \pi_R^S & \cdots & \cdots \\ (1 - \pi_R^{\text{SR}}) & \gamma \cdot \beta \pi_R^{\text{SR}} & 0 & \cdots & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (1 - \pi_R^{\text{SRRR}}) & \gamma \cdot \beta \pi_R^{\text{SRRR}} & 0 & \cdots & \cdots \end{pmatrix}. \quad (4)$$

Probabilities  $(1 - \pi_R^{\text{co}})$  and  $(1 - \pi_S^{\text{co}})$  do not depend on  $\gamma$  and  $\beta$ , when the filling pattern  $\text{co}$  is shorter than  $C_{\max}$ . This does not hold when the filling pattern  $\text{co}$  has length  $C_{\max}$ , and the current transmission results in overwriting in the buffer. Let  $\mathbf{X}_R$  be the sequence of past overwriting of copies from R in the buffer, and let  $q_R(\mathbf{x})$  represent the probability  $\mathbb{P}(\mathbf{X}_R = \mathbf{x})$ . For  $\text{co}$  of length  $C_{\max}$  the probability  $(1 - \pi_R^{\text{co}})$  is obtained by

averaging over all possible past overwriting sequences:

$$(1 - \pi_R^{\text{co}}) = \sum_{\mathbf{x}} ((1 - \pi_R^{\text{co}}) | \mathbf{X}_R = \mathbf{x}) q_R(\mathbf{x}). \quad (5)$$

Note that the probabilities  $q_R(\mathbf{x})$  are functions of parameters  $\beta$  and  $\gamma$ . In order to eliminate this statistical dependence, we choose to approximate (5) as follows

$$(1 - \pi_R^{\text{co}}) \approx ((1 - \pi_R^{\text{co}}) | \mathbf{X}_R = \emptyset), \quad (6)$$

where  $\mathbf{X}_R = \emptyset$  means that the current transmission from R will result in the first overwriting of the buffer. With this approximation, probabilities  $\pi_S^{\text{co}}$  and  $\pi_R^{\text{co}}$  in (4) do not depend on  $\gamma$  or  $\beta$ .

#### D. Performance evaluation using the FSMC

We evaluate the following metrics: the PDU Error Rate (PER), i.e. the proportion of PDUs that are dropped without being ACKed; and the average number  $\bar{T}$  of transmissions per PDU. Increasing  $\bar{T}$  allows decreasing PER. The trade-off between these quantities is expressed by the goodput  $G$ , the average number of ACKed PDUs per time-slot  $G = \frac{1 - \text{PER}}{\bar{T}}$ .

The FSMC representation allows an easy evaluation of the performance of the limited-buffer protocol. The steady state vector  $\mathbf{p}$  of the FSMC is the eigenvector associated to eigenvalue  $\lambda = 1$  of the transition matrix  $P$ . The steady state probability  $p_k$  represents the fraction of time that the FSMC spends in state  $k$  [11]. Observing that State 0 is visited only in case of correct acknowledgment of an information PDU, and that State 1 is visited only if a NACK-ed packet is dropped, the performance metrics can be evaluated as functions of  $\mathbf{p}$  as

$$\text{PER} = \frac{p_1}{p_0 + p_1}, \quad \bar{T} = \frac{1}{p_0 + p_1}, \quad G = p_0. \quad (7)$$

#### E. Optimization of the probabilistic protocol parameters

We turn to the problem of finding the values of  $\gamma$  and  $\beta$  that maximize the performance of the probabilistic protocol. Since the PER and the average number of transmissions  $\bar{T}$  are conflicting requirements, we define the optimization as the PER minimization, constrained to a maximum value of  $\bar{T}$ :

$$\gamma^*, \beta^* = \arg \min_{\gamma, \beta : \bar{T} \leq \bar{T}_{\max}} \text{PER}. \quad (8)$$

Depending on the propagation quality of the physical channels, we can identify two regimes:

**(1) Bad regime:** PDUs are seldom successfully decoded with very few transmissions. Decreasing the probability  $\gamma \cdot \beta$  of dropping a PDU decreases the PER and increases  $\bar{T}$  (since PDUs are likely to be retransmitted many times). In this regime varying  $\gamma$  and  $\beta$  affects the trade-off between PER and  $\bar{T}$ .

**(2) Good regime:** it is a very common event that a PDU is successfully decoded with very few transmissions. Decreasing the probability  $\gamma \cdot \beta$  of dropping a PDU decreases PER without increasing  $\bar{T}$  (since few PDUs are likely to be transmitted more times). In this regime there is no trade-off to be achieved between the PER and  $\bar{T}$ .

Section III-A illustrates the two regimes numerically, while the optimization (8) is numerically solved in Section III-B.

### III. NUMERICAL RESULTS

The performance of the probabilistic protocol can be obtained via Monte Carlo simulations. Its approximate prediction can be obtained using the FSMC analysis of the protocol with limited buffer decoder. For this, as detailed in Section II-D, we need to evaluate the steady-state vector of the transition matrix  $P$ . The parameters  $\pi_S^{\text{co}}$  and  $\pi_R^{\text{co}}$ , involved in matrix  $P$  depend only on the propagation conditions on the channels S-D, S-R and R-D, and can be evaluated by Monte-Carlo simulation.

In the following we consider Convolutional Coding with code rate  $R_c = 1/3$ , and 16-QAM modulation. The information PDU is 1000 bits long. The average energy per modulated symbol is  $E_s = 1$ , and the path-loss exponent is  $\alpha = 2.4$ . The transmit  $E_b/N_0$  is the same on the channels S-D and R-D. The decoder performs Chase Combining. We consider various positions of R with respect to D and S.

#### A. Approximate performance prediction using FSMC

First, we check that the FSMC analysis is accurate in predicting the performance of the probabilistic protocol. Consider  $d_{\text{SR}}/d_{\text{SD}} = 0.5$ . Each curve in Figure 3 depicts the various trade-offs between the PER and  $\bar{T}$ , obtained by varying  $\beta$  when  $\gamma = 1$  (in this case, R never retransmits twice in a row). The curves are in pairs: each red, solid curve is the simulation of the probabilistic protocol, and its corresponding blue, dashed curve is its prediction obtained with the FSMC analysis. Different curve pairs represent different values of transmit  $E_b/N_0$  on the channel S-D. The prediction of the performance via the FSMC analysis is accurate, except for high levels of noise on the channels. This implies that the FSMC analysis can be successfully employed to build the optimization procedure of  $\gamma$  and  $\beta$  in the probabilistic protocol.

Figure 3 confirms the existence of the two regimes described in Section II-E. The markers on a given curve correspond to the tested values of  $\beta$ . Recall that a smaller  $\beta$  implies a smaller probability to drop the current PDU. As expected, PER decreases with  $\beta$ , at the expense of increasing  $\bar{T}$ . However, after a critical value of PER (roughly  $10^{-2}$ ) further decreasing  $\beta$  allows to improve PER, without affecting  $\bar{T}$ . The value of  $\beta$  at which this critical value of PER is met varies for different values of the transmit  $E_b/N_0$  on the channel S-D.

#### B. Optimization of the probabilistic protocol

Since the FSMC analysis provides a good prediction of the performance of the probabilistic protocol, it is used to numerically solve the optimization problem of Section II-E. Given the propagation conditions on the channels S-R, S-D and R-D it is easy to get, via Monte-Carlo simulation, the probabilities  $\pi_S^{\text{co}}$  and  $\pi_R^{\text{co}}$  necessary to build the transition matrix  $P$  defined in (4). Then, the FSMC analysis is used to predict the performance for a wide range of values of  $(\gamma, \beta)$ . The best pair according to (8) is selected. Since the number of states in the FSMC is small, this procedure has a very small computational cost, even if the set to be explored is large.

Consider the case of  $d_{\text{SR}}/d_{\text{SD}} = 0.5$ , for  $\bar{T}_{\text{max}} = 2.5$ . The following situations are considered: **(i)**  $\beta = 1.0$ . This implies

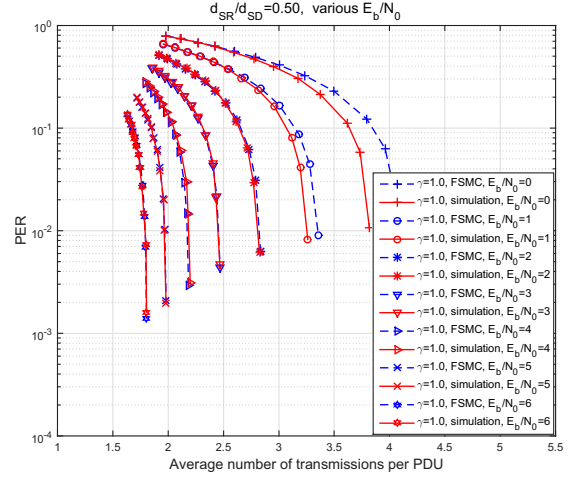


Fig. 3. Trade-off between PER and  $\bar{T}$ ,  $\gamma = 1$ ,  $d_{\text{SR}}/d_{\text{SD}} = 0.5$

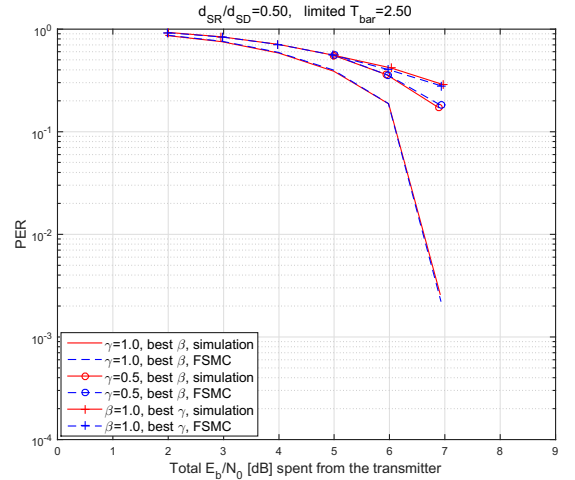


Fig. 4. PER optimization,  $d_{\text{SR}}/d_{\text{SD}} = 0.5$ ,  $\bar{T}_{\text{max}} = 2.5$

that S is not allowed to retransmit (it performs only the first transmission), and all retransmissions are performed by R. For each transmit  $E_b/N_0$  on channel S-D we select the optimum  $\gamma$ ; **(ii)**  $\gamma = 1.0$ . This implies that R never retransmits twice consecutively. For each transmit  $E_b/N_0$  we select the optimum  $\beta$ ; **(iii)**  $\gamma = 0.5$ . For each transmit  $E_b/N_0$  we select the optimum  $\beta$ . Figure 4 depicts the optimum PER obtained in the three cases, as a function of the total transmit  $E_b/N_0$ . The total transmit  $E_b/N_0$  considers the average energy per information bit spent by the protocol, including all retransmissions. The best performance is obtained for  $\gamma = 1$ , when R is not allowed to retransmit multiple times successively, and the worst is obtained for  $\beta = 1$ , when S never retransmits. This is due to the DMF mode of the relay: since R retransmits the most recently demodulated packet without checking its integrity, frequently refreshing data at R by allowing S to retransmit limits error propagation.

Now consider the impact of the location of R. Fig. 5 depicts the achievable PER for  $\bar{T}_{\text{max}} = 2.5$  as a function of the total transmit  $E_b/N_0$ , for  $d_{\text{SR}}/d_{\text{SD}} \in \{0.15, 0.35, 0.5, 0.85\}$ .



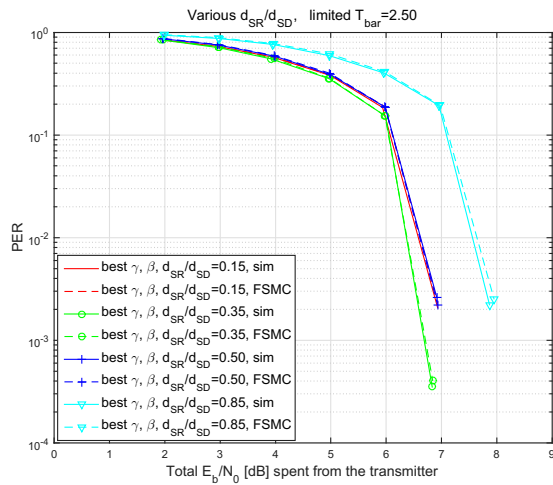


Fig. 5. Achievable PER,  $\bar{T}_{\max} = 2.5$ ,  $d_{SR}/d_{SD} \in \{0.15, 0.35, 0.5, 0.85\}$

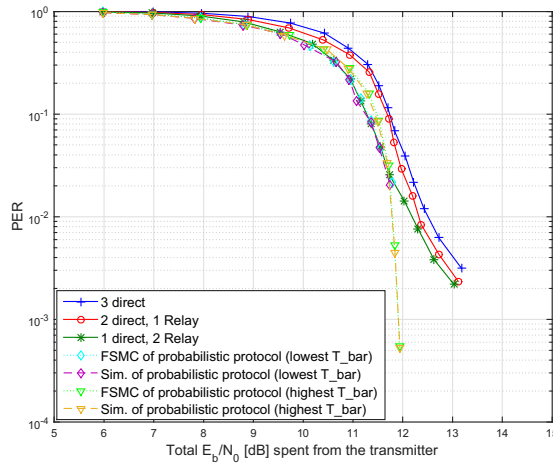


Fig. 6. Achievable PER with deterministic and probabilistic protocols

At any location of R the best performance is achieved for  $\gamma = 1$ , *i.e.* allowing R to refresh its own demodulated copy before each retransmission. The best performance is achieved for  $d_{SR}/d_{SD} = 0.35$ . This position achieves the best compromise between the necessity of correct demodulation at R and proximity of R to D. For  $d_{SR}/d_{SD} \ll 0.35$ , in fact, the probability of demodulation error at R is small, but the path loss on the channel R-D is strong. On the other hand for  $d_{SR}/d_{SD} \gg 0.35$  the path loss on the channel R-D is small, but the chance of relaying corrupted data is high. As before, this behavior is due to the DMF mode at R.

### C. Comparison with a deterministic protocol

Let us now compare the performance of the probabilistic protocol introduced in this paper with the performance of a deterministic protocol. We consider the protocol in [3] as a reference, and we adapt it to our S-R-D network. In particular, we consider DMF mode for the relay, while Amplify-and-Forward (AF) mode is considered in [3]. In [3] the total number of transmissions is limited to 3, and all possible

strategies are considered: all three transmissions by S, two transmissions by S and one by R, one transmission by S and two by R. The PER for each strategy is depicted in Fig. 6 as a function of the total transmit  $E_b/N_0$ , when  $d_{SR}/d_{SD} = 0.15$ .

To compare the probabilistic protocol with the deterministic protocol [3] we proceed as follows. Let  $\bar{T}_1$  be the highest value of  $\bar{T}$  obtained using the deterministic strategies; and let  $\bar{T}_2$  be the lowest. We solve the optimization problem (8) both for  $\bar{T}_{\max} = \bar{T}_1$  and  $\bar{T}_{\max} = \bar{T}_2$ , and we get the achievable PER for the probabilistic protocol. The values of  $\bar{T}_1$  and  $\bar{T}_2$  depend on the transmit  $E_b/N_0$  on the channel S-D, and the procedure needs to be repeated each time this parameter changes. The achievable PER of the probabilistic protocol under both constraints is compared with the PER of the deterministic protocol in Fig. 6. For  $\bar{T}_{\max} = \bar{T}_1$  the probabilistic protocol clearly outperforms the deterministic protocol. Moreover, even for  $\bar{T}_{\max} = \bar{T}_2$  the probabilistic protocol performs no worse than the best deterministic strategy.

## IV. CONCLUSIONS

In this paper we have proposed a retransmission protocol with HARQ technique, where the actions of each node on a S-R-D network depend on random parameters. We have shown that we can analyze this probabilistic protocol using FSMC and optimize very easily its performance. Beside the fact that Relay on DMF mode is very simple to implement in practice, this work shows that it can be also very beneficial to apply. Since this work is promising to networks with more nodes, we have addressed the dependence of the performance on relay location, so that in a case with many users we would know in advance which location to choose for cooperation.

## REFERENCES

- [1] A. Nosratinia, T. E. Hunter, and A. Hedayat, *Cooperative communications in wireless networks*, IEEE Communications Magazine, vol. 42, no. 10, pp. 74-80, 2004.
- [2] H. Anh Ngo, L. Hanzo, *Hybrid Automatic-Repeat-reQuest Systems for Cooperative Wireless Communications*, IEEE Communications Surveys & Tutorials, Year: 2014, Volume: 16, Issue: 1.
- [3] I. Krikididis, *Distributed truncated ARQ protocol for cooperative diversity networks*, IET Communications, IET Journals & Magazines, 2007.
- [4] I. Stanojevic, O. Simeone, Y. Bar-Ness, and D. H. Kim, *Energy efficiency of non-collaborative and collaborative Hybrid-ARQ protocols*, IEEE Transactions on Wireless Communications, vol. 8, no. 1, pp. 326-335, 2009.
- [5] M. S. Fazel Falavarjani, R. Hoshyari, and R. Tafazolli, *Performance Evaluation of A Flexible Amplify and Forward (AF) Combined with HARQ*, 21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Pages: 488-493, 2010.
- [6] Y. Yang, H. Hu, J. Xu, and G. Mao, *Relay technologies for wimax and lte-advanced mobile systems*, IEEE Communications Magazine, vol. 47, no. 10, pp. 100105, 2009.
- [7] A. Lindgren, A. Doria, and O. Scheln, *Probabilistic routing in intermittently connected networks*, ACM SIGMOBILE mobile computing and communications review, vol.7, no.3, pp.19-20, July 2003.
- [8] J. Fukuyama, *A Probabilistic Protocol for Multi-Hop Routing in VANETs*, ICC Workshops, 14-18 June 2009.
- [9] T. V. Ramabadran, S. S. Gaitonde, *A tutorial on CRC computations*, IEEE Journals & Magazines, Volume: 8, Issue: 4, Pages: 62-75, Year: 1988.
- [10] D. Chase, *Code combining: A maximum-likelihood decoding approach for combining an arbitrary number of noisy packets*, IEEE Transactions on Communications, Vol. Com-33, No.5, May 1985.
- [11] Robert G. Gallager, *Stochastic Processes, Theory for Applications*, Cambridge University Press, book chapter 4, February 2014.