

Motivation for Error-Tolerant Communication

Till Halbach

Department of Telecommunications
Norwegian University of Science and Technology (NTNU)
7491 Trondheim, Norway

ABSTRACT

The transmission of large data streams over error-prone channels as e.g. in multimedia applications is inherently linked to long transmission delays if automatic repeat request schemes are used. As this article* will show, the delay can be reasonably traded against residual bit errors if a short transmission time has highest priority. The dependency of the delay on two important factors, packet length and channel bit error rate, is determined to be non-linear and strictly monotonously growing. Furthermore, transmission behavior and properties of a plain binary symmetric channel and one with additional repeat request technique are simulated and compared to previous research. The simulations finally lead to a redefinition of the formula for the estimation of the residual bit error rate of a non-transparent channel.

Keywords: ARQ protocol, transparent/non-transparent channel, transmission delay, packet errors, residual bit errors.

1. INTRODUCTION

Still image and video sequence services over mobile and wireless networks for e.g. mobile phones or so-called PDAs will soon be introduced, and new systems, channels and protocols are proposed and introduced almost every day. For transmission of visual information at a perceptually acceptable quality, the amount of data is generally much higher than the data transmitted for normal speech or text. This leads to large transmission delays and an increased exposure of sensitive compressed data to channel errors. Transmission delay and the quality of the received images are therefore the most relevant parameters when transmitting over wireless, i.e. error-prone, channels.

But how is the relationship between these parameters, and which are the factors affecting their values? This is discussed in the following under various aspects.

2. CHANNEL TYPES

Two kinds of channels are considered for data transmission in general. Both are based on the binary symmetric channel (BSC) model. Furthermore, packet transmission is taken into consideration.

The first channel type is called transparent because, for a codec system developer, there are no effects of the channel on the transmitted code stream. It is a virtually error-free channel and can be realized by e.g. the popular automatic repeat request (ARQ) mechanism. ARQ is based on the concept of packet retransmission in case a transmission error is detected in that respective packet. The second channel type is the non-transparent channel which can affect the code stream. Hence, the decoder must be able to cope with a possibly large number of residual bit errors.

However, there is another important difference between these two channels. A plain BSC, i.e. a non-transparent channel — sometimes also denoted as speech channel in GSM context — means transmission once irrespective of the quality of the received signal. On the other hand, a BSC with build-in ARQ — the transparent channel, in GSM terminology sometimes denoted as data channel — would repeat sending the

Further author information: E-mail: halbach@tele.ntnu.no; Phone: +47 73 59 44 88; Fax: +47 73 59 26 40

*Parts of this article have been presented on the Norwegian national conference NORSIG-2001.

same packets several times in the presence of errors, hereby leading to larger transmission delays than for a non-transparent channel. An error-robust decoder may want to prefer shorter transmission delays and an error-affected code stream to large transmission times and a zero-error code stream. Undoubtedly, there is a trade-off here, depending on the data type transmitted.

The criteria for the decoder to determine this trade-off and their relationship to each other are discussed in the subsequent sections.

3. SYSTEM SPECIFICATION

Given the case that one wants to transmit a not further specified bit stream bits over both non-transparent and ARQ channel. Since ARQ actually denotes a family of transmission protocols, this comparison between a protocol and a channel is legal only with respect to the respective transmission delays D_i , residual bit error rate ($RBER$) and packet error rate (PER).

Transmission delay denotes the total time required for data transmission over a channel. The residual bit error rate is the relative number of corrupted bits at the channel output with respect to the complete bit stream. The packet error probability, also referred to as retransmission probability, is formally defined as

$$PER = \frac{N_{re}}{N_{tr,total}}, \quad (1)$$

where $N_{tr,total}$ is the total number of transmissions, including retransmissions, and N_{re} is the number of re-transmissions.

Both channel models split the bit stream into packets, sometimes known as bursts or frames. For a plain BSC, the transmission delay will be constant and depend linearly on the number of packets and the average time used for one-way transmission T_p of one packet. Hereby, it is assumed that the packets are completely filled by the bit stream and that parameters of lower layers like the time used for computation of packet overhead, modulation, etc. are included in T_p . $RBER$ is independent of L_p , the length of one packet in bits. The storage capabilities required on the sender side correspond to buffering one burst length of data. The non-transparent channel does not need a reverse or feedback channel.

When the term ARQ is mentioned in this context, it refers to the use in its simplest possible functionality, i.e. the so-called send-and-wait mode. This means that the transmitter sends out a frame and then waits for a positive or negative acknowledgment packet from the receiver that this frame has been received error-free.¹ Time monitoring guarantees that a certain time-out threshold is not exceeded. This is the slowest possible mode. Other modes like go-back- N or selective-repeat allow for shorter transmission delays but also demand more storage capabilities. For simplicity reasons, it is assumed further that every error-affected packet is detected as such and requested for retransmission. This means that the ARQ scheme can deliver data virtually error-free to the receiver. This assumption is valid for almost all ARQ channels since $RBER_{ARQ}$ is typically around 10^{-14} or less.² Further requirements of send-and-wait ARQ are a reverse or acknowledgment channel and the same storage capabilities as for BSC. The average time between sending of a packet and receiving the acknowledgment, the so-called round-trip propagation delay, is $2T_p$, suggesting that a packet and its acknowledgment are transmitted equally fast. It is further assumed that the number of retransmission is not limited and that successive retransmissions of the same frame are decoded without reference to previous transmissions.

4. PREVIOUS WORK

In 1999, Perkis and Cardelo theoretically derived an expression for the transmission delay D_{ARQ} of an ARQ channel as a function of the transmission time T_p and the retransmission probability PER ³:

$$D_{ARQ} = T_p \frac{1}{1 - PER} \quad \text{for } PER < 1. \quad (2)$$

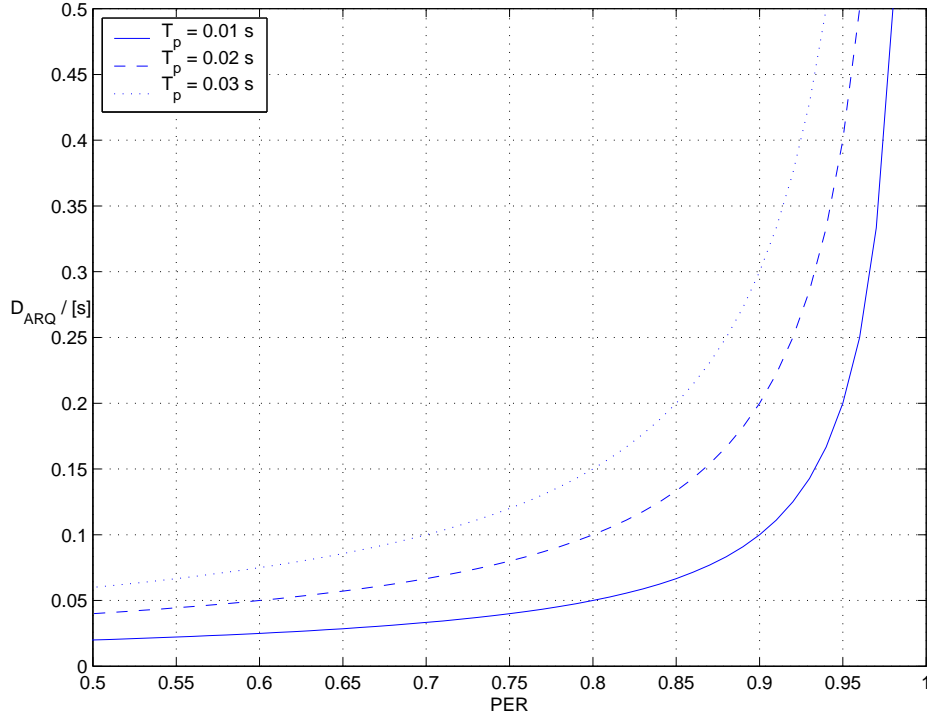


Figure 1: Behavior of D_{ARQ} dependent on PER with T_p as a parameter.

This is visualized by the plot in Fig. 1. The graph says that the delay of one burst grows strictly monotonously with increasing PER . The transmission delay for the whole bit stream will therefore show the same characteristic as plotted but, of course, be always larger.

The author argues further that noisy BSCs thus will result in significant transmission delays if an ARQ scheme is used to guarantee error-free code streams. To support this statement, another formula is developed, Ineq. 3, under the presumptions that $PER < 1$ and $L_p > 0$.

$$RBER \geq \begin{cases} \frac{1}{L_p} \frac{PER}{1-PER} & \text{if } PER \leq \frac{L_p}{1+L_p} \\ 1 & \text{else.} \end{cases} \quad (3)$$

There, the packet error rate of an ARQ channel is translated to the residual bit error rate of a non-transparent channel. Fig. 2 visualizes Ineq. 3 and shows that the residual bit error rate increases non-linearly and at least strictly monotonously with growing retransmission probability. These derivations are valid only under some conditions: First of all, Ineq. 3 is a lower bound, typical $RBER$ s will be higher. Moreover, it is a theoretical result as $RBER > 0.5$ for a BSC makes no sense. Packet overhead information is not accounted for.

To achieve error-robust communication, the argument is that Ineq. 3 can be used by an appropriate error resilience technique to trade transmission delay against residual bit errors. If a short delay is desired and errors are e.g. handled by an error-resilient decoder, then a non-transparent transmission mode could be chosen. On the other hand, if the data is required totally error-free, an ARQ mode can be used.

Since the results presented by Perkis and Cardelo are of theoretical nature only, it is therefore investigated in the following sections whether the results of transmission simulations support the presented theory.

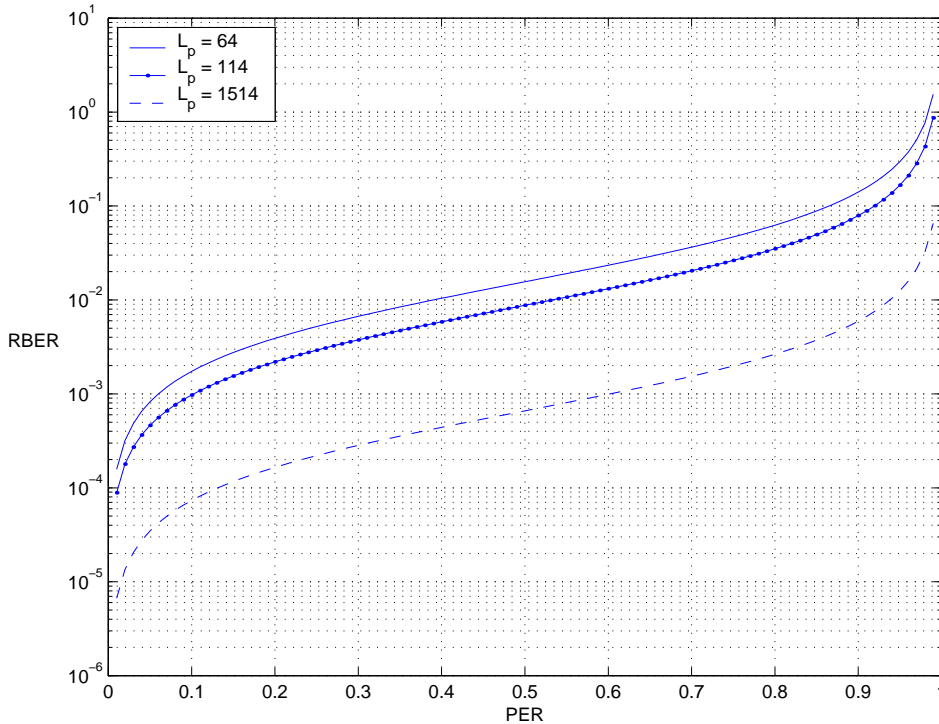


Figure 2: $RBER$ as a function of PER with the packet size L_p as a parameter.

5. COMPARISON PER — $RBER$, REVISITED

That the comparison between non-transparent channel and ARQ channel is not straight-forward shall be visualized by the help of an example, Fig. 3. The figure shows three bit streams, whereof the two lower streams are partitioned into so-called bursts similar to the method in the well-known GSM standard. The upper stream shows the localization of single random bit errors, denoted by the symbol \times . Note that the bit error pattern is independent of the type of channel used. The stream in the middle visualizes the forward transmission over a non-transparent channel, and the lower stream the use of ARQ as specified in Sec. 3. Perkins and Cardelo argue now that, if two retransmissions are necessary, at least two bits were inverted, one during first transmission and one during the second one. This might be correct for an ARQ channel, but it is not a proper suggestion for a non-transparent channel. Intuitively, the total number of bit errors might be just one for a non-transparent channel with the same error pattern that leads to two ARQ retransmissions, for the bit errors occurring after having finished transmission of the second packet over the non-transparent channel do not affect $RBER$! Hence, one can see that it is impossible to derive any deterministic dependency between these two kinds of channels, i.e. between $RBER$ and PER , only an empirical approach can be made.

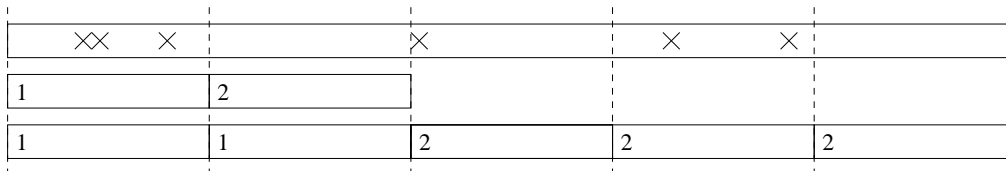


Figure 3: Transmission for two channel types.

Therefore, some simple simulations were done. Two channels are considered, a plain BSC and one with an additional ARQ scheme. A bit stream consisting of 6.336 Kbyte is to be transmitted over both channels in form of bursts. The amount of data corresponds to a monochrome QCIF video frame coded at an effective rate of 0.25 bpp. The average transmission time of one burst consisting of 1000 bits is set to be $T_p = 10$ ms, hereby also accounting for reverse channel and time-out threshold for ARQ. This value has usually to be replaced by two different delays for forward and reverse channel due to bandwidth asymmetry. On the forward channel, the bursts are exposed to random bit errors with varying bit error probability, $BER = 10^{-9}, 5 \cdot 10^{-9}, \dots, 10^{-2}, 5 \cdot 10^{-2}$. All experiment results are arithmetically averaged over 100 error patterns with different seeds. Additionally, the packet lengths vary in steps of 50 bits from 64 to 1514, also covering the raw data burst length of 114 bits as used in GSM. For both channel types, the transmission delays D_i and respectively PER and $RBER$ are recorded.

Of course, in plain BSC mode, the code stream may in real-life applications be protected by forward error correction (FEC) codes. However, since the specified simulations do not use any hybrid ARQ scheme, i.e. one with concatenated FEC and convolutional codes, the comparison between plain BSC and plain ARQ is indeed fair.

There are several dependencies that are of interest. First, how do PER and D_{ARQ} behave with varying burst size L_p and bit error probability BER ? Then, how can PER and BER be related to each other, and what kind of dependency is there between D_{ARQ} and D_{BSC} ? These questions will be answered in the following.

6. RESULTS AND DISCUSSION

As the ARQ protocol guarantees error-free transmission, one gets $RBER_{ARQ} = 0$. It is further trivial to see that $RBER_{BSC} = BER$ since transmission over a BSC is independent of bursts. Transmission delay D_{BSC} and $RBER$ are not functions of L_p . Sending of the specified bit stream over the BSC gives $D_{BSC} = 507$ ms. For an ARQ channel with zero or very low bit error rate, $D_{ARQ} \approx 1.014$ s since also the acknowledgment packet has to be waited for. This is because the special case of transmission time equality between forward and reverse channel was considered.

Considering the ARQ scheme, the packet error ratio increases non-linearly with growing BER (Fig. 4), $PER \sim \frac{1}{1-BER}$. This is also what one would expect intuitively. The general behavior is consistent but the actual value of PER depends strongly on the packet size L_p . For large packet lengths, the increase is more dramatic than for small ones. What one can see additionally is that the dependency of PER changes from linear for low BER s to a non-linear dependency for high BER s, $PER \sim \frac{L_p-1}{L_p}$. As a general rule of thumb, one can formulate that the product of the packet size and the channel's bit error rate has to be lower than 0.1 to yield moderate retransmission rates and to avoid a strong rise of PER :

$$L_p \cdot BER < 0.1. \quad (4)$$

The measured transmission delays for the ARQ scheme are depicted in Fig. 5. To keep track of the data, not the whole range of D_{ARQ} is displayed. For bit error rates over 0.001 and adequate burst lengths, D_{ARQ} quickly reaches values over 33 min. The delay shows a strongly non-linear dependency on both bit error rate and packet length. The latter statement is only valid for sufficiently large values of BER . Also here, the rule as formulated in Ineq. 4 is satisfied quite accurately. The main difference to the PER curve, however, is that the increase of the delay dependent on L_p is much stronger.

To be able to compare the simulation results to the theoretical results as presented in Sec. 4, one has to look at the ARQ channel delay as a function of PER which in turn is a function of both L_p and BER . In Fig. 6, the values of PER have been simulated by using different burst lengths. The graph shows clearly the dependency $D_{ARQ} \sim \frac{1}{1-PER}$.

Something similar can be seen in Fig. 7 where the values of PER have been computed by using different bit error rates. The simulations show further that Eq. 2 does not account for the transmission time of the reverse channel. Because of the special case of equal transmission times for forward and reverse channel, a difference of a factor of two is obtained here. So, Eq. 2 must actually be scaled by a factor larger than one. Since the

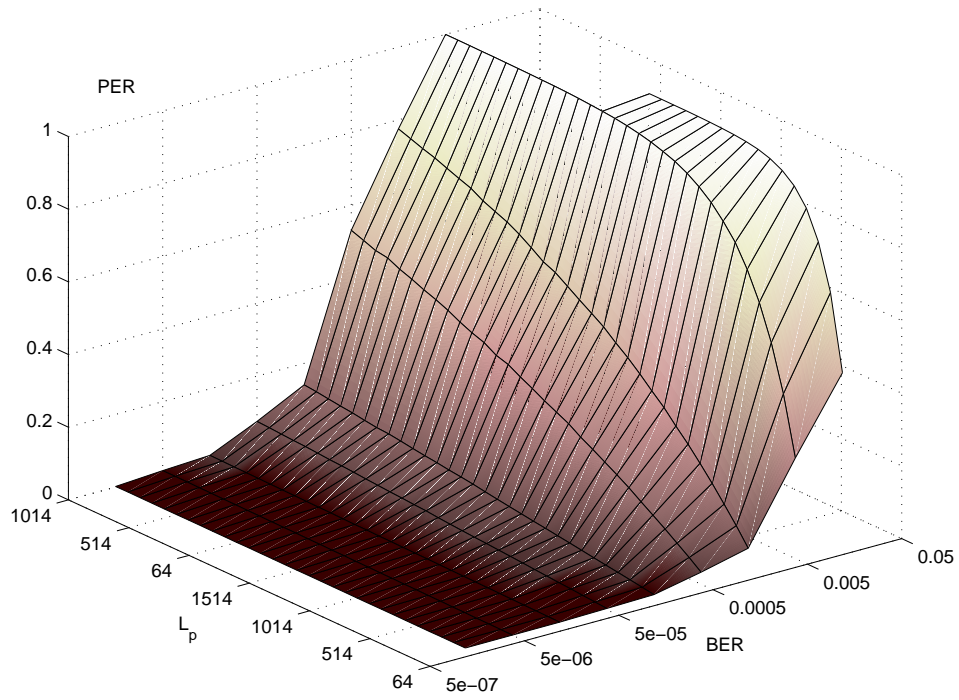


Figure 4. Packet error ratio on an ARQ channel. Some values of PER close to one have not been computed due to simulation duration.

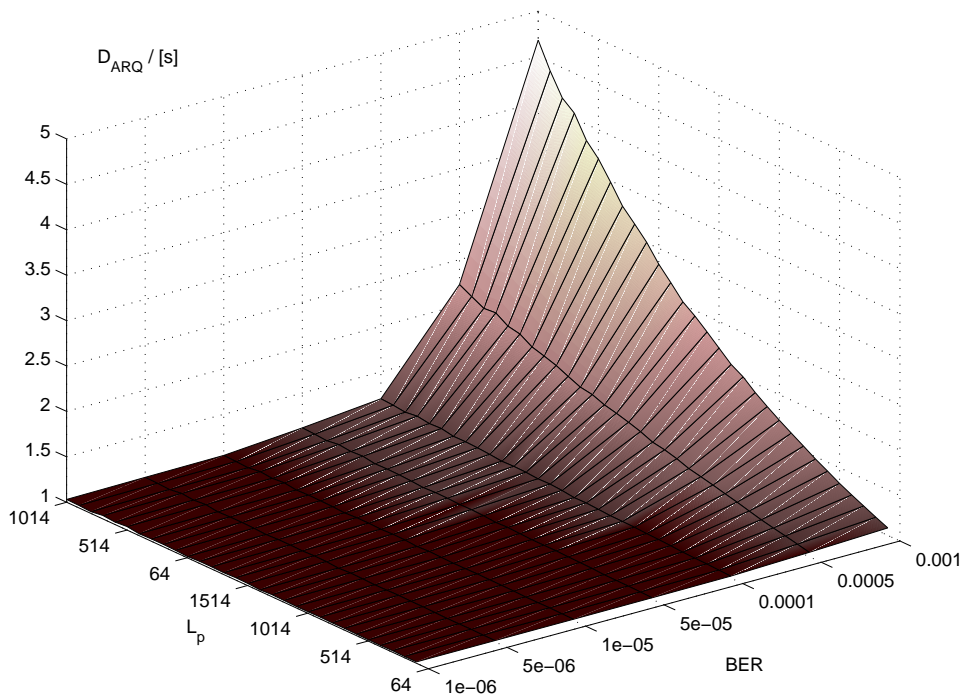


Figure 5: Transmission delays for an ARQ channel.

curves' behavior in both last mentioned graphs is equal, the influence of L_p and BER can be related to each other in an inverse-proportional relation. This justifies also the existence of Ineq. 4.

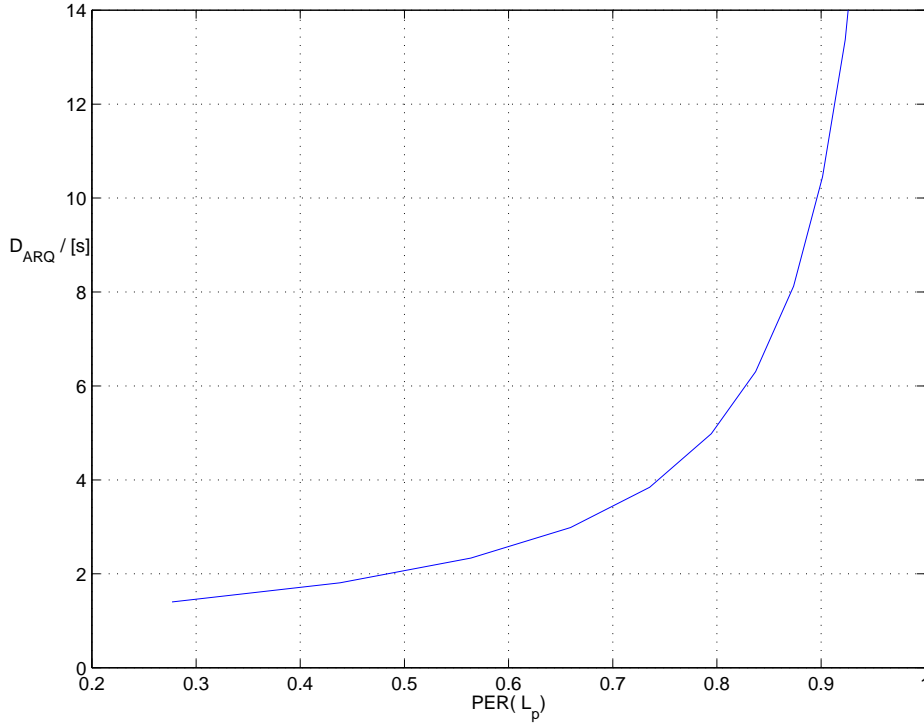


Figure 6: Behavior of D_{ARQ} , additional parameter: $BER = 0.005$.

The comparison between Fig. 2 and the simulation results are shown in Fig. 8. The first striking fact is that the values recorded in the experiments are below the lower bound as derived by Perkis and Cardelo. For small PER , however, the curves' behavior is quite equal, whereas for $PER > 0.8$, the simulations show that a retransmission rate close to one is already reached for much smaller $RBER$ values than the formula of Perkis and Cardelo might suggest. This behavior is consistent for different packet lengths of which two are displayed.

To explain this disparity, a closer look at Ineq. 3 is required. There, PER and L_p are two independent parameters that affect the characteristic of $RBER$ separately. This means that, in Fig. 2, the dependency of PER on L_p , proven in these simulations as varying from linear to non-linear, is not accounted for. To reflect the influence of L_p on PER , one generally has to calculate with a lower value for the retransmission probability than suggested. It is thus proposed to rewrite Ineq. 3 and replace it with

$$RBER \approx \frac{1}{L_p} \frac{c_{ARQ} \cdot PER}{1 - c_{ARQ} \cdot PER}, \quad (5)$$

where c_{ARQ} is a correction factor. This equation is an approximation for the simulation results and also displayed in Fig. 8. The correction factor depends on the ARQ mode used. Its possible range is $0 < c_{ARQ} < 1$ with typical values being around $c_{ARQ} \approx 0.8$ for send-and-wait mode.

Another important change in Eq. 5 is that it is not longer a lower bound but an approximation. Averaging of all simulation results over 100 different error patterns allows to specify typical values of $RBER$ much more accurately than it is theoretically derivable. This makes it also possible to remove the differentiation with respect to PER in Ineq. 3. It is superfluous in most cases anyway as already a packet length of 64 gives $PER = 0.985$.

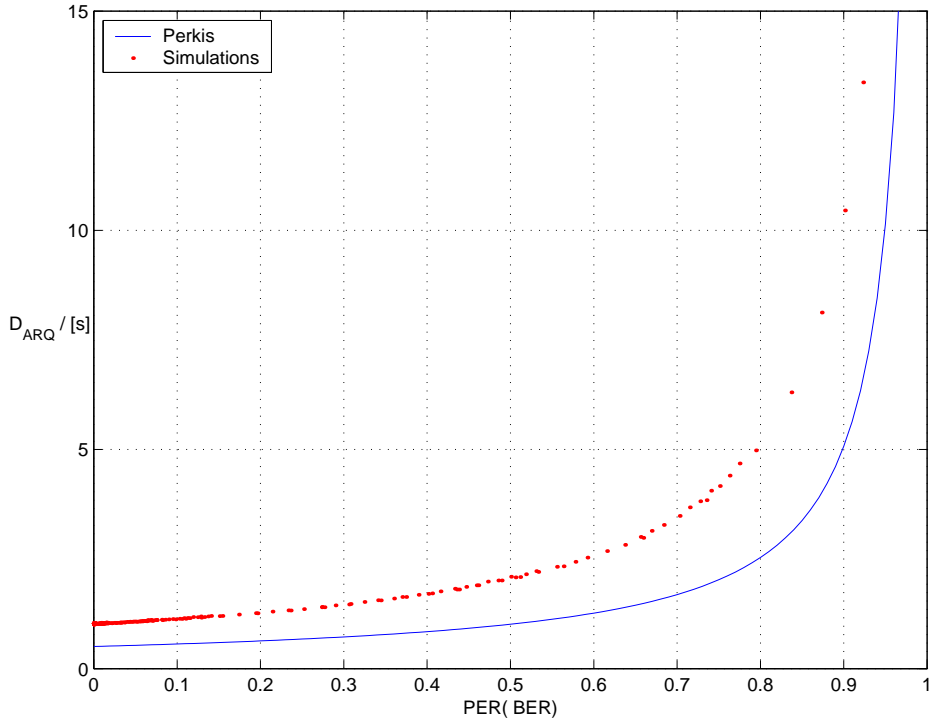


Figure 7: Behavior of D_{ARQ} , supported by all values of L_p to be able to show the curve also for large PER values.

7. CONCLUSIONS AND OUTLOOK

The performance of two channel models, a plain binary symmetric channel and one with additional send-and-wait ARQ scheme, were simulated and compared. The results show a strongly non-linear dependency of the total transmission delay on bit error rate on the channel and packet length. Both effects superpose each other and hence lead to a dramatically increased transmission delay already for moderate bit error rates and packet length values. This gives reason to believe that, under the assumption that the receiver is capable of coping with residual bit errors, it is advantageous to transmit large bit streams like audiovisual and multimedia data without reverse channel to save transmission time.

Moreover, the simulation results were compared to theoretical derivations by Perkis and Cardelo. It could be shown theoretically and by means of simulations that the formula for the transmission delay of an ARQ channel (Eq. 2) disregards the reverse channel and the delay needed to transmit acknowledgment packets. The simulations also suggest the modification of the inequality of Perkis and Cardelo for computation of the residual bit error rate to a more accurate approximation and the introduction of an adjusting factor.

For future research, it is proposed to do the same investigations for an FEC protected code stream and a hybrid ARQ scheme. It is also of interest how other faster ARQ techniques like go-back- N and selective-repeat can be related to the results of this paper.

ACKNOWLEDGMENTS

For inspiring discussion, I especially thank my colleague Tung Thanh Pham. I am further indebted to Professor Tor Audun Ramstad for helpful suggestions and advice.

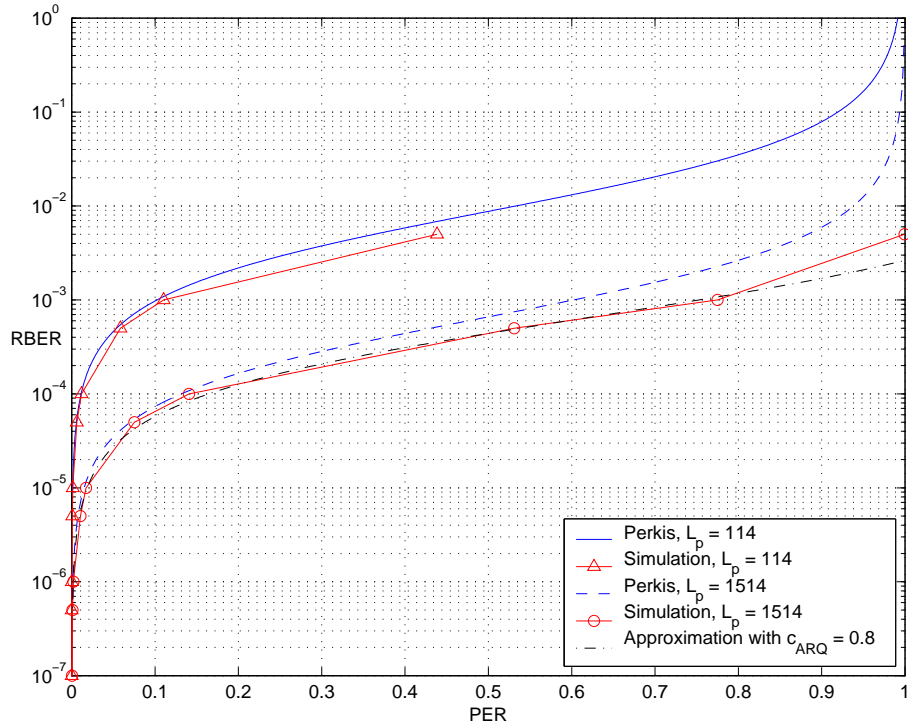


Figure 8: Comparison results by Perkis and Cardelo — own simulations.

REFERENCES

1. B. H. Walke, *Mobile Radio Networks*, John Wiley & Sons, Chichester (England), 1999.
2. F. Pollara and L. Ekroot, "Analysis of automatic repeat request methods for deep-space downlinks," Tech. Rep. 42-122, Jet Propulsion Laboratory, Pasadena (USA), Aug. 1995.
3. A. Perkis and D. G. Cardelo, "Transmission of still images over noisy channels," in *ISSPA*, (Brisbane (Australia)), Sept. 1999.