On the Effect of Feedback Errors in Markov Models for SR ARQ Packet Delays

Leonardo Badia
IMT Lucca Institute for Advanced Studies, Dept. of Computer Science and Applications (CSA),
piazza S. Ponziano 6, 55100 Lucca, Italy. e-mail: l.badia@imtlucca.it.

***Abstract*—Modern communication systems require effective error control techniques for wireless links. This problem is still relevant in many applications involving recently developed IEEE standards such as 802.16, or sensors for environmental monitoring in extreme conditions. A notable example is also underwater communication with acoustic transmission, where long delays may be present. Usually, such studies assume the availability of an error-free feedback channels which is used to perform retransmission requests in an ARQ fashion. This paper discusses the performance of the Selective Repeat ARQ scheme in terms of packet delay when the feedback that is sent back at the transmitter's side can be erroneous. A non instantaneous noisy feedback and a Bernoulli arrival process, for different traffic intensities, are considered. The system is modeled through Discrete Time Markov Chains, including the channel and the ARQ state. The system equations are derived and solved, and the impact of erroneous feedback is quantified. With respect to other similar contributions, the proposed approach presents the advantage of being directly implementable in any analysis using Markov chains to evaluate the system properties, instead of dedicated models. It can therefore be extended to a plethora of different scenarios, including different channel models and also including FEC features so as to obtain a Hybrid ARQ scheme.**

***Index Terms*—Queueing analysis, automatic repeat request, Markov processes, error analysis.**

I. Introduction

The focus of this paper is on the delay statistics for Selective Repeat Automatic Retransmission reQuest (SR ARQ), an error control technique which counteracts channel impairments via retransmissions of erroneous content, and does so selectively, i.e., retransmitting only those packets who are reported to be in error [1]. In the recent literature [2], [3], there has been a renewed interest for ARQ, also including hybrid ARQ techniques, where plain retransmission schemes are coupled with Forward Error Connection (FEC). Hybrid ARQ is used in high-quality multimedia applications [4] and can be included, as an optional extension, to improve the performance of the IEEE 802.16 standard [5]. Recent studies on this subject have analyzed ARQ schemes in detail [6], [7]. Another important field of application where investigations of ARQ techniques can be found is that of underwater acoustic communication [8], [9]. Underwater channels are highly variable and noisy, so the design of good error control mechanisms is key. Therefore, the evaluation of ARQ performance is relevant also to this case.

In this paper, non-instantaneous ARQ feedback is taken into account, i.e., the round-trip delay is significantly larger that the packet transmission time. This situation has been successfully investigated by means of Discrete Time Markov Chains (DTMC) [10]–[12]. As argued by these papers, several terms contribute to the overall delay. We adopt the following terminology, by considering the overall delay as subdivided into the *queueing delay* τ_Q and the *delivery delay* τ_D. The former is the time spent by a packet in the transmitter's queue, the latter is the delay between the first transmission of a packet and its release from the receiver buffer.[1] As SR ARQ packets may arrive out-of-sequence, their correct reception does not imply their delivery. A received packet is delivered after its resequencing, i.e., only when all packets with lower identifier have been correctly received as well.

Usually, some simplifications are considered, to make the system easier to study. One such hypothesis commonly made in the literature is that the feedback channel is error-free, i.e., the transmitter is required to retransmit only those packets which actually failed to be correctly received. It never happens that the packet is correctly arrived at destination, but the transmitter does not know it as it does not receive the acknowledgement. The goal of this paper is to take feedback errors into account in order to relax this limiting assumption and evaluate its consequences on the ARQ performance. This investigation is also important to correctly evaluate half-duplex communication setups where data packet and acknowledgements are sent on the same channel, as is the case in underwater transmission [13]. In these scenarios, feedback errors are caused not only by ambient noise but also by collisions.

In the literature, some similar approaches addressing feedback errors in ARQ systems have been proposed [14], [15]. However, these papers use dedicated models; here, instead, this issue is addressed right in the Markov model formulation, so that existing studies adopting this rationale can be directly extended. In this way, we are able to take error correlation into account in a quite natural manner, also for the reverse channel.

There are many possibilities to characterize the feedback channel as noisy. Errors can, for example, switch acknowledgement (ACK) and not acknowledgement (NACK) messages. However, it seems more realistic to assume that they are erasures, so that errors on the reverse channel cause the feedback to be lost. Then, the transmitter, assuming a stringent time out, retransmits the packet as if a negative acknowledgement were received. Our model also assumes that cumulative acknowledgment are used, so that, even in the case a feedback message is lost, the transmitter can be

[1] Without loss in generality, τ_D is evaluated at the receiver's side. This means omitting the constant propagation delay term, as done in [12]. In other words, if a packet is delivered at its first transmission, we say $\tau_D = 0$; if it is delivered in the next slot, $\tau_D = 1$ and so on.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

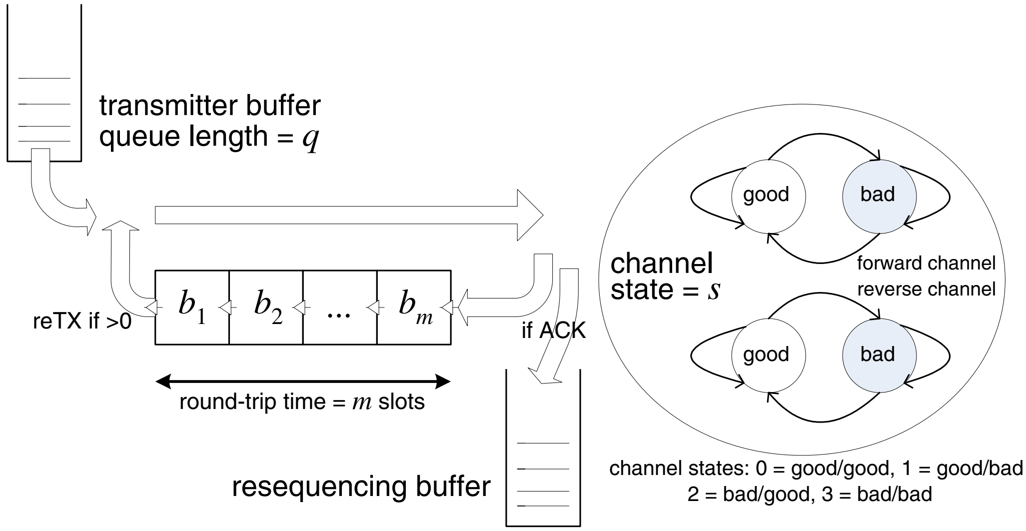

Fig. 1. A graphical representation of the system under analysis.

informed later by another feedback message that the packet has been indeed correctly received. This does not serve to avoid the first retransmission, but can prevent further unnecessary retransmissions from taking place. Thus, errors on the reverse channel have a somewhat lighter effect than those on the forward channel, as they can be discovered by the transmitter at the first instance a correct acknowledgement (or even a negative acknowledgement when the reverse channel is error-free) is received.

The contribution of this paper is therefore as follows. A Markov analysis of a SR ARQ system over a Markov channel is outlined. Feedback errors are explicitly taken into account in the model, which is used to derive the statistics of both queueing and delivery delays. Finally, numerical evaluations are shown, quantifying the effect of feedback errors.

The rest of this paper is organized as follows. Section II describes the model and the basic assumptions. Section III analytically derives the system equations. Section IV presents numerical results, and Section V concludes.

II. System Description

The SR ARQ scheme is considered over a slotted time axis where the transmission time of a packet equals one slot. The round trip time is m slots, with $m > 1$, i.e., a packet transmission outcome is known only m slots later. The arrivals at the transmitter's queueing buffer (assumed unlimited, as the re-sequencing buffer at the receiver's side) follow a Bernoulli process with intensity λ, i.e., either a packet enters the queue with probability λ or no packet is generated with probability $1-\lambda$; the case of multiple arrivals during the same time slot is neglected. Newly arrived packets are immediately available for transmission; however, retransmissions are prioritized. Hence, a packet can be transmitted in the same slot it arrives provided the queue is empty and no retransmission takes place.

The transmitter sends data packets to the receiver through a noisy channel. The receiver answers with ACK/NACK messages, according to the correct/erroneous reception of these packets, respectively, but also the feedback channel is noisy. To determine the packet outcome, we take a Markov model for the channel. In particular, the channel states are partitioned in four sets $\mathcal{S}_j$, labeled with j going from 0 to 3. Set $\mathcal{S}_0$ contains those states meaning correct reception and correct ACK, whereas in $\mathcal{S}_1$ the packet is correct but its ACK is erroneous so it is taken as a NACK. Set $\mathcal{S}_2$ corresponds to correctly receiving a NACK due to an error on the forward channel, whereas $\mathcal{S}_3$ means that both forward and reverse channel are erroneous. For simplicity, restrict to the case where the $\mathcal{S}_j$s are singletons, i.e., there is only one state in each set, which is denoted as j. Representing the state with binary digits, and assuming 0=correct, 1=erroneous, the most and the least significant bits denote the packet and the feedback outcome, respectively. The general case can be approached similarly, only with more complicated equations. When the channel is in state 0, a new packet is sent (in the case the queue is non empty, otherwise nothing is sent). In any other case, the transmitter assumes the packet was erroneous (this can be true, if the feedback is a NACK, or it may be that the feedback is a corrupted ACK) and retransmits it.

The whole system under analysis is represented in Fig. 1. In the literature, one frequent assumption is to have independent identically distributed (i.i.d.) channel errors on different time slots, which is a limiting approximation, as in reality wireless channels are known for their correlated behavior. To take the channel correlation into account, we make use of a Discrete Time Markov Chain. The transition from state i to state j, where $i, j \in \{0, 1, 2, 3\}$, over two subsequent time slots happens with probability p_{ij}. We collect these probabilities into the transition matrix $\mathbf{P} = \{p\}_{ij}$. By properly tuning these values, we can account for channel correlation in a tunable manner. This is one of the main advantages when modeling the wireless channel through Markov chains.

As in [10], the Markov analysis enlarges the state space of the channel chain by constructing a Markov chain for the whole system, including the ARQ process (i.e., the state of packets pending acknowledgement) and the queue at the transmitter's side at any time t.[2] This last part is simply described

[2]For brevity, we omit obvious time indices. However, the variables q, $\mathbf{b}$, and s introduced in the following should be written as $q(t)$, $\mathbf{b}(t)$, and $s(t)$.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

by an integer q denoting the queueing buffer occupancy. For what concerns the status of the pending packets, we use an m-sized retransmission window to track them; since retransmissions occur after m slots, any transmission outcome older than m slots do not impact on the delay statistics [12]. Thus, we consider an m-sized vector $\mathbf{b}$, with elements $b_i \in \{0,1,2,3\}$, for $1 \le i \le m$. The mth element describes the slot currently under transmission at time t, whereas b_j, $1 \le j \le m-1$ refers to the transmission at time $t-m+j$. If, at the corresponding time, a packet was transmitted, the value of b_j is equal to that the channel had then. However, it is even possible that no packet was transmitted (if the queue was empty and no packet arrived during that time slot). In this case $b_j = 0$; in fact, the two conditions of successful transmission or no transmission are equivalent for the delay statistics. Finally, as the channel is described by a Markov chain, we only need the last channel state s at time t, which may also be 0, 1, 2, or 3. According to the discussion above, we always have $b_m = s$ or $b_m = 0$ (the latter case being mandatory when no packet is transmitted; however, we also need s to track the Markov channel chain).

Following [12], one can prove that $(q, \mathbf{b}, s)$ evolves following a Markov chain, whose transition matrix is denoted with $\mathbf{T} = \mathbf{T}(\mathbf{P}, \lambda)$, as it depends on the arrival process intensity and on the channel matrix. (check again Fig. 1 where these variables are highlighted). In the following section, this Markov chain will be solved by deriving the steady-state equations. This is equivalent to find the steady-state distribution $\pi(q, \mathbf{b}, s)$ satisfying the condition $\pi(q, \mathbf{b}, s) = \pi(q, \mathbf{b}, s)\mathbf{T}(\mathbf{P}, \lambda)$.

Before proceeding with this derivation, an auxiliary function $\psi(\mathbf{x})$, which will be used in the next section, is needed. The argument of ψ is an $(m-1)$-sized vector. In practical cases, we will take $\mathbf{x}$ as the first $m-1$ elements of $\mathbf{b}$, i.e., $x_j = b_j$ for $j = 1, 2, \ldots, m-1$. The meaning of $\psi(\mathbf{x})$ is to characterize whether the transmitter is made aware that some packets previously transmitted were acknowledged by the receiver, but their acknowledgement were lost due to noisy feedback. Due to the stringent timeout requirement, if a packet is transmitted during channel state 1, i.e., correct forward channel but erroneous reverse channel, it is treated as if it were not acknowledged and therefore retransmitted. However, upon reception of any correct feedback (even a negative acknowledgement, but with correct reverse channel), the transmitter is informed of all previously acknowledged packets. From this instant on, all past transmissions performed when the channel was in state 1 can be treated as acknowledged. This implies, for example, that the transmitter will not further retransmit the corresponding packet. However, in case a long burst of errors is present on the reverse channel, the missing acknowledgement may be unnoticed at the transmitter's side.

We define $\psi(\mathbf{x})$ as the descriptor that all the preceding $m-1$ transmissions experienced feedback errors, so that the transmitter is unaware of any correctly received packet. Thus, $\psi(\mathbf{x})$ is equal to 1 if every x_j (i.e., b_j for $1 \le j \le m-1$) is odd, and 0 otherwise. In other words,

$$\psi(\mathbf{x}) = \prod_{j=1}^{m-1} \chi(x_j \in \{1,3\}) . \tag{1}$$

where the indicator $\chi(\cdot)$ is 0 or 1 according to the condition being false or true, respectively.

III. Solution of the System Equations

Having denoted with $\pi(q, b_1, b_2, \ldots, b_m, s)$ the stationary probability of state $(q, \mathbf{b}, s)$ one can write the set of balance equations (2)–(5) reported below. Note that $\pi(q, \mathbf{x}, y, z)$ is to be read as if $\mathbf{b} = (\mathbf{x}|y)$. The delta function $\delta[k]$ is equal to 1 if $k = 0$, and 0 otherwise.

The explanation of the equations is as follows. All the equations can be read similarly, i.e., as a sum on the transitions coming from all possible past channel states with a compatible evolution. Within brackets, two cases are considered, according to either zero or one packet arrived in the last slot (recall that multiple arrivals are forbidden), which happen with probabilities $(1-\lambda)$ and λ, respectively.

for $y = 0, 1, 2, 3$: (2)

$$\begin{aligned}\pi(q,\mathbf{x},y,y) = \sum_{z=0}^{3} p_{zy} \Bigg\{ &(1-\lambda)\bigg[\pi(q+1,0,\mathbf{x},z) + \Big(\delta[y]+\delta[y-1]\psi(\mathbf{x})\Big)\pi(q,1,\mathbf{x},z) + \delta[y]\delta[q]\,\pi(0,0,\mathbf{x},z)\Big) + \sum_{\ell=2}^{3}\pi(q,\ell,\mathbf{x},z)\bigg] \\ &+ \lambda\bigg[\pi(q,0,\mathbf{x},z) + \Big(1-\delta[q]\Big)\bigg(\Big(\delta[y]+\delta[y-1]\psi(\mathbf{x})\Big)\pi(q-1,1,\mathbf{x},z) + \sum_{\ell=2}^{3}\pi(q-1,\ell,\mathbf{x},z)\bigg)\bigg]\Bigg\}\end{aligned}$$

for $y = 1, 2, 3$: (3)

$$\begin{aligned}\pi(q,\mathbf{x},0,y) = \sum_{z=0}^{3} p_{zy} \Bigg\{ &(1-\lambda)\bigg[\Big(1-\psi(\mathbf{x})+\psi(\mathbf{x})\,\delta[y-2]\Big)\pi(q,1,\mathbf{x},z) + \delta[q]\pi(0,0,\mathbf{x},z)\bigg] \\ &+ \lambda\Big(1-\delta[q]\Big)\Big(1-\psi(\mathbf{x})+\psi(\mathbf{x})\,\delta[y-2]\Big)\pi(q-1,1,\mathbf{x},z)\Bigg\}\end{aligned}$$

$$\pi(q,\mathbf{x},1,3) = \sum_{z=0}^{3} p_{zy}\,\psi(\mathbf{x})\bigg\{(1-\lambda)\pi(q,1,\mathbf{x},y) + \lambda\Big(1-\delta[q]\Big)\,\pi(q-1,1,\mathbf{x},y)\bigg\} \tag{4}$$

for any other case, i.e., $(y,z) = (1,2)$, or $y \in \{2,3\}$: $\qquad \pi(q,\mathbf{x},y,z) = 0$ (5)

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

Now, the equations are discussed in detail. In the most common cycling of the system, the channel state and the status of the last transmitted packet coincide. This circumstance is described by (2). Read this equation term by term as it appears above. The system status evolve to $(q, \mathbf{x}, y, y)$ from $(Q, \alpha, \mathbf{x}, z)$ due to the cycling property of the packets; Q is the length of the queue in the previous time slot and z is a generic channel state. Thus, the evolution happens from a queue of length $Q = q + 1$ if no packet arrives and no retransmission occurs, i.e., $\alpha = 0$.

Alternatively, a retransmission can happen instead, meaning that $Q = q$, in three possible cases: (i) retransmission of a packet which was correctly received, but with lost acknowledgement, i.e., $\alpha = 1$ (this case is further discussed below); (ii) nothing to transmit, meaning that the queue is empty and there are no retransmissions (this case only exists when $q = 0$ and only leads to channel $y = 0$, which explains the multiplicative term $\delta[q]\delta[y]$); (iii) retransmission of an erroneous packet (whose NACK can be received or not, i.e., α can be either 2 or 3). Case (i) means that, as a result of the retransmission of a packet in state 1, the channel state and the status of the last transmitted packet are identical; this can only happen either if they are both 0 or both 1, so the term $\delta[y]$ or $\delta[y-1]$ is included, respectively. However, this last case also requires that the missing acknowledgment is unnoticed, i.e., the additional coefficient $\psi(\mathbf{x})$ is needed.

Analogously, consider instead the case where a packet arrives, described in the line below. The terms are the same, only the old queue is one packet shorter, as we also have the newly arrived one, and term (ii) from the previous computation is excluded, as in that case we have the new packet to transmit.

The exceptions to the case where the channel state does not coincide with the last packet status are as follows. Consider first, in (3), the case where the channel state is $1, 2, 3$ and the last packet status is 0. This can happen because of two reasons: either a packet with status 1 was retransmitted and the transmitter later recognizes it was already received, or there is nothing to transmit. This latter is identical, only with a different channel state, to case (ii) above. The former instead involves that either the channel state is 2, in which case the acknowledgement is correct (the forward transmission is not, but we do not need it as the packet has already been decoded before), or the missing acknowledgement has been reported, i.e., $\psi(\mathbf{x})$ is 0. The term $1 - \psi(\mathbf{x}) + \psi(\mathbf{x})\delta[y-2]$ is indeed equal to 0 only if y is *not* equal to 2 and the missing acknowledgement is undetected, i.e., $\psi(\mathbf{x}) = 1$.

There is only one case left, which is that where the channel state is 3 and the last packet status is 1, which happens when a correct packet with undetected missing acknowledgement (this justifies the coefficient $\psi(\mathbf{x})$) is transmitted over an erroneous channel and remains as such. Again, two terms are present, depending on either no packet arrival, thus the queue length is unchanged, or packet arrival, with shorter queue length in the previous time slot; clearly, this latter case is possible only if the current queue length is not zero.

This system of equations can be solved, together with the additional condition $\sum \pi = 1$, to derive the steady-state probabilities π. After the steady-state probabilities have been found, the queueing and delivery delay statistics are derived as explained in [10], [12]. The basic idea is to consider a *tagged* packet entering the queue and derive its statistics. Since the generation of the tagged packet implies to condition on having an arrival (the tagged packet's), we have a conditional probability equal to $\pi(q, \mathbf{b}, s)\mathbf{T}(\mathbf{P}, 1)$, where λ is put equal to 1 for one slot. After the tagged packet entered the system, its delay terms are not affected by subsequent packet arrivals. Thus, we can turn off the arrival process and consider another transition matrix equal to $\mathbf{T}(\mathbf{P}, 0)$. Now, define $\mathcal{Q} = \{(0, \mathbf{b}, s) : \mathbf{b} \in \{0, 1, 2, 3\}^m, s \in \{0, 1, 2, 3\}\}$ and $\mathcal{G} = \{(0, \mathbf{b}, s) : \mathbf{b} \in \{0, 1\}^m, s \in \{0, 1, 2, 3\}\}$. Both $\mathcal{Q}$ and $\mathcal{G}$ are absorbing sets for the Markov chain; set $\mathcal{Q}$ comprises all the states corresponding to the release of the tagged packet from the queueing buffer, i.e., empty queue, where set $\mathcal{G}$ is entered when the tagged packet and all packets transmitted prior to it are acknowledged. Note that in $\mathcal{G}$ any b_j can take value equal to either 0 or 1; in the latter case, it may happen that the transmitter still keeps retransmitting some packets, nevertheless the tagged packet has already been delivered. The cumulative distribution of the queueing delay is

$$\mathcal{C}_{\mathcal{Q}}[t] = \pi(q, \mathbf{b}, s) \cdot \mathbf{T}(\mathbf{P}, 1) \cdot [\mathbf{T}(\mathbf{P}, 0)]^t \cdot \mathbf{e}_{\mathcal{Q}}, \quad k \geq 0\,, \quad (6)$$

where $\mathbf{e}_Q$ is a column vector of indicator functions of the set $\mathcal{Q}$, i.e., its values are 1 in correspondence with states belonging to $\mathcal{Q}$ and 0 elsewhere. The distribution $\mathcal{C}_{\mathcal{Q}}[t]$ is the probability that the queueing delay is lower than or equal to k slots. Thus, the probability $\text{Prob}\{\tau_Q = t\}$ is determined as:

$$\text{Prob}\{\tau_Q = t\} = \begin{cases} \mathcal{C}_{\mathcal{Q}}[0] & \text{if } t = 0 \\ \mathcal{C}_{\mathcal{Q}}[t] - \mathcal{C}_{\mathcal{Q}}[t-1] & \text{if } t > 0 \end{cases} \quad (7)$$

The statistics of the overall delay τ_G can be evaluated by following the same approach, i.e., by taking a column vector $\mathbf{e}_G$ containing the indicator functions of the set $\mathcal{G}$ instead. Finally, the delivery delay τ_D, corresponding to the delay between the first transmission of the tagged packet over the channel and the resolution of every pending packet, is $\tau_G - \tau_Q$. We can similarly define its cumulative distribution $\mathcal{C}_{\mathcal{D}}$ as well.

IV. Numerical Results

In the following, we present a numerical evaluation of SR ARQ delays with erroneous feedback. The Markov channel has been obtained as the Kronecker product of two independent Markov chains. The channel transition matrices relative to the forward and the reverse channel are denoted with

$$\mathbf{F} = \begin{pmatrix} f_{00} & f_{01} \\ f_{10} & f_{11} \end{pmatrix} \quad \text{and} \quad \mathbf{R} = \begin{pmatrix} r_{00} & r_{01} \\ r_{10} & r_{11} \end{pmatrix} \quad (8)$$

respectively. For the sake of simplicity, they are both taken with 2 states, labeled 0 and 1, where state 0 is error-free and state 1 is erroneous with probability 1. This is the same model reported in [12] and can be promptly extended to more elaborate Markov chains without changing the numerical insight.

All the results shown in the following have been confirmed by means of simulations, showing an excellent agreement, as the statistics of the delay terms is derived in an exact manner without any approximation. For this reason, simulation results

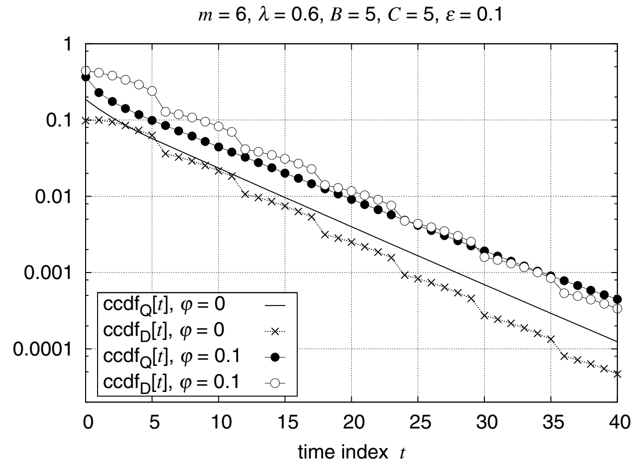

Fig. 2. Complementary cumulative distribution of the delays for $m = 6$, $\varepsilon = 0.1$, $B = 5$, for error-free/noisy feedback with $\varphi = 0.1$, $C = 5$.

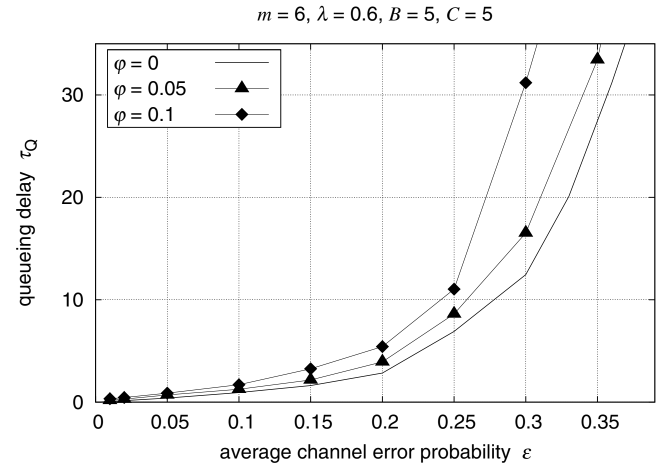

Fig. 3. Queueing delay τ_Q as a function of ε, for $m = 6$, $B = C = 5$, and various values of the average error probability on the reverse channel.

are not shown as they would be redundant (i.e., they are almost everywhere overlapping with the analytical curves).

In order to have a clean representation on the graphs, the following parameters are defined, which fully characterize the matrices $\mathbf{F}$ and $\mathbf{R}$. For the forward channel, we define the average error probability as $\varepsilon = f_{01}/(f_{10} + f_{01})$ and the average error burst length as $B = 1/f_{10}$. Similarly, for the reverse channel the average error probability is $\varphi = r_{01}/(r_{10} + r_{01})$ and the average error burst length is $C = 1/r_{10}$. Finally, the resulting four-state channel matrix $\mathbf{P}$ is defined as

$$\mathbf{P} = \begin{pmatrix} p_{00} & \cdots & p_{03} \\ \vdots & \ddots & \vdots \\ p_{30} & \cdots & p_{33} \end{pmatrix} = \mathbf{F} \bigotimes \mathbf{R} \tag{9}$$

In the following, we consider a round-trip time equal to $m = 6$ slots. The arrival process is taken with intensity $\lambda = 0.6$. The default values for B and C are taken equal to 5 slots, indicating mild correlation in both the forward and the reverse channel. The values of ε and φ are taken as the independent variable in some plots. The reference cases are $\varepsilon = 0.1$ and $\varphi \in \{0, 0.05, 0.1\}$, i.e., a moderately erroneous forward channel and a reverse channel which can be without errors, with the same error rate of the forward channel, or in an intermediate situation. The model supports also higher error rates on the reverse channel, and will be extended in this sense in some results, but in general having $\varphi > \varepsilon$ may be deemed as not very realistic as feedback packets are usually small thus their error rate is expected to be not higher than that on the forward channel.

A first result which can be obtained through the proposed model, shown in Fig. 2, concerns the complementary cumulative distributions of the queueing and delivery delay for $\varepsilon = 0.1$, $\varphi = 0$ or $\varepsilon = \varphi = 0.1$. In the graphs, "ccdf$_Q$" and "ccdf$_D$" are the complementary functions to $\mathcal{C_Q}$ and $\mathcal{C_D}$ derived before. As in the latter case the feedback is noisy, the curves are shifted to higher values, implying a heavier tail of the delay distribution. However, it is important to observe that, from a qualitative standpoint, the behavior of the statistics is similar. The cumulative distribution is approximately linear for the queueing delay and has a cyclic behavior, with cycle size equal to m, for the delivery delay, in both cases. This confirms that the main impact of noisy feedback in ARQ system is to re-scale the distributions of the delays, without significantly changing the qualitative behavior.

However, their re-scaling happens in a different manner for the queueing and the delivery delay; it should be noted that different choices of values obtain different variations of the delay, so the performance trends shown here should not be generalized. For this reason, it is interesting to quantitatively evaluate in more detail the width of the increases, due to feedback errors, of the average queueing delay and the average delivery delay.

This is investigated in Figs. 3 and 4, where the average delays are plotted, respectively, as functions of ε for various values of φ. It is important to notice that the queueing delay is only slightly increased when the value of φ is limited. On the other hand, the delivery delay grows more than proportionally. In particular, τ_D is higher for $\varphi > \varepsilon$ than the corresponding situation with φ and ε reverted. For instance, compare in Fig. 4 the case with $\varphi = 0.1$, $\varepsilon \to 0$, with the analogous values when $\varphi = 0$, $\varepsilon = 0.1$. This suggests that errors on the reverse channel have a lower impact than those on the forward channel for the queueing delay, whereas they are more significant for the delivery delay.

Although the delivery delay is more sensitive to feedback errors, at the same time it should be observed that the value of τ_D is lower compared to that of τ_Q in the considered range of values. Thus, in general, it is confirmed that the overall impact of feedback errors is comparable with an increased error probability on the forward channel, though slightly lower than this. Moreover, the impact of feedback errors is limited when they are rare events; the delay increases become more acute when ε and φ are comparable.

Finally, Fig. 5 generalizes these results to a variable φ. In this case the value of ε has been fixed to 0.1. It is visible how even in this case the average delivery delay exhibits an almost linear increase, whereas the queueing delay explodes for high

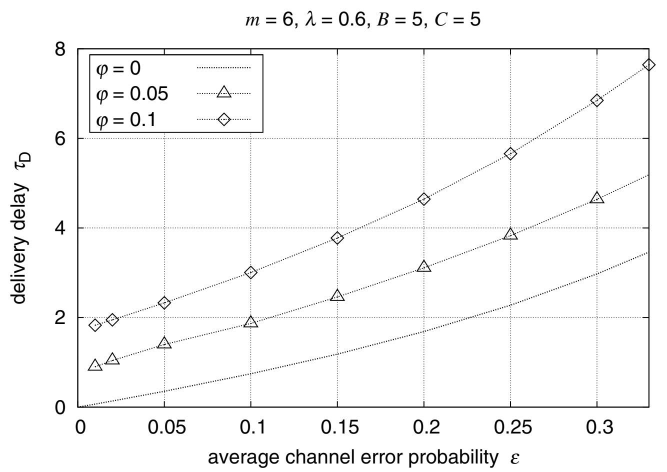

Fig. 4. Delivery delay τ_D as a function of ε, for $m = 6$, $B = C = 5$, and various values of the average error probability on the reverse channel.

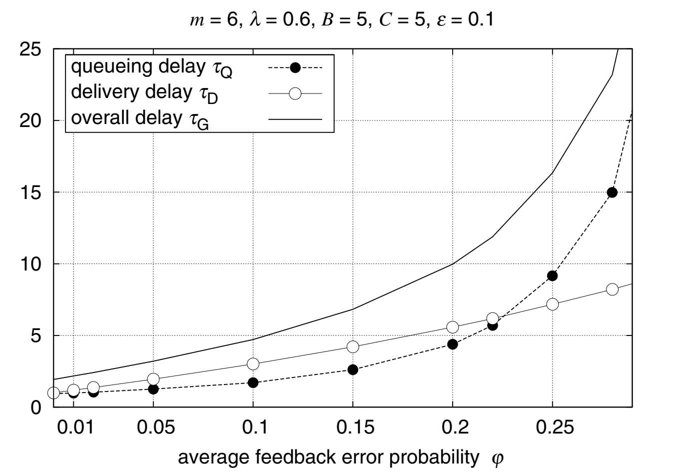

Fig. 5. Average delay terms as a function of φ, for $m = 6$, $\varepsilon = 0.1$, $B = C = 5$.

values of φ. This behavior is again qualitatively similar to that of these delay terms when the error probability on the forward channel is increased instead [12]. According to the error probability, the dominant term of the overall delay is shown to be the delivery delay for low error rates, and the queueing delay for high error rates. The delivery delay is also less sensitive to the variation of the error rate on the reverse channel, as was shown to hold by a similar analysis for the errors on the forward channel.

V. Conclusions

Through a Markov analysis of the SR ARQ technique, we studied the impact of errors in the reverse channel carrying acknowledgements. The proposed model is based on a direct extension of existing Markov approaches and enables a quantitative evaluation of the effect of errors in the reverse channel.

Several numerical results were presented to quantify the whole statistics of the process. The investigation is entirely analytical and can be regarded as an add-on for existing studies based on Markov models. At the same time, it can also be seen as a practical way to evaluate the performance of SR ARQ schemes, or those based on similar methodologies, in the presence of feedback errors.

These evaluations are especially useful for scenarios where the transmitter and receiver do not have a separate feedback channel and/or multiple users share the same one. Compared to traditional single link communications, these cases have a significantly higher rate of feedback errors, which should be carefully taken into account.

To this end, possible extensions of the present paper, which are currently ongoing, include the analysis of specific feedback error models which more directly describe these situations and the evaluation of their impacts, as well as the identification of possible enhanced ARQ mechanisms that take these phenomena into account.

References

[1] S. Lin, D. J. Costello, and M. J. Miller, "Automatic-Repeat-reQuest error control schemes," *IEEE Commun. Mag.*, vol. 22, no. 12, pp. 5–17, 1984.

[2] L. Badia, M. Levorato, and M. Zorzi, "Markov analysis of selective repeat type II hybrid ARQ using block codes," *IEEE Trans. Commun.*, vol. 56, no. 9, pp. 1434–1441, Sep. 2008.

[3] W. Luo, K. Balachandran, S. Nanda, and K. Chang, "Delay analysis of selective-repeat ARQ with applications to link adaptation in wireless packet data systems," *IEEE Trans. Wireless Commun.*, vol. 4, no. 3, pp. 1017–1029, May 2005.

[4] F. Zhai, Y. Eisemberg, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Rate-distortion optimized hybrid error control for real-time packetized video transmission," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 40–51, Jan. 2006.

[5] A. Ghosh, D. R. Wolter, J. Andrews, and R. Chen, "Broadband wireless access with WiMax/802.16: current performance benchmarks and future potential," *IEEE Commun. Mag.*, vol. 43, no. 2, pp. 129–136, Feb. 2005.

[6] M. Moh, T.-S. Moh, and Y. Shih, "On enhancing WiMAX hybrid ARQ: A multiple-copy approach," in *Proc. IEEE Consumer Communications and Networking Conference (CCNC)*, 2008, pp. 902–906.

[7] A. Sayenko, V. Tykhomyrov, H. Martikainen, and O. Alanen, "Performance analysis of the IEEE 802.16 ARQ mechanism," in *Proc. ACM MSWiM*, 2007, pp. 314–322.

[8] J.-W. Lee, J.-P. Kim, J.-H. Lee, Y. Jang, K. C. Dho, K. Son, and H.-S. Cho, "An improved ARQ scheme in underwater acoustic sensor networks," in *Proc. MTS/IEEE Oceans*, Kobe, Japan, Apr. 2008, pp. 1–5.

[9] P. Xie and J.-H. Cui, "An FEC-based reliable data transport protocol for underwater sensor networks," in *Proc. Int. Conf. on Computer Communications and Networks (ICCCN)*, 2007, pp. 747–753.

[10] L. Badia, "On the impact of correlated arrivals and errors on ARQ delay terms," *IEEE Trans. Commun.*, vol. 57, no. 2, Feb. 2009.

[11] M. Rossi, L. Badia, and M. Zorzi, "Exact statistics of ARQ packet delivery delay over Markov channels with finite round-trip delay," *IEEE Trans. Wireless Commun.*, vol. 4, no. 4, pp. 1858–1868, Jul. 2005.

[12] L. Badia, M. Rossi, and M. Zorzi, "SR ARQ packet delay statistics on Markov channels in the presence of variable arrival rate," *IEEE Trans. Wireless Commun.*, vol. 5, no. 7, pp. 1639–1644, Jul. 2006.

[13] P. Casari, M. Rossi, and M. Zorzi, "Towards optimal broadcasting policies for HARQ based on fountain codes in underwater networks," in *Proc. WONS*, 2008, pp. 11–19.

[14] R. Cam and C. Leung, "Throughput analysis of some ARQ protocols in the presence of feedback errors," *IEEE Trans. Commun.*, vol. 45, no. 1, pp. 35–44, Jan. 1997.

[15] E. Malkamäki and H. Leib, "Performance of truncated Type-II hybrid ARQ schemes with noisy feedback over block fading channels," *IEEE Trans. Wireless Commun.*, vol. 48, no. 9, pp. 1477–1487, Jul. 2000.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.