IPN Progress Report 42-194 • August 15, 2013

Problem Formulation and Analysis of the 1-Hop ARQ Links

Kar-Ming Cheung*

ABSTRACT. — In space communications, standard link analysis assumes that messages are sent once. For a communication link that uses an error-correction coding scheme, bit-error rate (BER) or frame-error rate (FER), and link margins are common metrics that characterize the quality of a link, and they are used to determine the supportable data rate. With the advent of Automatic Repeat-reQuest (ARQ) protocols, when messages are corrupted during transmission, they can be resent multiple times automatically until they are correctly received and acknowledged. The concept of BER, FER, and link margin cannot be directly applied, and the link analysis approach for ARQ links needs to be reexamined.

In this article, we provide the problem formulation and extend the standard link analysis approach to a 1-hop ARQ link (no routing), under the assumption that code-block errors occur independently as in the case of space communications. We develop analytical models that describe the statistical behavior of standard 1-hop ARQ links. By integrating these analytical ARQ protocol models into the standard link analysis, we bypass the need to simulate or emulate the ARQ protocol operations, and generate relevant statistics on effective data rate, effective throughput, latency, and FER.

I. Introduction

In space communications, standard link analysis assumes that messages are only sent once.[1] Also, when an error-correction coding (ECC) scheme is used, to ensure that the decoder at the receiving side does not misinterpret an erroneous code block to be a correct one, the ECC scheme is typically designed with powerful error-detection capability such that the undetected error probability is negligible [1]. Under these assumptions, bit-error rate (BER) or frame-error rate (FER) and link margin are common metrics that characterize the quality of a link, and they are used to determine the supportable data rate. Common BERs of choice in many NASA links (which we can find in many requirements documents) are 10^{-6} and 10^{-8}. For a given BER, there is a corresponding signal-to-noise ratio threshold — denoted by SNR_{Th} or $\left(E_b/N_0\right)_{Th}$ — which is a characteristic of the underlying coding scheme used. In

* Communications Architectures and Research Section.

[1] We denote this kind of link as a "send-once" link.

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

the standard link analysis and planning process, the link margin policy specifies the link margin M to ensure that the random signal-to-noise ratio (SNR) fluctuations would have a small likelihood to cause the received E_b/N_0 to dip below the coding threshold $\left(E_b/N_0\right)_{Th}$. Link analysis and the corresponding link margin policy are typically expressed in logarithmic scale (in decibels, or dB). The supportable data rate R_b is determined such that the received E_b/N_0 exceeds the sum of $\left(E_b/N_0\right)_{Th}$ and M (in dB) to ensure reliable communication. Typical link margins of choice are 3 dB, 2-σ,[2] and 3-σ. In the above standard link analysis process, the underlying assumption is that data that are corrupted in the channel are nonrecoverable. Therefore, the link analysis approach is to apply link margin to maintain a maximum tolerable error rate (BER or FER) so as to ensure data integrity and to minimize data gaps in the received data. On the receiving side, code frames that are undecodable are discarded.

With the advent of Automatic Repeat-reQuest (ARQ) protocol, when data are corrupted during transmission, messages can be resent multiple times until they are received and acknowledged. Much work has been done in the performance analysis of ARQ protocols in the wireless communication areas. Throughput and latency analyses can be found in early papers [2,3] under the assumption that code-block errors occur independently. To analyze wireless communication channels that are characterized by fast fading and bursty errors, recent literature introduces channel models that assume an error process that is not random, and is modeled as a Markovian process [4,5,6,7].

In this article, we limit the scope to 1-hop space communication scenarios (no routing) that assume independent code-block errors. For now, let us consider the case that the acknowledgment process (the reverse channel and the error-checking mechanism) is lossless. The ARQ link is therefore "error-free" in the sense that a data frame will eventually be successfully delivered (at the first transmission or a subsequent retransmission). However, the penalties for the ARQ link are i) increased latency for retransmission, and ii) reduced link efficiency (measured in higher power or lower data rate) to accommodate the retransmitted data frames.

Thus, to analyze the error-free ARQ link, we need to consider:

(1) Transmission latency in some statistical sense (e.g., maximum latency, mean latency, etc.).

(2) Effective data rate R_{eff} in terms of the net data throughput, discounting the portion of the bandwidth that accommodates retransmissions.

(3) Link margin for the error-free ARQ link.

The concept of effective data rate is also applicable to send-once links. Assuming the smallest data unit to be a frame, and denoting P_{bk} as the FER. The effective data rate R_{eff} in terms of the amount of reliable data available on the received side can be measured as

---

[2] σ is the standard deviation of the Gaussian distribution used in statistical link analysis [12,13].

$$R_{eff} = R_b(1 - P_{bk}). \tag{1}$$

Note that in this interpretation of effective data rate for the send-once link, R_{eff} includes the portion of the data frames that are successfully received. The corrupted data frames are lost and nonrecoverable.

Earlier work in analyzing ARQ methods for deep-space links can be found in [8], which investigates the ARQ links mainly from a coding performance perspective, and considers only the Selective Repeat protocol. This article addresses the ARQ problem from the viewpoint of extending the standard link analysis techniques to the ARQ links, and provides explicit analytical expressions to estimate the supportable effective data rate R_{eff}. It provides a more comprehensive treatment in the following areas: a) taking into account the error probability of the acknowledgment link, b) including Go-Back-N protocol, and c) characterizing overall latency in terms of the latency components of light-time delay, receiver processing time, and time-out mechanism of the ARQ protocol.

This article first discusses the ARQ links that assume no limit on the number of retransmissions. The derivation of the key mathematical expressions for the more complicated truncated ARQ links that allow a finite number of retransmissions is outlined in the Appendix. The rest of the article is organized as follows: Section II describes the link models for ARQ links that use the Selective Repeat protocol and the Go-back-N protocol. Using the average energy argument, we derive the analytical expressions of R_{eff} that are a function of a) the link models of the send-once link (including the error-correction coding model), b) the FER of the acknowledgment link, and c) the protocol used. Section III shows that for an ARQ link, there exists an optimal operating point that maximizes R_{eff} (in terms of the operating SNR). This optimal SNR is observed to be smaller than the SNR used in a send-once link, and generates a higher BER or FER that is typically not acceptable in a send-once link. This suggests that the link should operate at a higher data rate and a higher error rate and without margin, and then use retransmission to compensate for errors and to maximize the net throughput in the expense of additional latency. We illustrate this result using two examples of Proximity-1 ARQ protocols using a convolutional code and a low-density parity-check (LDPC) code, respectively. Section IV discusses the latency behavior of the ARQ links, and Section V provides the concluding remarks.

II. Link Model for an ARQ Link

Let us consider the above problems of sending data through an ARQ link by counting the discrete quantities of energy required to transmit and retransmit a code block until it is correctly received. The assumptions for this analysis are as follows: a) the ECC scheme has negligible decoder error probability; b) the link condition does not change during the transmission and retransmission process; and c) there is no limit on the number of retransmissions to get the code block through (for now). Let us denote E_{bk} as the energy required to transmit a code block through the return link channel. We will show in subsequent sections that the straightforward link analysis techniques can be extended to evaluate the ARQ links.

A. Lossless Acknowledgment Channel

Let us consider a lossless acknowledgment channel. For a long-haul link like the deep-space link with long latency, we assume the use of the more complicated Selective Repeat ARQ scheme, which sends one code block per retransmission. This is the same case as discussed in [8]. In this case, the probability that the code block is successfully transmitted in the first trial with energy E_{bk} is $1 - P_{bk}$. If the first trial is not successful, then the code block has to be retransmitted. If successful, the code block uses a total energy of 2 E_{bk}, and the probability of successful transmission in two trials is $P_{bk}\left(1 - P_{bk}\right)$. Using this reasoning, the average energy required to successfully transmit a code block is therefore

$$\left(1 - P_{bk}\right)E_{bk} + 2P_{bk}\left(1 - P_{bk}\right)E_{bk} + \cdots + nP_{bk}^{n-1}\left(1 - P_{bk}\right)E_{bk} + \cdots = \frac{E_{bk}}{1 - P_{bk}}. \qquad (2)$$

Using the above argument, the effective data rate R_{eff} turns out to be $R_{eff} = R_b\left(1 - P_{bk}\right)$, which has the same form as in the case of no retransmission. This result is consistent with the findings in information theory that state that feedback does not increase channel capacity [8,9].

For a proximity link like the Mars relay link that uses the Proximity-1 protocol, we consider the use of the simple Go-Back-N ARQ protocol, where up to N code blocks are sent for each retransmission.[3] The choice of N depends on data rate and latency for acknowledgment. Typical values of N are 2 and 4. In this case, the worst-case energy required to transmit a code block reliably in the n-th retransmission is $\left(nN + 1\right)E_{bk}$, and the average energy is given by

$$\begin{aligned}\left(1 - P_{bk}\right)E_{bk} + \left(N + 1\right)P_{bk}\left(1 - P_{bk}\right)E_{bk} + \cdots + \left(nN + 1\right)P_{bk}^{n-1}\left(1 - P_{bk}\right)E_{bk} + \cdots \\ = \frac{E_{bk}}{1 - P_{bk}}\left(1 + \left(N - 1\right)P_{bk}\right).\end{aligned} \qquad (3)$$

Thus, the effective data rate R_{eff} is given by

$$R_{eff} = R_b \frac{1 - P_{bk}}{1 + \left(N - 1\right)P_{bk}}. \qquad (4)$$

Note that for $N = 1$, the R_{eff} of the Go-Back-N ARQ protocol performance is the same as the Selective Repeat, and is of the same form as the R_{eff} for the send-once link as defined in Equation (1). Also, R_{eff} for Selective Repeat $\left(N = 1\right)$ is always greater than that of Go-Back-N for $N > 1$.

B. Lossy Acknowledgment Channel

For space communications, there are scenarios when the acknowledgment channel cannot be considered as lossless. For example, while performing a software upload from ground to spacecraft, or from orbiter to lander, the acknowledgment link can be operating at reasonably low margin such that the code-block errors are not insignificant. We assume that the acknowledgment link also uses an ECC scheme with negligible undetected error probability.

[3] This is the worst case when no two consecutive frames in error are less than N frames apart.

For nonzero error probability P_{ack}, the analysis can be more complicated as there can be different methods to do acknowledgment, and each can lead to a different efficiency analysis. To analyze this problem, we provide a simple upper bound and a lower bound that can be applied to the different acknowledgment methods. The upper bound is trivial; by setting $P_{ack} = 0$, the upper bound is equivalent to the case of lossless acknowledgment and is given by Equation (4).

For the lower bound, note that the Go-back-N scheme, which is a sliding-window protocol, assumes a maximum receiver window size M that is the number of code blocks the receiver may receive before returning an acknowledgment to the sender. Different methods of cumulative acknowledgment handshake can be used to mitigate the effects of lost or erroneous code blocks previously transmitted in the acknowledgment channel. By setting $M = 1$, no cumulative acknowledgment can be used and this reduces to the conservative yet simple case that a code block delivery can only be concluded with successful consecutive transmissions in both the return link direction and the acknowledgment link direction, with probability $\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)$.

Based on this conservative setup, for the case of Go-Back-N, the energy required and the probability of occurrence for n retransmission are summarized in Table 1.

Table 1. Energy and probability of occurrence of retransmission.

Number of Retransmissions	Total Energy	Probability of Occurrence
0	E_{bk}	$\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)$
1	$\left(N + 1\right)E_{bk}$	$\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\left(1 - \left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\right)$
2	$\left(2N + 1\right)E_{bk}$	$\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\left(1 - \left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\right)^2$
⋮	⋮	⋮
n	$\left(nN + 1\right)E_{bk}$	$\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\left(1 - \left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\right)^n$
⋮	⋮	⋮

Using the same approach as in Section II.A, the average energy required to transmit a code block using Go-Back-N ARQ protocol is

$$E_b\left(1 + \frac{N\left(1 - \left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\right)}{\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)}\right). \tag{5}$$

The corresponding lower bound of the effective data rate R_{eff} is therefore

$$R_b\left(1 + \frac{N\left(1 - \left(1 - P_{bk}\right)\left(1 - P_{ack}\right)\right)}{\left(1 - P_{bk}\right)\left(1 - P_{ack}\right)}\right)^{-1}. \tag{6}$$

Note that for $P_{ack} = 0$, the upper bound (Equation [4]) and the lower bound (Equation [6]) are identical.

C. Tying Everything Together — Express R_{eff} in Terms of R_b, N, P_{ack}, and the Code

In this article, we assume R_b to be a tunable and continuous parameter. Note that P_{ack} is a characteristic of the acknowledgment link, and is independent of the return link parameters of R_b and P_{bk}. To simplify the discussion, we regard P_{ack} to be a constant in subsequent analysis.[4] For the sake of simplifying the discussion, let us assume an instantaneous link resource.[5] E_b/N_0, R_b, and C are related as follows:

$$\frac{E_b}{N_0} R_b = C. \tag{7}$$

Let $f(\,.\,)$[6] denote the FER performance curve of the code used in the return link, and from Equation (6), we get

$$P_{bk} = f\left(\frac{E_b}{N_0}\right) = f\left(\frac{C}{R_b}\right). \tag{8}$$

Substituting Equation (7) into Equation (5), we can express the lower bound of R_{eff} as a function of R_b, namely

$$R_{eff} = R_b \left(1 + \frac{N\left(1 - \left(1 - f\left(\frac{C}{R_b}\right)\right)\left(1 - P_{ack}\right)\right)}{\left(1 - f\left(\frac{C}{R_b}\right)\right)\left(1 - P_{ack}\right)} \right). \tag{9}$$

Note that in the above expression, R_b is a tunable parameter, and is not an output of the standard link analysis. The effective data rate R_{eff} is a function of R_b, N, P_{ack}, and the code, characterized by the function $f(\,.\,)$. The above analysis assumes that there is no limit on the number of retransmissions for the code blocks. To minimize delays and buffer sizes in practice, truncated ARQ protocols have been widely adopted to limit the maximum number of retransmissions. We describe in the Appendix the extension of link analysis techniques for links with truncated ARQ protocols, where there is a counter that restricted the number of retransmissions of a code block to be less than a predetermined value of K.

III. Finding the E_b/N_0 That Maximizes R_{eff}

Note that Equation (8) allows one to evaluate the R_b that maximizes R_{eff} for a given link resource C, and this in turn allows one to compute the optimal E_b/N_0[7] for the given P_{ack}, Go-back-N protocol, and the code of the return link. In the following subsections, we will investigate the ARQ link performance of the current Proximity-1 protocol using the (7,1/2) convolutional code only, and two hypothetical ARQ protocols using the AR4JA LDPC

---

[4] As in the case of P_{bk} of the return link, P_{ack} can be expressed using the FER performance curve of the code used in the forward link.

[5] C is in fact the data signal-to-noise ratio P_d/N_0, which can be computed from standard link analysis. To simplify the illustration, we assume C to be a constant, which is a good assumption for medium- and high-rate space links. For low-rate communications, C and P_{bk} can be complicated functions of R_b. This formulation can also be applied to links with a slow-fading channel. In this case, C is the slowly changing P_d/N_0, which can be provided by external simulation and analysis.

[6] E_b/N_0 of $f(E_b/N_0)$ is expressed as a ratio, not in dB.

[7] From Equation (7).

codes, with rate 1/2 and rate 2/3, and block size of 1024. For the purpose of illustration, we assume $P_{ack} = 0$; also, we assume $K \to \infty$.

A. ARQ Link Analysis for Proximity-1 Protocol with (7,1/2) Convolutional Code

The Mars Exploration Rover Proximity-1 protocol in operation uses the (7,1/2) convolutional code with frame size 908 bits for a data rate less than or equal to 8 kbps, and frame size 2040 bits for a data rate above 8 kbps. To obtain the FERs of the (7,1/2) convolutional code, we interpolate and extrapolate the codeword error rate (CWER) versus E_b/N_0 performance using Figure 4.7 of [11], and use the fourth-order polynomial curve-fit to construct the FER performance functions for frame sizes of 908 and 2040 bits, respectively (x is expressed in dB):

$$P_{908}[x] = \begin{cases} 1 & x < 1.8 \\ e^{20.4774-22.0216x+7.6164x^2-0.9414x^3} & 1.8 \le x < 10.0 \\ 0 & x \ge 10.0 \end{cases}$$

$$P_{2040}[x] = \begin{cases} 1 & x < 1.87 \\ e^{1.1282-0.1244x-0.3708x^2-0.0086x^3} & 1.87 \le x < 10.0 \\ 0 & x \ge 10.0 \end{cases}$$

Assuming C = 1000 (C can be any value for this analysis), we plot R_{eff} as a function of R_b for Selective Repeat, Go-Back-2, and Go-Back-4 ARQ protocols, for the cases of (7,1/2) convolutional code with frame size of 908 bits and 2040 bits (Figure 1 and Figure 2, respectively).

Note that for the (7,1/2) convolutional code, the $(E_b/N_0)_{Th}$ for BER of 10^{-6} and 10^{-8} are 5.1 dB and 6.0 dB, respectively. For frame size of 908 bits, Figure 1 indicates that the optimal operation point for Selective Repeat, Go-back-2, and Go-Back-4 are 2.71 dB, 3.30 dB, and 3.66 dB. The gain of using ARQ compared to a link without retransmission is approximately 1 to 3 dB depending on the choices of BER operation point and ARQ protocol. Similarly, for frame size of 2040 bits, Figure 2 shows that the optimal operation point for Selective Repeat, Go-back-2, and Go-Back-4 are 3.17 dB, 3.43 dB, and 3.69 dB, plus the margin M required for standard "sent-once" links. The gain of using ARQ is approximately the same.

B. ARQ Link Analysis for Proximity-1 Protocol Using LDPC Codes

We choose two AR4JA LDPC codes, rate 1/2 and rate 2/3 codes with block size 1024 bits. We use curve-fit on the FER versus E_b/N_0 data to construct the following piece-wise functions (x is expressed in dB):

$$P_{12}[x] = \begin{cases} 1.0 & x < 0.1 \\ 1.031 - 0.4433x + 1.8135x^2 - 2.5016x^3 & 0.1 \le x < 0.7 \\ e^{-1.4455+5.3661x-5.2249x^2-0.1065x^3} & 0.7 \le x < 10.0 \\ 0.0 & x \ge 10.0 \end{cases}$$

$$P_{23}[x] = \begin{cases} 1.0 & x < 1.05 \\ 0.6954 + 0.1879x + 0.6298x^2 - 0.5045x^3 & 1.05 \le x < 1.6 \\ e^{-15.9358+24.9398x-11.2898x^2+1.0549x^3} & 1.5 \le x < 5.0 \\ 0.0 & x \ge 5.0 \end{cases}$$

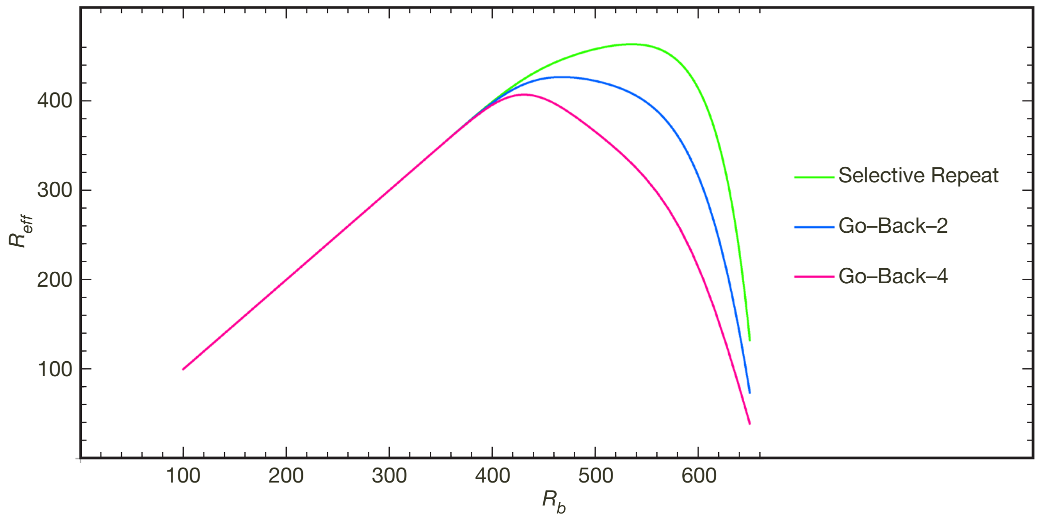

Figure 1. R_{eff} versus R_b for (7,1/2) convolutional code (frame size = 908 bits).

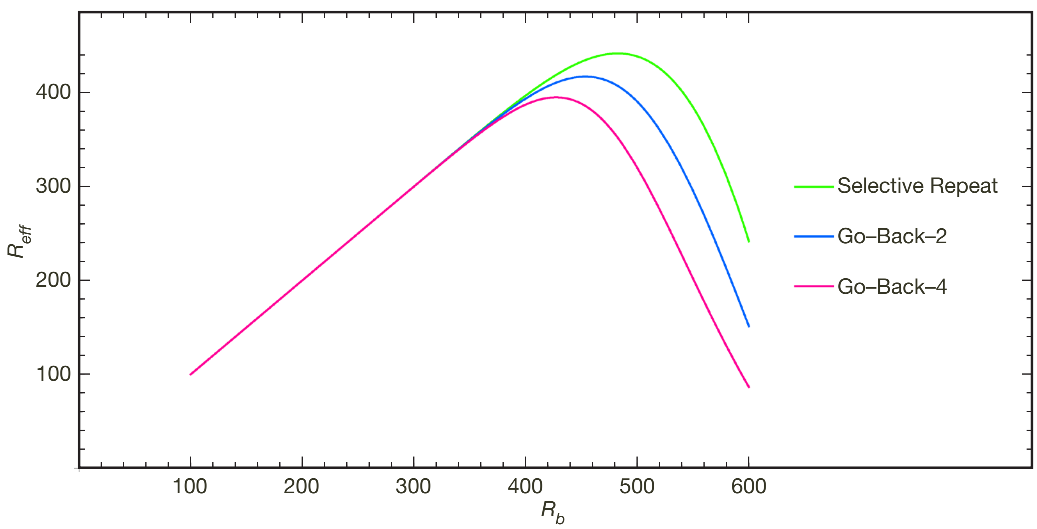

Figure 2. R_{eff} versus R_b for (7,1/2) convolutional code (size = 2040 bits).

Assuming C = 1000[8] (C can be any value for this analysis), we plot R_{eff} as a function of R_b for Selective Repeat, Go-Back-2, and Go-Back-32 ARQ protocols, for the cases of LDPC rate 1/2 code and LDPC rate 2/3 code (Figure 3 and Figure 4, respectively).

For the case of rate 1/2 code, the $(E_b/N_0)_{Th}$ for BER of 10^{-6} and 10^{-8} are 2.0 dB and 2.2 dB, respectively. The maximum effective data rates for Selective Repeat ARQ, Go-Back-2 ARQ, and Go-Back-32 ARQ occur at E_b/N_0 = 1.32 dB, E_b/N_0 = 1.40 dB, and E_b/N_0 = 1.66 dB, respectively. The gain of using ARQ compared to a link without retransmission is approximately 0.35 to 0.9 dB, plus the margin M required for standard send-once links.

---

[8] Defined in Section II.C.

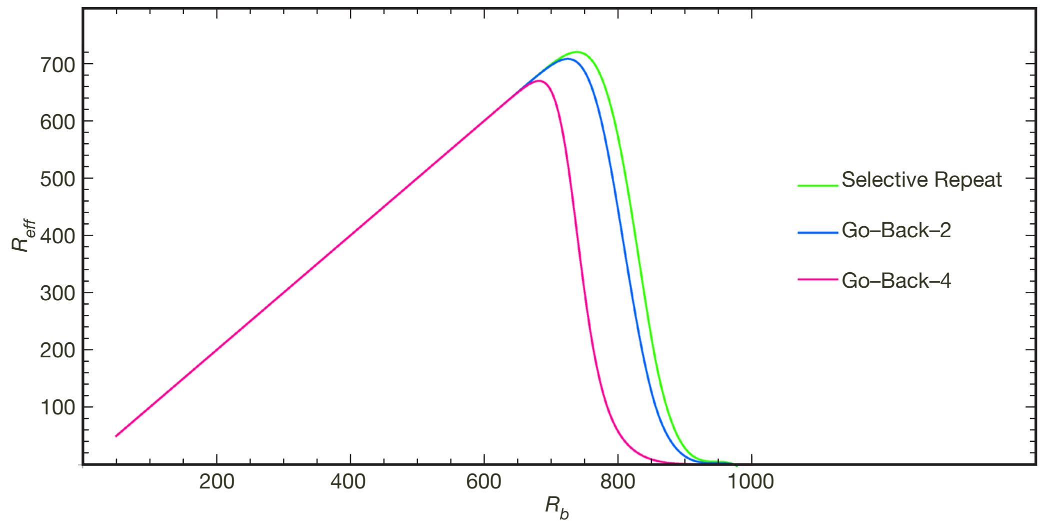

Figure 3. R_{eff} versus R_b for LDPC rate 1/2 code (size = 1024 bits).

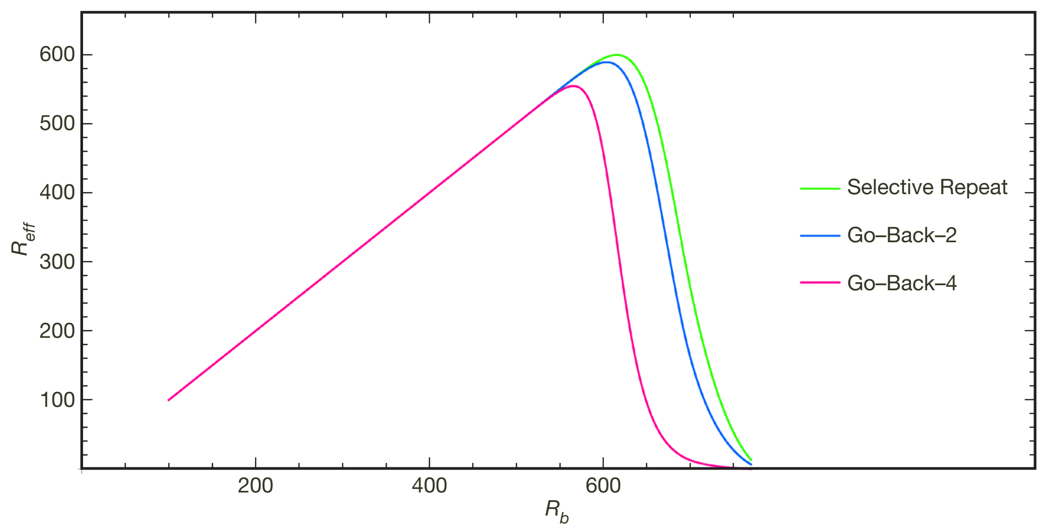

Figure 4. R_{eff} versus R_b for LDPC rate 2/3 code (size = 1024 bits).

For rate 2/3 code, the $\left(E_b/N_0\right)_{Th}$ for BER of 10^{-6} and 10^{-8} are 2.8 dB and 3.1 dB, respectively. The maxima for Selective Repeat ARQ, Go-Back-2 ARQ, and Go-Back-32 ARQ occur at $E_b/N_0 = 2.11$ dB, $E_b/N_0 = 2.19$ dB, and $E_b/N_0 = 2.47$ dB, respectively. The gain for the rate 2/3 LDPC code is approximately 0.31 to 1.0 dB, plus the margin M required for standard send-once links.

Thus, for the more powerful LDPC codes with a steeper slope, the benefit of ARQ is not as noticeable as in the case of the convolutional code.

C. Observations

Note that the optimal E_b/N_0 for each code results from the FER versus E_b/N_0 (an inherent characteristic of the code) only, and is independent of C and R_b. Thus, we arrive at the following interesting observations:

(1) For ARQ links, for every error-correcting code and ARQ protocol chosen, there exists an optimal operational E_b/N_0 that maximizes the effective data throughput.

(2) The above approach that maximizes the effective data rate R_{eff} provides a simple strategy to establish the transmission data rate R_b for an ARQ link.

(3) For a given forward error-correcting code, the E_b/N_0 that maximizes R_{eff} using ARQ can be much lower than the E_b/N_0 that meets the BER requirement (10^{-6} or 10^{-8}) and when not using ARQ.

(4) For a good code with steep FER versus E_b/N_0 performance curve (for example, LDPC codes), the performance of using an ARQ protocol is not as substantial.

(5) When ARQ is used, one can design the link with no link margin or a much smaller link margin M, as we know that a code block will eventually be corrected, transmitted, and acknowledged (at the expense of latency). This further improves the operation efficiency of the link.

IV. Discussion on Latency

In this section, we consider the simplified case of space communications where code-block errors can be regarded as independent. Also, we assume that when either or both of the code block and acknowledgment message are in error, the transmitter would wait for a predetermined time T_{out} before retransmitting the code block.[9] For a well-designed ARQ system, $T_{out} \geq 2T_c + \Delta_R$, where T_c denotes the one-way light time and Δ_R denotes the receiver processing time to determine if the code block is correctly decoded and to send an acknowledgment. As discussed in Section II.B, there can be different ways to respond to missing acknowledgment messages and to those that are received and not decodable, resulting in different latency respond time to retransmit. To simplify the problem, we assume that the transmitter always retransmits after time T_{out} if it does not receive an acknowledgment message, or if it receives an undecodable acknowledgment message. The code block transmission timeline, the acknowledgment message receiving timeline, and the processing latencies are shown in Figure 5.

Using this simplified assumption, the latency of an ARQ link follows the discrete geometric distribution

$$\text{Prob}\left[\text{latency} = T_c + iT_{out}\right] = \theta(1-\theta)^i \text{ for } i = 0, 1, 2, \cdots, \tag{10}$$

where $\theta = (1 - P_{bk})(1 - P_{ack})$, which is the probability that the code block is successfully sent and acknowledged, and the mean latency as observed by the receiver is computed to

[9] For Go-Back-N, this corresponds to the conservative case that results in the lower bound of Equation (6) when no cumulative acknowledgment can be used.

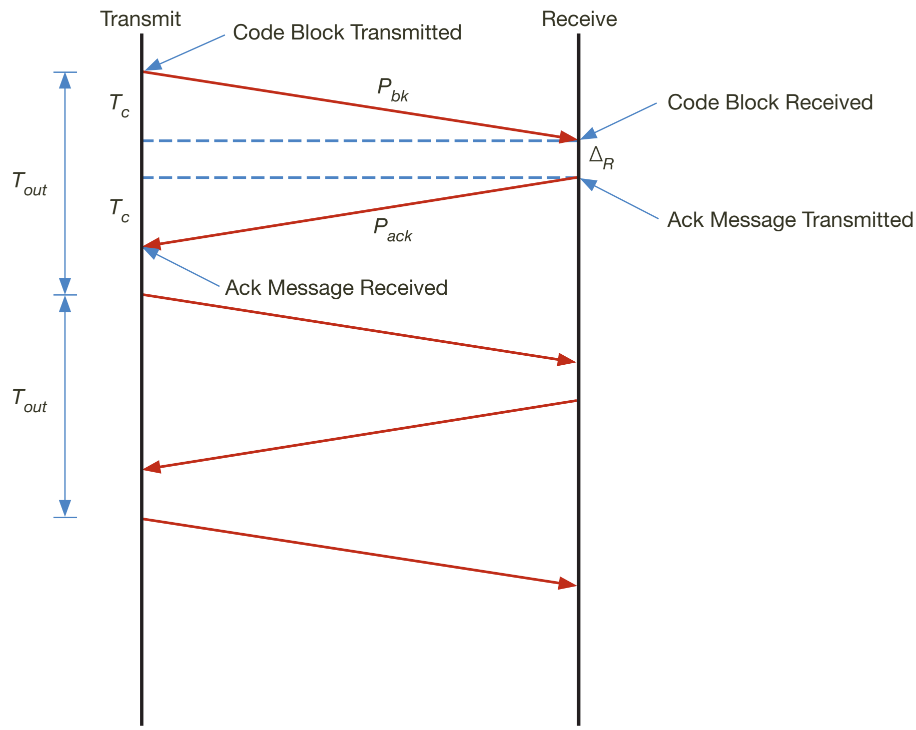

Figure 5. Simplified transmission and receiving timeline.

be $T_c + T_{out}\frac{1-\theta}{\theta}$.[10] Thus, in an average sense, the additional latency cost of an ARQ link compared to a "send-once" link is $T_{out}\frac{1-\theta}{\theta}$.

V. Concluding Remarks

In this article, we provide the problem formulation and extend the standard link analysis approach to the ARQ links. To measure the efficiency of an ARQ link, we introduce the concept of effective data rate that only includes the net data that are correctly sent, and discount the portion of the bandwidth that accommodates retransmission. Based on this definition and counting the average energy to transmit a code block, we develop the analytical models of effective data rate, latency, and FER of standard ARQ protocols. We then use the standard link analysis approach to drive the ARQ analytical protocol models to compute the optimal signal-to-noise ratio operating point, effective data rate, latency, and FER of the link, which are quantities of interest in most communication architecture studies. This ARQ link analysis approach bypasses the need to simulate or emulate ARQ protocol procedures.

Acknowledgment

The authors would like to thank Fabrizio Pollara (JPL), Loren Clare (JPL), and Eytan Modiano (MIT) for their comments and discussion. The research was supported by NASA's Space Communication and Navigation (SCaN) Program.

[10] Variance = $T_{out}^2\frac{1-\theta}{\theta^2}$.

References

[1] K.-M. Cheung, "On the Decoder Error Probability of Linear Codes," *IEEE Transactions on Communications*, vol. 40, no. 5, pp. 857–859, May 1992.

[2] G. Fayolle, E. Gelenbe, and G. Pujolle, "An Analytic Evaluation of the Performance of the 'Send and Wait' Protocol," *IEEE Transactions on Communications*, vol. 26, no. 3, pp. 313–319, March 1978.

[3] D. Towsley and J. K. Wolf, "On the Statistical Analysis of Queue Lengths and Waiting Times for Statistical Multiplexers with ARQ Retransmission Scheme," *IEEE Transactions on Communications*, vol. 27, no. 4, pp. 693–702, April 1979.

[4] D. Towsley, "A Statistical Analysis of ARQ Protocols Operating in a Nonindependent Error Environment," *IEEE Transactions on Communications*, vol. 29, no. 7, pp. 971–981, July 1981.

[5] M. Zorzi and R. Rao, "Throughput Analysis of Go-Back-N ARQ in Markov Channels with Unreliable Feedback," *Proceedings of IEEE ICC'95*, vol. 2, pp. 1232–1237, Seattle, Washington, June 18–22, 1995.

[6] M. Zorzi and R. Rao, "On the Use of Renewal Theory in the Analysis of ARQ Protocols," *IEEE Transactions on Communications*, vol. 44, no. 9, pp. 1077–1081, September 1996.

[7] M. Zorzi and R. Rao,"Performance of ARQ Go-Back-N Protocol in Markov Channels with Unreliable Feedback," *Mobile Networks and Applications*, vol. 2, no. 2, pp. 183–193, October 1997.

[8] F. Pollara and L. Ekroot, "Analysis of Automatic Repeat Request Methods for Deep-Space Downlinks," *The Telecommunications and Data Acquisition Progress Report*, vol. 42-122, Jet Propulsion Laboratory, Pasadena, California, pp. 66–83, April–June 1995, article dated August 15, 1995.
http://ipnpr.jpl.nasa.gov/progress_report/42-122/122L.pdf

[9] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[10] Course notes: Data Communication Networks, MIT Course No. 6.263 J/16.37J, Fall 2002.
http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-263j-data-communication-networks-fall-2002/index.htm

[11] Consultative Committee for Space Data Systems (CCSDS), *TM Synchronization and Channel Coding — Summary of Concept and Rationale*, 130.1-G-2, Green Book, Issue 2, November 2012.
http://public.ccsds.org/publications/GreenBooks.aspx

[12] J. H. Yuen, *Deep Space Telecommunications Systems Engineering*, New York: Plenum Press, 1983.

[13] K.-M. Cheung, "Statistical Link Analysis — A Risk Analysis Perspective," *The Interplanetary Network Progress Report*, vol. 42-183, Jet Propulsion Laboratory, Pasadena, California, pp. 1–16, November 15, 2010.
http://ipnpr.jpl.nasa.gov/progress_report/42-183/183B.pdf

Appendix

Link Analysis for Truncated ARQ Links

To impose a maximum delay on the code block reception, some ARQ links limit the number of retransmissions to K. In this case, the ARQ links cannot be considered as error-free. K is determined by the maximum allowable delay and/or buffer sizes that the communication nodes can tolerate. The metric to measure the link quality is similar to the metric that measures the quality of the send-once link; that is, the supportable data rate $R_{b,K}$ that meets the FER requirement $R_{b,K}$. $R_{b,K}$ is interpreted as the effective supportable data rate with a maximum of K retransmissions, and $P_{b,K}$ is defined as the FER with a maximum of K retransmissions.

Using a similar approach in Section II that evaluates the average energy to transmit the code block,[11] $R_{eff,K}$ can be computed as

$$R_{eff,K} = R_b\left(1 + \frac{\delta N(1-\delta^K)}{1-\delta}\right)^{-1} = R_b\left(1 + \frac{\delta N}{1-\delta} - \frac{\delta^{K+1}}{1-\delta}\right)^{-1} \tag{A-1}$$

and $P_{b,K}$ is derived to be

$$P_{b,K} = \delta^{K+1} \tag{A-2}$$

where

$$\delta = 1-(1-P_{bk})(1-P_{ack}) = 1-\left(1-f\left(\frac{C}{R_b}\right)\right)(1-P_{ack}). \tag{A-3}$$

The effective data rate $R_{b,K}$ and the FER $P_{b,K}$ are expressed in the forms shown in Equations (A-1), (A-2), and (A-3) to illustrate the following intuitive remarks:

(1) δ as defined in Equation (A-3) corresponds to the FER as viewed by the transmitter when a code block is either not correctly sent (with probability $f(\frac{C}{R_b})$), or when a code block is correctly sent but the acknowledgment signal fails to reach the transmitter (with probability P_{ack}). Also, $\delta = 1-\theta$.

(2) Equation (A-1) differs from Equation (5) by the last term in the parenthesis, which is always positive, and tends to zero when $K \to \infty$. Thus, the effective data rates for an ARQ link that allows infinite retransmission, for a truncated ARQ link with up to K retransmissions, and send-once link are related by $R_{eff} = R_{eff,\infty} \le R_{eff,K} \le R_b$.

(3) The truncated ARQ link with up to K retransmission buys down the FER of the send-once link from $P_b = f(\frac{C}{R_b})$ to $P_{b,K} = \delta^{K+1}$ at the expense of reduced effective data rate and increased average latency of the code block.

---

[11] In this case, one needs to take into account both the energy required for the successful transmission of the code block, and the energy that is spent on unsuccessful transmissions.

For latency analysis, most engineering applications are only interested in the latency of the code blocks that are successfully transmitted and received. Following a similar argument in Section IV, the discrete probability that the latency equals to T_c, $T_c + T_{out}, T_c + 2T_{out}, \cdots, T_c + KT_{out}$, given that the code block is successfully transmitted, can be expressed as the following conditional probability:

$$\begin{aligned}&\text{Prob}\left[\text{latency} = T_c + iT_{out} \middle| \text{code block successfully transmitted and received}\right]\\ &= \frac{1-\delta}{1-\delta^{K+1}}, i = 0, 1, \cdots, K.\end{aligned} \quad \text{(A-4)}$$

The average latency can be computed to be

$$T_c + \left(K + \frac{1}{1-\delta} - \frac{1+K}{1-\delta^{K+1}}\right)T_{out}$$

and the variance is

$$T_{out}^2 \frac{\delta + \delta^{2K+3} - \delta^{K+1}\left(\left(K+1\right)^2 - 2K\left(K+2\right)\delta + \left(K+1\right)^2\delta^2\right)}{\left(1-\delta\right)^2\left(1-\delta^{K+1}\right)^2}.$$

Another view of coupling truncated ARQ protocol with an error-correction code is to transform an error-correction code into a more powerful code, at the expense of additional data management complexity[12] and latency. We illustrate this concept using the Proximity-1 convolutional (7, 1/2) code with frame size 2040 bits as discussed in Section III.A, and assume the use of Selective Repeat ARQ protocol truncated to K retransmission, and that the acknowledgment link is error-free. We use the same definition of "effective SNR" in [8], and generate the FER versus "Effective SNR" performance curves of the truncated ARQ coding system for $K = 0$ (no retransmission), 1, 2, 4, and 8, and ∞, and the results are illustrated in Figure A-1.

Note that for K approaches ∞,the optimal effective SNR of the ARQ coding system is observed to converge to 3.5 dB , which is the optimal effective SNR computed in Section III.A.[13] The particular case shown in Figure A-1 is consistent with the results shown in [8].

[12] Data management complexity refers to the ARQ data handling efforts on the transmitter side and the receiver side.

[13] In Section III.A, the optimal "nominal SNR" for Selective Repeat protocol is computed to be 3.17 dB at FER = 0.07858. The effective SNR is therefore 3.17 + 0.36 = 3.53 dB.

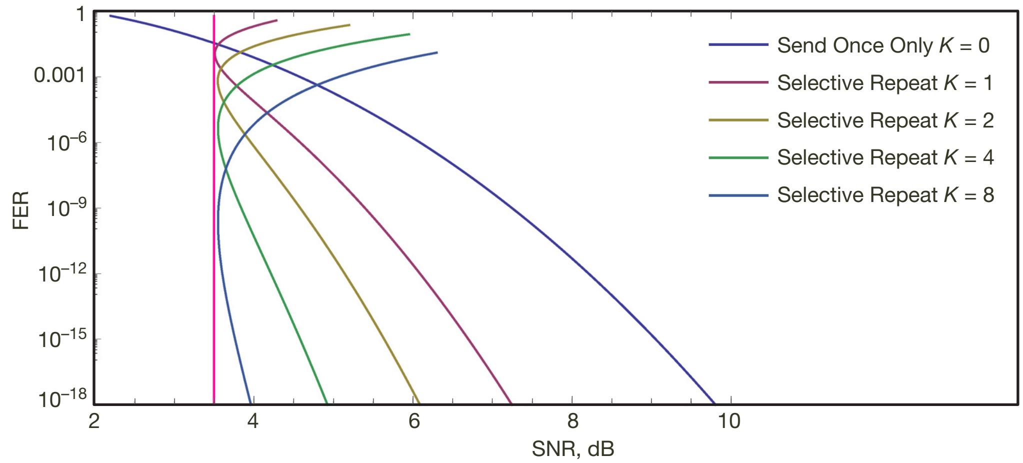

Figure A-1. Performance of truncated ARQ coding system.