SR-ARQ Delay Statistics on N-State Markov Channels with finite Round Trip Delay

Michele Rossi, Leonardo Badia, Michele Zorzi
Department of Engineering, University of Ferrara
Via Saragat 1, 44100 Ferrara, Italy, e-mail: *[mrossi|lbadia|mzorzi]@ing.unife.it*.

***Abstract*— In this paper the packet delay statistics of a fully reliable Selective Repeat (SR) ARQ scheme is investigated. A N-State Discrete Time Markov Channel model is used to describe the packet error process and the channel round trip delay is considered to be finite, i.e., ACK/NACK messages are received at the transmitted m channel slots after the packet transmission is started. The ARQ packet delay statistics is evaluated by means of an exact analysis by jointly tracking packets errors and channel state evolution. Furthermore, a procedure to derive a Markov Channel description of a Rayleigh fading process is presented and the delay statistics obtained from the Markovian analysis is compared with the ones estimated by simulation of the ARQ protocol over the actual fading process. Some discussions on the accuracy of the delay statistics obtained from the Markov Channel representation of the actual fading process is finally reported.**

I. Introduction

Multimedia applications in modern communication systems are highly sensitive to channel impairments and require effective error control techniques. Such techniques often rely on Forward Error Correction (FEC), Automatic Retransmission reQuest (ARQ), or a combination of the two. A trade-off opens up between data reliability, latency, and efficient bandwidth usage. In such scenarios, it is important to have a good understanding of the impact of error control strategies to provide adequate application level performance depending on the underlying channel state.

In this work, we focus on a fully reliable Selective Repeat (SR) ARQ [1]; our aim is to derive delay statistics so as to understand how the retransmission process introduced by the ARQ error recovery algorithm affects the packet delay experienced at higher layers. In SR ARQ, the transmitter sends packets (PDUs) in order of their arrival time at the link layer buffer, while the receiver replies to each received PDU with ACK/NACK messages by sending them over a feedback channel. The sender retransmits only negatively acknowledged (NACK) packets and then resumes the transmission process from the last packet sent so far. In this paper, we consider the statistics of the *delivery delay*, defined in the literature [1] as the sum of *transmission delay* and *re-sequencing delay*. These quantities are the delay between the first transmission and the correct reception of the PDU and the time spent in the receiver re-sequencing buffer for the packet to be released in-sequence at higher layers, respectively.

This problem has already been studied in the literature. Basically, the complexity of the analysis has lead to the introduction of several approximations [1][2][3] or to considering an indirect approach, i.e., to study the transmitter/receiver buffer occupancy [4][5][6]. In general, the assumption of a *static channel*, or other restrictive hypotheses are common, but these approaches only allow to derive mean values for the various delays involved in the ARQ transmission process.

To overcome these problems, in [7] we presented a study to obtain delivery delay statistics by deriving an exact analysis in the time-varying channel case. However, this work was developed under the assumption of a Two-State Markov packet error model, which somehow may limit the validity of the analysis. Here, we relax this assumption by allowing the packet error process to be described by means of a Markov chain with an arbitrary number of states. Note that, with respect to other analyses presented so far in the literature, we still derive full statistics for the delivery delay of a fully reliable SR ARQ scheme instead of mean values. Moreover, observe that the knowledge of such statistics allows to easily find the delay distributions of an aggregate of link layer PDUs as well [8] (i.e., of a higher layer packet.) This last statistics can be found by means of a convolution that does not affect the complexity of the analysis. The approach is presented in [8] and its extension to the N state channel model is straightforward.

The remaining part of the paper is organized as follows. In Section II, the SR-ARQ transmission process is described, whereas the Markov Channel model is described in Section III. In Section IV, the analysis for the evaluation of the ARQ delivery delay statistics over an N-State Markov Model is presented. In Section V we present a method to obtain an N-State Markov Model to represent a quantized Rayleigh fading channel while, in Section VI, the derived Markov Channel models are used to obtain the statistics for the SR-ARQ delivery process; this statistics is compared with the ones obtained by simulation of the actual Rayleigh fading channel. Finally, in Section VII some conclusions are given.

II. Model for ARQ queueing and transmission processes

Consider a transmitter and a receiver, transmitting data packets through a noisy wireless link. We assume that a fully reliable Link Layer protocol (unlimited retransmission attempts) is used to counteract channel impairments. It is also assumed that both nodes have unlimited buffer size.

The time is slotted, where the slot duration corresponds to the (constant) transmission time for a single packet. The transmitter sends data packets (ARQ PDUs) and the receiver replies with ACK or NACK messages to inform the sender about correctly received and erroneous packets, respectively. ACK/NACK messages are assumed to be error-free for simplicity (this hypothesis could be easily removed, see [9]). As long as ACKs are received, new PDUs are transmitted in numerical increasing order. The ACK message for a packet transmitted in the generic slot t is received after the transmission of up to $m-1$ PDUs (new or retransmitted), i.e., at the end of slot $t+m-1$. The round-trip delay m is commonly referred to in the literature [1] as the *ARQ window size*. In case of NACK, instead of a new one, the PDU transmitted m slots earlier is re-transmitted.

The delivery delay is only slightly affected by the traffic model. So, it is reasonable to consider a simple model for the arrival process, although our analysis can be extended, if necessary. Hence, we consider that once a PDU is correctly transmitted, a new one is always present in the source buffer. This assumption is called in the literature [1] *Heavy Traffic* condition, and describes exactly a continuous packet source. Thus, it is accurate, e.g., for a TCP file transfer (FTP-like session) or video/audio continuous data streaming. Reliable ARQ almost completely avoids TCP timeouts (when the channel error rate is not too large) and the TCP level, after filling the bandwidth-delay product, behaves as a continuous packet source (the TCP window size is not decreasing because error recovery is never triggered). Should the *Heavy Traffic* assumption not be verified, the delivery delay computed can be seen as an upper bound (worst case analysis). It is possible to relax this hypothesis by following an approach as in [5].

III. CHANNEL MODEL

Consider an N-State Discrete Time Markov Model [10][11], where the slot duration corresponds to the ARQ packet transmission time. Let $\mathcal{S} = \{0, 1, ..., K-1\}$ be the set of states composing the Markov Chain and t_{ij} the transition probability from state i to state j, i.e., the probability that the state in the next slot is j given that the state in the current one is i. Consider also that every state $k \in \mathcal{S}$ is characterized by a PDU error rate $\epsilon_k \in [0, 1]$, i.e., a PDU transmitted in state k is erroneous with probability ϵ_k. The Markov Chain is completely specified by the pair $(\mathbf{T}, \mathcal{E})$, where $\mathbf{T}$ is the transition probability matrix and $\mathcal{E}$ is the state error probability vector. To carry out the analysis of the ARQ protocol (Section IV) we extend this model to an equivalent one, composed by the $N = 2K$ states in $\mathcal{S}' = \{0, 1, \ldots, N-1\}$. This model is fully described by the transition probability matrix $\mathbf{P} = \{p\}_{ij}$, where the transition probabilities p_{ij}, $0 \le i, j \le N-1$ are derived as follows

$$p_{ij} = \begin{cases} t_{xy}(1-\epsilon_y) & j \in \{0, 1, \ldots, K-1\} \\ t_{xy}\epsilon_y & j \in \{K, K+1, \ldots, N-1\} \end{cases} \tag{1}$$

where $(x, y) = (i, j) - Ku[(i, j) - K]$, $0 \le x, y \le K-1$, $u[\cdot]$ is the unit step, i.e., $u[n] = 1$ if n is greater than or equal to 0, $u[n] = 0$ otherwise. Moreover, note that in the extended model states $\{0, 1, \ldots, K-1\}$ are error free, i.e., a PDU is always transmitted correctly in these states, whereas in states K through $N-1$ PDUs are always transmitted erroneously.

IV. COMPUTATION OF THE DELIVERY DELAY STATISTICS IN A N-STATE MARKOV CHANNEL

Our goal here is the computation of the delay statistics for a single PDU transmitted using Selective Repeat ARQ. In order to do so, we develop a model which tracks the successful delivery of the PDU of interest (called *tagged PDU*), as well as all previous PDUs. First of all, suppose the tagged PDU is transmitted for the first time in slot $t = m$. This implies that all previous PDUs (i.e., those whose identifier is smaller than the tagged PDU) excluding the $m-1$ PDUs transmitted in slots 1 through $m-1$ have been successfully received, and that in slot 0 a successful transmission occurred (otherwise in slot m we would have a retransmission). Therefore, the tagged PDU is finally released upon correct reception of all PDUs transmitted in slots 1 through m. On the other hand, all PDUs which are transmitted for the first time during slots $t > m$ must have a larger id than the tagged PDU, and therefore do not affect the delivery of the tagged PDU. We can then ignore all future PDU arrivals in our study.

The problem to be solved is therefore to find the time it takes for all PDUs transmitted in slots 1 through m to be eventually received correctly, given that a successful transmission occurred in slot 0. Consider the evolution of the system after slot m. If slot 1 contained an erroneous transmission, a retransmission will be scheduled in slot $m+1$. If this retransmission is successful, then slot $m+1$ will be marked as *resolved*, otherwise it will be marked as *unresolved* (with the effect of a further retransmission in slot $2m+1$ and so on until success). On the other hand, if slot 1 contained a successful transmission, then slot $m+1$ corresponds to a slot which was already resolved, and will be itself resolved (recall that for our purposes future arrivals are ignored, and therefore once a packet is successfully delivered the corresponding slot remains empty). In general, slot $m+k$ (with $k > 0$) will be marked as resolved if either slot k was itself resolved or the channel state in slot $m+k$ is good, and will remain unresolved otherwise. Since in case of transmission failure in an unresolved slot a packet is rescheduled for transmission m slots later, a suitable model to track the relevant events is one in which memory is kept about the resolved/unresolved status of the $m-1$ most recent past slots, i.e., at any given time t, we need to know the status of slots $t-m+1, t-m+2, \ldots, t-1$. A binary variable is therefore assigned to each slot to carry this information.

Considering t as the current slot, $b_k = 1$ if slot $t-m+1+k$ is still unresolved, and $b_k = 0$ otherwise, for $k = 0, 1, \ldots, m-2$. This string of bits keeps memory of which slots are yet to be resolved, and can also be represented by the integer $i = \sum_{k=0}^{m-2} b_k 2^k$. In addition, we need to specify the status of the current slot, i.e., slot t. In this case a binary variable is no longer sufficient, since we also need to track the channel state, which is necessary to determine the future evolution of successful transmissions. (Note that the Markovian nature of the channel evolution makes it possible to ignore the channel state in slots $t-m+1, t-m+2, \ldots, t-1$ once the channel state in t is known.) For the current slot, many situations are possible, in particular there are three main cases: the channel is in a *good* state, which implies that the slot is resolved (if it was not resolved already, the good channel state makes it resolved now); the channel is in a *bad* state and the slot is resolved (in a previous transmission); the channel state is *bad* and the slot is still unresolved. These three possibilities comprise ν, $N-\nu$ and $N-\nu$ states of the channel, respectively. Thus, these $2N-\nu$ states for the last PDU condition will be denoted in numerical order, i.e., by $\{0, 1, \ldots, \nu-1\}$, $\{\nu, \nu+1, \ldots, N-1\}$, $\{N, N+1, \ldots, 2N-\nu-1\}$, respectively. The associated variable will be denoted by ω. Consider now the random process $X(t) = (i(t), \omega(t))$ which jointly tracks slot-by-slot the Markov channel evolution and the status of the m latest slots. This process is a Markov chain. In order to determine the possible transitions and the corresponding transition probabilities, suppose at time t the bitmap which describes the slot status is $\boldsymbol{b} = (b_0, b_1, \ldots, b_{m-2})$, where the most significant bit b_{m-2} denotes the status of the most recent among the past slots. At time $t+1$ this bitmap is clocked one position into the past, i.e., $\boldsymbol{b}' = (b'_0, b'_1, \ldots, b'_{m-3}, b'_{m-2}) = (b_1, b_2, \ldots, b_{m-2}, f(\omega))$, where $f(\omega) = 1$ if $\omega \geq N$ (current slot at time t was still unresolved), and $f(\omega) = 0$ if $\omega < N$. More compactly, in this case $f(\omega) = u[\omega - N]$, where $u[\cdot]$ is the unit step. Regarding the value of $\omega' = \omega(t+1)$, note the following. If, at time t, $b_0 = 0$, the corresponding slot has already been resolved, and therefore $0 \leq \omega' \leq N-1$ according to the channel state y at time $t+1$ ($\omega' = y$). On the other hand, if $b_0 = 1$, the slot is still unresolved at time t, and therefore we have $0 \leq \omega' \leq \nu - 1$ if the channel at time $t+1$ is good (slot is resolved at this time) and $N \leq \omega' \leq 2N - \nu - 1$ otherwise (slot remains unresolved). In the former case, it is again $\omega' = y$, whereas in the latter $\omega' = y + N - \nu$. Note that given $X(t)$ there are only N possible destinations for $X(t+1)$, since the shift of the bitmap is deterministic and the only random variable is the channel state which can assume N values. More precisely, the transition probabilities are given as follows

- if i is even (i.e., $b_0 = 0$), then:

$$P[X(t+1) = (i', \omega') | X(t) = (i, \omega)] = \\ = \begin{cases} p_{xy} & \text{if } i' = \lfloor \frac{i}{2} \rfloor + u[\omega - N] 2^{m-2}, \\ & x = \omega - (N-\nu) u[\omega - N], \\ & \omega' = y, y = 0, 1, \ldots, N-1 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

- if i is odd (i.e., $b_0 = 1$), then:

$$P[X(t+1) = (i', \omega') | X(t) = (i, \omega)] = \\ = \begin{cases} p_{xy} & \text{if } i' = \lfloor \frac{i}{2} \rfloor + u[\omega - N] 2^{m-2}, \\ & x = \omega - (N-\nu) u[\omega - N], \\ & \omega' = y + (N-\nu) u[y - \nu], \\ & \quad y = 0, 1, \ldots, N-1 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where the use of $\omega' = y + (N-\nu) u[\omega - N]$ in the latter case means that a good channel $0 \leq y \leq \nu - 1$ leads to $\omega' = y$ whereas a bad channel $\nu \leq y \leq N-1$ leads to $\omega' = y + N - \nu$, $N \leq \omega' \leq 2N - \nu - 1$, i.e., a situation of bad channel and unresolved slot. According to the above rules, the transition probability matrix can be built, which will have only N non-zero entries per row.

In order to find the delay statistics, we proceed as follows. First of all, let us define an appropriate function τ

$$\begin{aligned} \tau : \mathcal{I}_N^{m-1} &\rightarrow \{0,1\}^{m-2} \\ \tau(\boldsymbol{\beta}) = \tau(\beta_0, \beta_1, \ldots, \beta_{m-2}) &= (b_0, b_1, \ldots, b_{m-2}) \\ \text{so that} \quad b_j &= u[\beta_j - \nu] \\ & \quad j = 0, 1 \ldots, m-2 \end{aligned} \quad (4)$$

where $\mathcal{I}_N = \{0, 1, 2, \ldots, N-1\}$. The meaning of $\tau(\cdot)$ is to transform vectors of base-N digits into binary digits so that the output digit is 0 if the input digit is less than ν, 1 otherwise (i.e., 0 corresponds to a slot were a successful transmission has occurred, whereas 1 corresponds to an erroneous slot.) Finally, note that the output vector is called $\boldsymbol{b}$ as $\tau(\cdot)$ is indeed used to obtain the bit-wise form $\boldsymbol{b}$ of the binary expression of $i(t)$.

Let $\boldsymbol{\Pi} = [\Pi_0 \ \Pi_1 \ \cdots \ \Pi_{\mathcal{K}}]^T$ be a column vector whose $\mathcal{K} = (2N - \nu) \cdot 2^{m-1}$ scalar entries represent the probabilities that the system starts in a given state. $\boldsymbol{\Pi}$ is computed as follows

- if $\omega \in \{0, 1, \ldots, \nu - 1\} \cup \{N, N+1, \ldots, 2N - \nu - 1\}$:

$$\boldsymbol{\Pi}_{(i,\omega)} = \sum_{z=0}^{\nu-1} \frac{\pi_z}{\mathcal{S}_\pi} \sum_{\boldsymbol{\beta} \in \mathcal{G}_b} p_{z\beta_0} \left[\prod_{j=0}^{m-2} p_{\beta_{j-1}\beta_j} \right] p_{\beta_{m-2}g} \quad (5)$$

- if $\omega \in \{\nu, \nu+1, \ldots, N-1\}$:

$$\boldsymbol{\Pi}_{(i,\omega)} = 0 \quad (6)$$

where $\mathcal{S}_\pi = \sum_{j=0}^{\nu-1} \pi_j$, $g = \omega - (N-\nu) u[\omega - N]$ and $\mathcal{G}_{\boldsymbol{b}} = \{\boldsymbol{\beta} \in \mathcal{I}_N^{m-1} : \tau(\boldsymbol{\beta}) = \boldsymbol{b}\}$, where $\boldsymbol{b}$ is the bit-wise representation of i and π_z, $z \in \{0, 1, \ldots, N-1\}$ is the Markov Channel steady state probability of state z. Let $\boldsymbol{e}_0 = [(i_0, \omega_0) \ (i_1, \omega_1) \ \cdots \ (i_{\mathcal{K}}, \omega_{\mathcal{K}})]^T$ be a column $\mathcal{K}$-sized vector of all zeros except for the entries corresponding to states $(i, w) \in \{(0,0), (0,1), \ldots, (0, N-1)\}$, that are equal to 1. If $\boldsymbol{\Phi}$ is the transition matrix of the Markov chain $X(t)$, we determine

$$\mathcal{P}_c[k] = \boldsymbol{\Pi} \boldsymbol{\Phi}^k \boldsymbol{e}_0, \quad k \geq 0\,. \quad (7)$$

The distribution $\mathcal{P}_c[k]$ is the probability that the delivery delay is less than or equal to k slots. Finally, the delivery delay statistics $P_d[k]$ is determined as

$$P_d[0] = \mathcal{P}_c[0]\,, \quad P_d[k] = \mathcal{P}_c[k] - \mathcal{P}_c[k-1] \quad \forall k > 0\,. \quad (8)$$

Note that $P_d[t]$ is determined by neglecting the propagation delay (that can be approximated as $t_{prop} = m/2$). In fact, what we are obtaining here is the statistics at the transmitter, whereas the actual delay process is evaluated at the receiver. However, since t_{prop} is constant these two distributions are simply the same distribution shifted shifted by this constant factor. For this reason, without loss of generality t_{prop} will not be considered in the sequel.

V. Derivation of the N-State Markov Model

In this section we report the procedure that we have used to derive an N-State Markov model representation of a Rayleigh fading channel. Let Γ denote the received signal to noise ratio (SNR), the pdf of Γ is exponential as follows [10]

$$p_\Gamma(\gamma) = \frac{1}{\gamma_0} e^{-\gamma/\gamma_0}, \ \ \gamma \geq 0 \quad (9)$$

where $\gamma_0 = E[\Gamma]$. Let $0 = \Gamma_0 < \Gamma_1 < \cdots < \Gamma_{K-1} < \Gamma_K = +\infty$ be $K+1$ thresholds for the SNR. The Rayleigh channel is said to be in state $k = 0, 1, \ldots, K-1$ if the received SNR is in the interval $[\Gamma_k, \Gamma_{k+1})$. Moreover, associated with each state there is an error probability ϵ_k that is the PDU error rate experienced in state k. We define $\mathcal{F}(\gamma)$ as the function mapping the instantaneous SNR level γ into the conditional PDU error probability. Once the threshold levels are chosen for every state, the PDU error rate in the generic state k is found as $\epsilon_k = (\int_{\Gamma_k}^{\Gamma_{k+1}} \mathcal{F}(\gamma) p_\Gamma(\gamma) d\gamma)/\theta_k$, where θ_k is the steady state probability to be in state k. In this work we assume a $\pi/4$-DQPSK modulation scheme [11], i.e., the bit error probability can be approximated as $\varepsilon(\gamma) \approx (4/3)\mathrm{erfc}(\sqrt{\gamma})$. $\mathcal{F}(\gamma)$ is then derived as $1 - (1 - \varepsilon(\gamma))^L$, where L is the ARQ packet length expressed in bits. The steady state probability θ_k is computed as

$$\theta_k = \int_{\Gamma_k}^{\Gamma_{k+1}} p_\Gamma(\gamma) d\gamma = e^{\Gamma_k/\gamma_0} - e^{\Gamma_{k+1}/\gamma_0} \quad (10)$$

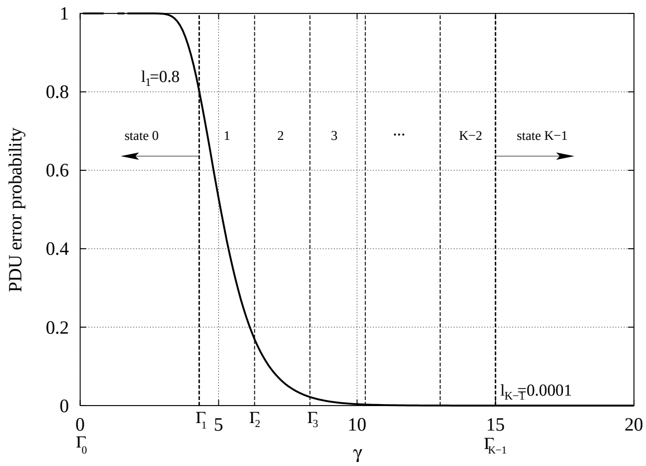

Fig. 1. SNR values partition method.

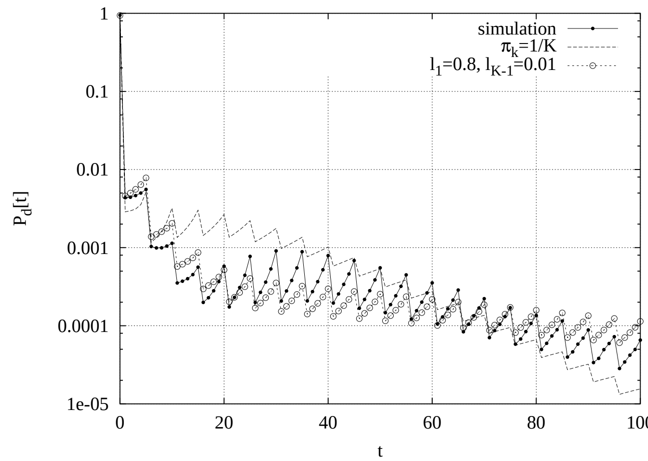

Fig. 2. Delivery delay statistics: comparison between Markov Channel analysis and Rayleigh channel simulation with $L = 360$ bits, bit rate 1024 Kbps, $f_d = 10$ Hz, $f_d T_p = 0.00343$, $m = 5$, $N = 10$.

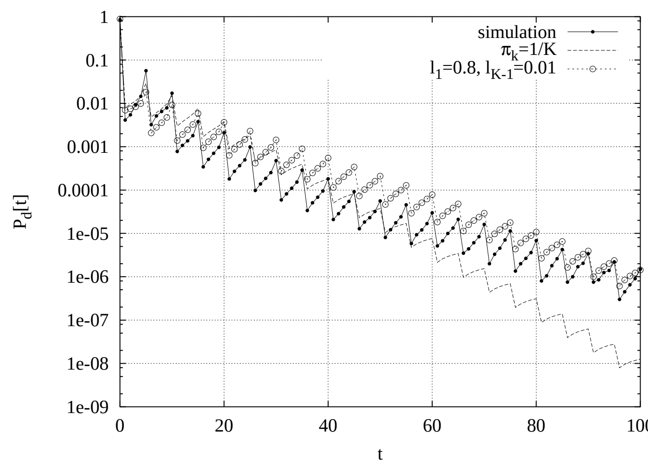

Fig. 3. Delivery delay statistics: comparison between Markov Channel analysis and Rayleigh channel simulation with $L = 360$ bits, bit rate 1024 Kbps, $f_d = 80$ Hz, $f_d T_p = 0.0274$, $m = 5$, $N = 10$.

The simplest approach for choosing the SNR thresholds [10] is to consider $\theta_k = 1/K$, $\forall\ k = 0, \ldots, K-1$. In this case, the threshold levels can be easily estimated by recursively applying Eq. (10), given that Γ_0 is known. However, this procedure leads to a rough estimation of the underlying fading process [11][12]. For this reason, we consider here an improved threshold selection criterion. We first choose two numbers, l_1 and l_{K-1} so that l_1 is close to zero and l_{K-1} is close to 1. Then we choose the first (Γ_1) and the last (Γ_{K-1}) unknown thresholds such that $\Gamma_1 = \mathcal{F}^{-1}(l_1)$ and $\Gamma_{K-1} = \mathcal{F}^{-1}(l_{K-1})$. Once Γ_1 and Γ_{K-1} are known, θ_0 and θ_{K-1} can be evaluated by Eq. (10). In this procedure we assign first the states 0 and $K-1$ to the SNR levels corresponding to a PDU error rate that is smaller than l_1 and larger that l_{K-1}, respectively. At this point, we use the remaining $K-2$ states to characterize the SNR interval between Γ_1 and Γ_{K-1}, i.e., where the PDU error rate is in the range $[l_1, l_{K-1}]$. The remaining $K-2$ thresholds are chosen to satisfy $\theta_k = (1-\theta_0-\theta_{K-1})/(K-2) = e^{\Gamma_k/\gamma_0} - e^{\Gamma_{k+1}/\gamma_0}$. In practice, the aim of our method is to assign the remaining states $1, 2, \ldots, K-2$ to the region containing the knee of the function $\mathcal{F}(\gamma)$. A graphical representation of this procedure is reported in Fig. 1. Once the thresholds have been computed, the transition probabilities are derived as in [12] according to:

$$\begin{aligned} t_{ij} &= \frac{\int_{\zeta_i}^{\zeta_{i+1}} \int_{\zeta_j}^{\zeta_{j+1}} f_{R_1R_2}(r_1, r_2, \rho) dr_1 dr_2}{\theta_i} \\ f_{R_1R_2}(r_1, r_2, \rho) &= \frac{4r_1r_2}{\lambda} e^{-(r_1^2+r_2^2)/\lambda} I_0(2\rho r_1 r_2/\lambda) \end{aligned} \tag{11}$$

where $\zeta_i = \sqrt{\Gamma_i/\gamma_0}$, $f_{R_1R_2}(r_1, r_2, \rho)$ is the bivariate Rayleigh joint pdf [12], $\rho = J_0(2\pi f_d T_p)$ is the correlation of two samples of the underlying Gaussian process that are spaced by T_p seconds, f_d is the Doppler frequency, T_p is the ARQ PDU transmission duration, $J_0(\cdot)$ and $I_0(\cdot)$ are the Bessel function and the modified Bessel function of the first kind and order zero. The Markov Chain $(\mathbf{T}, \mathcal{E})$ derived with this method is then transformed into the $N \times N$ transition matrix $\mathbf{P}$ following the procedure presented in Section III. The matrix $\mathbf{P}$ is finally used in the analysis presented in Section IV to derive the ARQ delay statistics.

VI. Results and Discussions

In this section we report some examples for the delivery delay statistics and we discuss the goodness of a Markov channel model in the approximation of the ARQ packet delay statistics in a Rayleigh fading channel. As a first result, in Figs. 2 and 3, we report the delivery delay statistics considering $f_d = 10$ Hz and $f_d = 80$ Hz, respectively. In both graphs, the statistics obtained by simulation is compared against the two threshold selection methods, i.e., the equal probability method ($\pi_k = 1/K$) and the refined procedure presented in the previous section. For both $f_d = 10$ Hz and $f_d = 80$ Hz the equal probability criterion leads to a rough estimation of the underlying fading process and to a poor approximation of the delay distributions as well. As a second observation, one can note that a Markov approximation of the actual channel error process is not able to perfectly match the real statistics. Also if the two distributions are in good agreement, the one derived using the Markov model can not reproduce the behavior characterizing the actual distribution. This is, indeed,

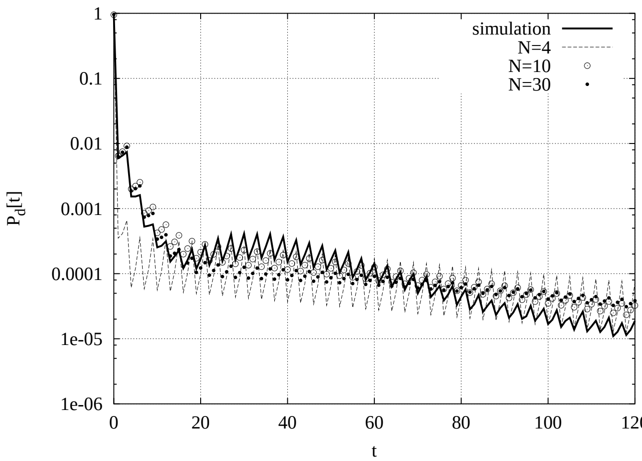

Fig. 4. Delivery delay statistics: comparison between Markov Channel analysis and Rayleigh channel simulation as a function of N with $L = 360$ bits, bit rate 1024 Kbps, $f_d = 10$ Hz, $m = 3$.

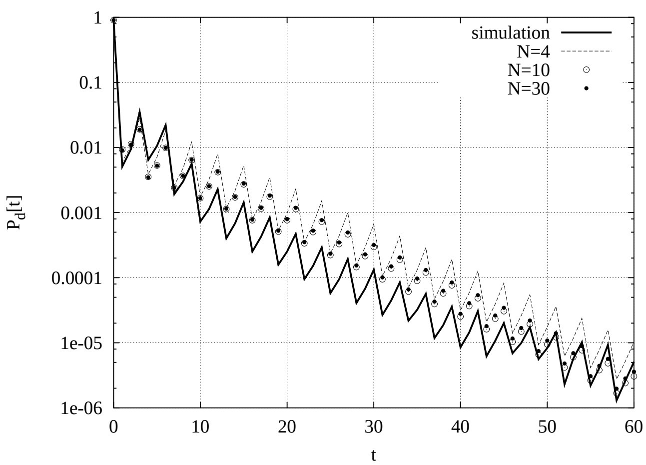

Fig. 5. Delivery delay statistics: comparison between Markov Channel analysis and Rayleigh channel simulation as a function of N with $L = 360$ bits, bit rate 1024 Kbps, $f_d = 100$ Hz, $f_d T_p = 0.0343$, $m = 3$.

a limitation of the Markov model that, even when a large number of states is considered, does not perfectly fit the actual fading process statistics. However, it is worth noting that the fading is a complex process that we are trying to approximate using a relatively simple model. In this sense, the obtained statistics is reasonably close to the real ones. In general, the match is good for the first three retransmission rounds, then the two distributions differ and the Markov model approach fails to reproduce the behavior of the actual distribution. It is also worth noting that the substantial differences observed in the autocorrelation function between the Markov and the fading model [12] are not observed for the delay performance of the ARQ protocol, where indeed the analytical performance is quite close to simulations.

In the following Figs. 4 and 5 we report some curves to discuss the dependence on the number of states of the Markov model (N). In general, the fit between simulation and analysis improves as N increases; this is particularly true when the Doppler frequency is low (i.e., $f_d = 10$ Hz). However, as the number of states increases ($N = 30$ in the presented graphs) the resulting statistics remains unchanged and no further improvements are observed. Moreover, taking as an example the $f_d = 10$ Hz case, an increase in N leads to a better approximation for the first part of the statistics, whereas the match is degraded elsewhere. In order to obtain better results it would be interesting to investigate how the statistics improves considering a different approach to derive the Markov chain. For instance, in [13] the authors considered the fading derivative as an additional dimension for this purpose. In that paper, they proved that this method can somehow reproduce the oscillatory behavior of the autocorrelation function. In [14] some results are reported concerning the mean throughput value of the SR ARQ protocol. Further investigations on how these techniques can improve the ARQ delivery statistics are left for future research.

VII. Conclusions

In this work two contributions are presented. First of all, an exact analysis to derive the delivery delay statistics of SR ARQ packets in an N-State Markov Model is presented. Secondly, this analysis is used to provide some results on the goodness of the Markov approximation of a Rayleigh fading channel in terms of delay statistics. The obtained results show that the statistics obtained using a Markov channel is reasonably close to the actual ones. However, the match between these distributions can not be made arbitrarily good by increasing the number of states due to intrinsic limitations of the Markov channel model.

References

[1] J. G. Kim and M. M. Krunz, "Delay Analysis of Selective Repeat ARQ for a markovian Source over a Wireless Channel," *IEEE Trans. Veh. Technol.*, vol. 49, no. 5, pp. 1968–1981, Sept. 2000.

[2] A. G. Konheim, "A Queueing Analysis of Two ARQ Protocols," *IEEE Trans. Commun.*, vol. 28, pp. 1004–1014, 1980.

[3] M. E. Anagnostou and E. N. Protonotarios, "Performance Analysis of the Selective-Repeat ARQ Protocol," *IEEE Trans. Commun.*, vol. 34, no. 2, pp. 127–135, Feb. 1986.

[4] Z. Rosberg and M. Shacham, "Resequencing Delay and Buffer Occupancy under the Selective Repeat ARQ," *IEEE Trans. on Inf. Theory*, vol. 35, no. 1, pp. 166–173, Jan. 1989.

[5] Z. Rosberg and M. Sidi, "Selective-Repeat ARQ: the Joint Distribution of the Transmitter and the Receiver Resequencing Buffer Occupancies," *IEEE Trans. Commun.*, vol. 38, no. 9, pp. 1430–1438, Sept. 1990.

[6] R. Fantacci, "Queueing Analysis of the Selective Repeat Automatic Repeat Request Protocol for Wireless Packet Networks," *IEEE Trans. Veh. Technol.*, vol. 45, no. 2, pp. 258–264, May 1996.

[7] M. Rossi, L. Badia, and M. Zorzi, "Exact Statistics of ARQ Packet Delivery Delay over Markov Channels with Finite Round-Trip Delay," *Accepted for Publication in Globecom 2003*.

[8] ——, "On the delay statistics of an aggregate of SR-ARQ packets over Markov channels with finite round-trip delay," in *IEEE WCNC 2003*, vol. 3, New Orleans, Mar. 2003, pp. 1773–1778.

[9] D. L. Lu and J. F. Chang, "Performance of Arq Protocols in Nonindependent Channel Errors," *IEEE Trans. Commun.*, vol. 41, no. 5, pp. 721–730, May 1993.

[10] H. S. Wang and N. Moayeri, "Finite-state Markov Channel–A Useful Model for Radio Communication Channels," *IEEE Trans. Veh. Technol.*, vol. 44, no. 1, pp. 163–171, Feb. 1995.

[11] Q. Zang and S. A. Kassam, "Finite-State Markov Model for Rayleigh Fading Channels," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1688–1692, Nov. 1999.

[12] C. C. Tan and N. C. Beaulieu, "On First-Order Markov Modeling for the Rayleigh Fading Channel," *IEEE Trans. Commun.*, vol. 48, no. 12, pp. 2032–2040, Dec. 2000.

[13] P. Bergamo, D. Maniezzo, A. Giovanardi, G. Mazzini, and M. Zorzi, "An improved Markov model for Rayleigh fading envelope," *IEE Electronics Letters*, vol. 38, no. 10, pp. 477–478, May 2002.

[14] ——, "An improved Markov Chain Description for Fading Processes," in *IEEE ICC 2002*, vol. 3, New York, Apr. 2002, pp. 1347–1351.