Error Control and Energy Consumption in Communications for Nomadic Computing

Michele Zorzi, Member, IEEE, and Ramesh R. Rao, Senior Member, IEEE

Abstract—We consider the problem of communications over a wireless channel in support of data transmissions from the perspective of small portable devices that must rely on limited battery energy. We model the channel outages as statistically correlated errors. Classic ARQ strategies are found to lead to a considerable waste of energy, due to the large number of transmissions. The use of finite energy sources in the face of dependent channel errors leads to new protocol design criteria. As an example, a simple probing scheme, which slows down the transmission rate when the channel is impaired, is shown to be more energy efficient, with a slight loss in throughput. A modified scheme that yields slightly better performance but requires some additional complexity is also studied. Some references on the modeling of battery cells are discussed to highlight the fact that battery charge capacity is strongly influenced by the available “relaxation time” between current pulses. A formal approach that can track complex models for power sources, including dynamic charge recovery, is also developed.

Index Terms—Mobile computing, mobile communications, wireless systems, error control, energy consumption, channel probing, battery modeling.

1 INTRODUCTION

SMALL portable communication devices are an integral element of nomadic computing systems. These mobile devices must rely on limited battery energy to conduct communications over a wireless channel. Since low data rates and high error rates are endemic to the wireless environment, energy efficient error control over the wireless link is a significant part of the effort to develop nomadic computing systems.

The first step is to understand the nature and impact of errors. Voice applications are traditionally supported with a “thin” protocol stack and so the impact of errors on the wireless links on voice transmissions is relatively easy to study. In contrast, control, data, and other multimedia communications are implemented as applications at the upper layers of a protocol stack. Consequently, the impact of physical layer impairments are harder to track, especially since errors on the physical channel are statistically correlated.

To elaborate, note that errors that occur on the physical channel are a function of specific propagation phenomena such as multipath, fading, and user mobility. The raw channel variations are by themselves not directly useful since processes such as coding, interleaving, and power control alter the dynamics of the physical channel errors. Furthermore, for most nonvoice applications, the more relevant quantities are related to block errors, since such applications run on top of a data-link layer that exchanges blocks of data. As we shift our attention to the higher layers, we find that the correlations that arise in bit errors will in principle lead to correlations in the block errors.

It is worth noting that Markov models are useful in dealing with correlated errors. Once the structure of the error process is known, it is possible to study the performance of well known error control schemes such as FEC, ARQ, or Hybrid ARQ. Although such studies have been performed recently [1], [2], [3], [4], they have ignored constraints imposed by limited power sources.

The main theme of this paper is its focus on energy consumption. By quantitatively studying the trade-off between throughput and energy consumption, we show that classic error control protocols perform poorly with respect to energy, and that error control schemes which are suboptimal in the absence of energy constraints may be superior when such constraints are factored in. Furthermore, we identify opportunities to use the memory of the correlated error process to design more energy efficient error control protocols.

2 APPROACH

We focus on two fundamental issues in our study of error control schemes for the wireless channel under energy constraints. We need to understand the error process induced by the physical channel and we need to understand the role of finite energy battery-powered devices.

In this section, we discuss energy constraints and their consequences and, in Section 3, we discuss the channel model. A fundamental conclusion we arrive at is that both dependent errors and energy constraints have a very significant impact on the performance. This emphasizes the need for energy-aware and energy-efficient protocols. The probing protocol is proposed and analyzed in Section 4, and an enhanced version thereof is studied in Section 5. Numerical results and performance comparisons with classic ARQ strategies are discussed in Section 6. An extended analytical model which can track energy dependent transitions is described in Section 7.
2.1 Modeling Power Sources

A significant amount of work on the development of new battery sources has been done in recent years. One performance metric that is commonly reported is the constant power, constant current, and constant load capacity. With most types of cells, these three capacities are not equal, implying that one can get more out of a given cell by draining it in the “right way” (here, the terms “cell” and “capacity” are used with reference to batteries [5], and should not be confused with other usages of the same words in telecommunications). The characterization of cell discharge characteristics under dynamic (load, current, and power) conditions is arguably a more interesting figure of merit in the context of telecommunications. Although the topic appears not to have received much attention yet, one reference [6] indicates that cell capacity is strongly influenced by the available “relaxation time” between current pulses. In [6], the authors studied cylindrical alkaline cells subject to a periodically pulsed current discharge, and found that the cell capacity increases (up to a limit) as the duty cycle decreases or the frequency increases. In general, the actual relationship between how much of the cell capacity is recovered during an off period depends on the cell chemistry and geometry. We found no published reports describing the effect of aperiodic or stochastic discharge conditions on cell capacity.

Thus, some of the useless transmissions performed during an OFF period depend on the cell chemistry and geometry. We found no published reports describing the effect of aperiodic or stochastic discharge conditions on cell capacity.

To further understand the implications of these observations on protocol and system design, note that in any electronic device a number of subsystems will draw energy. Some of these subsystems may draw a steady current while others may draw bursts. It is also true that regulators are often designed to smooth the flow of current to ensure a constant potential. Notwithstanding these prevailing approaches, it seems possible to conceive designs in which a significant portion of the energy drawn follows the most energy efficient profile exhibited by the cell. Thus we may wish to develop protocols that time packet transmissions (and retransmissions) in a manner that maximizes energy efficiency. Such an approach would be most meaningful when the energy necessary for transmission represents a significantly larger portion of the net energy drawn than the portion drawn for other internal housekeeping functions. The use of various sleep modes [7], [8] is a way to minimize the energy that is drawn for internal housekeeping and suggests that the dynamic time profile of the radiated energy from a communication device may indeed represent a significant portion of the total energy drawn.

As a first model for battery sources, we assume that the battery charge is a step function, providing a constant output as long as there is some charge and then dropping to zero when the battery is discharged. With this assumption, the steady state behavior of the scheme described above can be determined without tracking detailed state information. On the other hand, actual battery charge profile may display a more complex functional form, so that a larger set of state variables have to be tracked. Furthermore, there is evidence that some batteries have the ability to recover some of their charge when energy is not drained from them. For example, it is shown in [6] that the capacity of a cell depends on the manner in which it is discharged, and that parameters such as the frequency and the duty-cycle of a pulsed discharge regime have an impact on the battery life. A detailed approach, which can take into account such nonstationary behavior, is outlined in Section 7. The technique is powerful not only because it provides complete statistical information but also because it allows all transitions to depend on the energy state.

2.2 Error Control under Energy Constraints

In the design of error control protocols, one frequently trades off complexity and memory requirements for throughput and delay. Acknowledging energy constraints opens up some new possibilities. In their native modes, classic ARQ protocols, such as Go-Back-N (GBN) or Selective Repeat (SR), recover errors by retransmitting the erroneously transmitted packet regardless of the state of the channel. Thus, even when the channel is bad, these protocols keep retransmitting and expending energy since such action does not degrade the performance. In fact, as soon as the channel becomes good again, packets are received correctly with minimal delay. In the energy constrained environment, repeated retransmission when the channel is bad leads to a loss of energy. Subsequently, when the channel becomes good again the depletion in energy may make transmissions more prone to errors resulting in potentially more transmissions, leading to a catastrophic loss of energy. Therefore, the focus on energy constraints may help capture a crucial aspect of portable system design, one that has gone unmodeled in the past. Error control schemes that are known to be suboptimal in the absence of energy constraints may turn out to be superior when energy constraints are factored in. New strategies that exploit the opportunities the environment presents may also emerge [9].

The point here is that persistence may not be desirable when energy consumption is a major concern. While the persistent strategy maximizes channel use, it also maximizes the number of transmissions, making it energy inefficient. In these protocols, the number of transmissions per delivered packet is \(1/\eta\), where \(\eta\) is the throughput. Ideally, we would like the number of transmissions per delivered packet to equal 1. One way to approach the ideal is to anticipate transmission failures based on feedback and refrain from transmitting data until the channel has recovered. Thus, some of the useless transmissions performed during an OFF period can be avoided.

Therefore, when energy consumption and throughput are considered jointly, new criteria emerge in designing error control protocols. As an example, in Sections 4 and 5, we propose and analyze ARQ protocols in which, when the channel conditions deteriorate, the transmitter enters a probing mode, and sends a short probing packet every so often. When channel conditions improve, the transmitter switches back to the normal mode and restarts transmission from the point it was interrupted. The probe packet can be thought of as a packet with a header but either no or little payload. The potential gain of this protocol depends on how rapidly fading occurs relative to the round trip delay time on the link. We find that accepting a moderate throughput reduction makes it possible to significantly reduce the energy consumption of the system, relative to classic...
ARQ protocols. Besides reducing energy consumption, minimizing the number of transmissions per correctly received packet may be desirable in multiple access environments as a form of interference reduction.

3 MARKOV MODEL FOR THE FADING WIRELESS CHANNEL

The wireless channel is prone to correlated bursts of errors. A natural way of dealing with channels with memory is by using Markov models. Application of such models to the fading channel has been recently studied in [1], where it was shown that the fading envelope process is well approximated by a first-order Markov process. The applicability of a similar model to describe the binary process of packet success/failure has been discussed in [2], [3], where it was shown that, for most situations typically encountered in personal communications and mobile computing, a first-order binary Markov process adequately characterizes the error process at the packet level for transmission on a fading channel with additive Gaussian noise. More specifically, in [2], we studied the effect of correlated fading, by assuming that blocks are successfully received if and only if the value of the fading envelope is above a certain threshold. Using mutual information, and following an approach similar to that used in [1], it was shown that the block error process is well approximated by a binary Markov process. (Similar results have been obtained for packet transmission on the Gilbert-Elliott channel, another popular model for burst-error communications [4].)

Based on these results, the patterns of packet errors on the forward and reverse channels will be assumed to follow two independent first-order Markov models. The two transition matrices are $M_F(x) = M_F(1)^x$ and $M_B(x) = M_B(1)^x$, with

$$M_F(x) = \begin{pmatrix} p(x) & q(x) \\ r(x) & s(x) \end{pmatrix}$$

$$M_B(x) = \begin{pmatrix} a(x) & b(x) \\ c(x) & d(x) \end{pmatrix}.$$  

The parameters of the model at the packet level that result are as follows [2], [9]

$$\varepsilon = 1 - e^{-1/F} \quad (3)$$

$$r = \frac{Q(\theta, \rho \theta) - Q(\rho \theta, \theta)}{e^{\rho F} - 1}, \quad (4)$$

$$p = 1 - \frac{r e}{1 - \varepsilon} \quad (5)$$

where

$$\theta = \frac{2}{\sqrt{F(1 - \rho^2)}}, \quad (6)$$

and $\varepsilon$ is the steady-state packet error rate, $\rho = f_0(2\pi f_D t)$ is the Gaussian correlation coefficient of two samples of the complex amplitude of a fading channel with Doppler frequency $f_D$, taken at time distance $t$ ($f_0(\cdot)$ is the Bessel function of the first kind and zeroth order) [10], and $Q(\cdot, \cdot)$ is the Marcum Q function.

The simple threshold model of [2] ignored the type of modulation used and implicitly assumed that the value of the fading envelope is essentially constant throughout the duration of a block. In [3], we investigated the behavior of block errors, using a more detailed approach that took into account coding and modulation (coherent BPSK and noncoherent M-ary FSK) and tracked the fading process symbol by symbol. We found that a Markov approximation for the block error process (possibly degenerating into an iid process for sufficiently fast fading) remains a very good model for a broad range of parameters, and that a model which takes into account all the details of the modulation/coding scheme can be mapped into a threshold model by an appropriate choice of the parameters. Therefore, to mask the physical layer details, throughout this study we will consider the threshold model of [2], which is completely characterized by the values of the parameters $F$ and $f_0$ through (3), (4), (5), (6).

We assume, in the following, that the packet length is a constant, equal to one time unit (slot). This assumption is consistent with the current trend towards ATM-compatible transport modes, where fixed size packets are envisioned. The round-trip delay (accounting for propagation and processing times) is $m - 1$ slots from the end of a transmission to the reception and decoding of the corresponding feedback information. If a packet transmitted in slot $i$ is negatively acknowledged, it will be retransmitted in slot $i + m$. We assume that positive (ACK) and negative (NAK) acknowledgments can never be confused with each other, i.e., the effect of backward errors is to map the ACK and NAK symbols to an Erasure symbol. Also, we assume that each ACK/NAK carries the necessary information to identify all past correctly received packets.

4 PROBING PROTOCOL

We now propose a protocol that adapts to the channel by probing. The usefulness of the probing aspect of the protocol stems entirely from the correlations in the errors on the channel. As such, this protocol represents a refinement that is specifically tailored for the wireless channel.
4.1 Description

The probing ARQ protocol operates as follows. The transmitter sends one block per slot, as long as it senses that both channels are ON—this can be easily tracked by checking that ACKs are properly received. As soon as the transmitter detects that either channel is in error, it enters a probing mode, and sends a probing packet every $t$ slots. The probing is continued until an ACK is received, acknowledging the good status of both channels. At this time, the transmitter switches back to the normal mode and restarts transmission from the point where it had been interrupted. The transmission of a probe packet, which can be thought of as a packet with a header but either no or little payload, is assumed to consume a smaller amount of energy.

In summary, the receiver responds with an ACK every time it detects a valid transmission, and with a NAK every time it fails to do so. The transmitter acts according to the following rules:

- In the normal mode (N), it sends packets and receives the corresponding feedback after $m$ slots from the beginning of the transmission (round-trip).
- It switches from N to the probing mode (P) upon receiving a NAK or a garbled feedback.
- While in P, it transmits a probing packet every $t$ slots.
- It switches from P to N as soon as it gets an ACK to a probing packet.

In order to focus on the trade-off between the throughput efficiency and the energy consumption performance of the protocol, we assume that there are always packets to be transmitted. No queues are considered. We also recognize that ACKs consume energy and hence strategies to reduce their frequency may be desirable. However, this issue is not addressed in this paper.

4.2 Evolution

The evolution of the protocol depends on the status of the channel. This is adequately described by four states, namely,

1) state 1: erroneous transmission, correct feedback (E/C),
2) state 2: correct transmission, correct feedback (C/C),
3) state 3: erroneous transmission, erroneous feedback (E/E), and
4) state 4: correct transmission, erroneous feedback (C/E),

where the notation in brackets stands for the outcomes of the forward and backward channels, respectively, with E denoting erroneous and C denoting correct transmissions. Each transition occurs right after the packet transmission.

In the regular mode of operation (N), an ACK is received in every slot, and the system keeps making transitions from state 2 back to state 2. These transitions occur at a rate of one per slot. Define the time of stay (TOS) in a state as the number of slots the system stays inactive in that state. The TOS in state 2 is always one slot, and the probabilities of the transitions exiting state 2 are found from the one-step matrices $M_2(1)$ and $M_3(1)$. For example, the probability of going from state 2 to itself is $ap$ (i.e., $P[E|C]$ on the forward channel and $P[C|E]$ on the feedback channel), and so on.

On the other hand, upon leaving state 2, the system enters the probing mode, where a probe transmission and the corresponding feedback occur every $t$ slots. Therefore, the TOS in states 1, 3, and 4 is $t$ slots, since a transition from those states occurs right after the transmission of the next probe packet. The transition probabilities can be derived as before, except for the fact that now the $t$-step matrices, $M_2(t)$ and $M_3(t)$, are to be considered.

Note that in our analysis, the pair of outcomes on the two channels are tracked together. In the actual system, a transition from states 1, 3, or 4 to state 2 certainly takes place when C/C occurs, but only $m$ slots later will the transmitter be aware that a transition to mode N has occurred, when it actually receives the ACK. After that, it starts transmitting one block per slot. Therefore, the behavior in state 2 depends on where the transition came from. In order for the model to be Markovian, state 2 must therefore be split into two states, which we call 2 and 2', where state 2' is entered when a C/C occurs, but only $t$ slots after the transmitter be aware that a transition to mode N has occurred that the system was in states 1, 3, or 4. Due to the time it takes to the transmitter to detect the resumption of the N mode, the TOS in 2' is therefore $m$ slots. Also, state 2' is entered from states 1, 3, and 4, whereas state 2 can be entered only from 2' or itself.

Based on the above discussion, the transition matrix can be found, and is given by

$$P = \begin{bmatrix}
  a(t) & 0 & a(t) & b(t) & b(t) \\
  ap & 0 & b(t) & b(t) & b(t) \\
  c(t) & 0 & c(t) & d(t) & d(t) \\
  c(t) & 0 & c(t) & d(t) & d(t) \\
  c(t) & 0 & c(t) & d(t) & d(t)
\end{bmatrix},$$

with states ordered as 1, 2, 2', 3, 4; the TOS for each state is summarized in the vector

$$\begin{pmatrix} t & 1 & m & t & t \end{pmatrix},$$

and the transition diagram of the embedded Markov chain is depicted in Fig. 1.

![Fig. 1. Transition diagram for the probing ARQ protocol.](image-url)
4.3 Throughput Performance

Let $i$ be a state, arbitrarily chosen as a reference, and let $\pi_i$ denote its steady-state probability. The vector $\pi$ is the steady-state distribution of the Markov chain, i.e., the solution of the equation

$$\pi = \pi P, \quad \sum_{i \in \Omega} \pi_i = 1,$$

(9)

where $P$ is the transition matrix, whose elements $P_{ij} = P[j | i]$ are the transition probabilities, and $\Omega = \{1, 2, 2', 3, 4\}$. Also, let $D_j$ be the time delay associated to transition from state $i$ to state $j$ and define the mean waiting time in state $i$ as

$$D_i = \sum_{j \in \Omega} P_{ij} D_j.$$

(10)

In steady state, the mean recurrence time of state $i$, $\mathbf{D}_i$, defined as the average time between two consecutive entrances into state $i$, is given by (see [11, p. 641]):

$$\mathbf{D}_i = \frac{\sum_{i \in \Omega} \pi_i D_i}{\pi_i}.$$

(11)

Equation (11) holds for a generic reward function as well. If $R_{ij}$ is the reward (e.g., the number of successful transmissions, i.e., of packets which have been correctly received and acknowledged) associated to the transition from state $i$ to state $j$, we have, as in (10) and (11),

$$R_i = \sum_{j \in \Omega} P_{ij} R_{ij}, \quad \overline{R}_i = \frac{\sum_{i \in \Omega} \pi_i R_i}{\pi_i}.$$

(12)

From a fundamental theorem of renewal reward processes, we have that [12]

$$\lim_{\xi \rightarrow \infty} \frac{R(\xi)}{\xi} = \overline{R}_i = \frac{\sum_{i \in \Omega} \pi_i R_i}{\sum_{i \in \Omega} \pi_i D_i},$$

(13)

where $R(\xi)$ is the number of rewards (i.e., of correct receptions) at time $\xi$, and $\overline{R}_i$ is the average number of rewards during a cycle. A cycle is defined as the time between two consecutive passages for the reference state, $i$ (renewals). Note that in (13) the probability $\pi_i$ cancels out, and the result is independent of the choice of the reference state, as expected [11]. Note also that, if the rewards are the successful transmissions, (13) gives the steady-state throughput efficiency.

In the present context, the transitions from 2, 2' to 2 involve a single reward, whereas all others have none. The reward matrix is given by

$$R = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix},$$

(14)

and from (12) we obtain

$$R_2 = a p, \quad R_{2'} = a(m)p(m), \quad R_1 = R_3 = R_4 = 0.$$

(15)

Finally, the delays $D_i$s are just the components of (8). Therefore, the throughput performance of the protocol can be easily computed from (13), after the steady state distribution of the associated Markov chain has been determined.

4.4 Energy Consumption Performance

The above approach can be generalized: If $X(\xi)$ and $Y(\xi)$ are two reward functions, from the renewal reward process theory, we have [12]:

$$\lim_{\xi \rightarrow \infty} \frac{X(\xi)}{\xi} = E[X] = \sum_{i \in \Omega} \pi_i X_i, \quad Y = R.$$

(16)

This equation is used to compute the average number of transmissions per successful transmission, with $X = T$ (total number of transmissions) and $Y = R$ (number of successful transmissions).

A transition from states 2 and 2' to 2 involves one block transmission, transitions from states 2, 2' to states in $P$ involve transmission of $m$ blocks, whereas all outgoing transitions from states 1, 3, and 4 involve a probing packet transmission. The matrix of the block transmissions and the probing packet transmissions associated with the transitions are therefore

$$T^{(b)} = \begin{pmatrix}
0 & 0 & 0 & 0 \\
m & 1 - m & m \\
m & 1 - m & m \\
0 & 0 & 0 & 0
\end{pmatrix},$$

$$T^{(p)} = \begin{pmatrix}
1 & 1 & 1 \\
0 & 0 & 0 \\
1 & 1 & 1 \\
1 & 1 & 1
\end{pmatrix}.$$

(17)

If the energy associated with a block transmission is normalized to unity and that of a probing packet is denoted by $w \leq 1$, the matrix of the transmitted energy (measured in equivalent block transmissions) can be computed as $T^e = T^{(b)} + wT^{(p)}$. The average number of equivalent transmissions per successful transmission is therefore given by

$$N = \frac{\sum_{i \in \Omega} \pi_i T_i}{\sum_{i \in \Omega} \pi_i R_i}.$$

(18)

5 Selective Probing Protocol

The scheme described above can be enhanced in a simple way, if some complexity and buffering can be undertaken. Due to the round trip delay of $m$ slots, the transmitter is aware of the channel failure (in either direction) only after the ACK is due, and therefore it will transmit $m - 1$ more packets. Clearly, in the above scheme (as happens in GBN) all these $m$ packets are discarded and retransmitted as soon as the normal mode is resumed, regardless of the fact that some of them might have been correctly received.

Instead, upon detecting a failure, the transmitter could retain the packets already transmitted but not yet acknowledged. When an ACK is eventually received, the packets which were correctly received exit the buffer, and only those that were lost are retransmitted. Such a scheme, which can be seen as the probing version of ARQ Selective-Repeat (SR), could yield a significant improvement especially for large $m$, when
1) error bursts are short and
2) the failure was on the reverse direction and all m packets were in fact correctly received.

This refinement requires additional buffering at the receiver (as for standard SR) and the ACKs to the probe packets must indicate which packets have been correctly received and which have not. Since the normal mode of transmission is resumed upon receiving a correct ACK to a probe packet, the acknowledgment of the pending packets is always guaranteed.

Our interest here is in a basic study of the throughput/energy trade-off, and other issues such as reassembly times, queuing delays, packet numbering and buffer requirements will not be addressed. Note, however, that various techniques to deal successfully with these issues have been proposed in the past, and can be adapted to fit the present situation as well.

With respect to the previous analysis, we must add some rewards corresponding to states 1, 3, and 4, which correspond to the fact that after state 2 is left some pending packets may be correctly received and contribute to the total reward. In particular, it might be expected that the reward associated with state 4 will be the significant one (forward channel is good).

The throughput efficiency is given, in this case, by

\[ \eta_{GBN} = \frac{\pi_2}{\pi_2 + m(1 - \pi_2)} \]  

and the delays associated to the four states are \( m, 1, m, m \). The throughput efficiency is given, in this case, by

\[ \eta_{GBN} = \frac{\pi_2}{\pi_2 + m(1 - \pi_2)} \]  

\[
R = \begin{pmatrix}
0 & 0 & 0 & 0 \\
R_E & 1 - R_E & R_E & 0 \\
R_E & 1 - R_E & R_E & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
\]  

\( R_E \) is the reward corresponding to a sequence of \( m \) blocks whose first block was in error (transitions to 1 and 3), with \( 0 \leq R_E \leq m - 1 \). \( R_C \) is the reward corresponding to a sequence of \( m \) blocks whose first block was correct (transitions to 4), with \( 1 \leq R_C \leq m \). Also, \( R_E = E[R_E] \) and \( R_C = E[R_C] \), which are computed in Appendix A.

6 THROUGHPUT AND ENERGY CONSUMPTION PERFORMANCE

In this section, we present results on the energy consumption of the classic protocols and the proposed probing protocols. We present our numerical results for a slowly varying mobile radio fading channel. In particular, we assume slow Rayleigh fading with the normalized Doppler frequency \( f_D, \tau = 0.02 \). This corresponds to pedestrian speeds with a TDMA frame of \( \tau = 16 \) ms [13], or to the transmission of ATM cells (53 bytes) at 2 Mbps from or to a vehicle traveling at 120 km/h with a carrier frequency of 900 MHz. We also assume a symmetric structure for the binary Markov channels, i.e., \( \mathbf{M}_E = \mathbf{M}_B \) implying that the fading margins for data blocks, ACKs, and probe packets are the same. The effect of asymmetric link qualities has been discussed in [14] for the Selective-Repeat protocol, and similar considerations apply to this work.

Given its predictive role, the probe packets should have the same fading margin as the full packets. Being shorter in length, it is conceivable that the probe packets has a different success probability. In this case, one should choose different coding strategies for the two different packets, making the fading margins effectively equal. If this can not be done, say because no coding is used in the regular packets and so the probe packet transmissions are more robust, then the equivalence can be engineered by using a lower power for the transmission of the probe packets.

6.1 Classic Protocols

The performance of the classic ARQ protocols has been computed in the past. For later use in comparison, we recall here, briefly, the expressions for the throughput efficiency of GBN and SR.

For GBN, an approach similar to [15] can be used, where the protocol can be described by a four-state Markov chain, whose transition matrix is given by

\[ P = \begin{pmatrix}
(a(m)s(m)) & (a(m)r(m)) & (b(m)s(m)) & (b(m)r(m)) \\
aq & ap & bq & bp \\
(c(m)s(m)) & (c(m)r(m)) & (d(m)s(m)) & (d(m)r(m)) \\
(c(m)p(m)) & (c(m)p(m)) & (d(m)q(m)) & (d(m)p(m))
\end{pmatrix} \]  

and the delays associated to the four states are \( m, 1, m, m \). The throughput efficiency is given, in this case, by

\[ \eta_{GBN} = \frac{\pi_2}{\pi_1 + \pi_2 + m\pi_3 + m\pi_4} = \frac{\pi_2}{\pi_2 + m(1 - \pi_2)}. \]
For SR, an analysis similar to [16] leads to the following throughput expression:
\[
\eta_{SR} = \frac{r}{q} + \frac{c}{b + c}.
\] (22)

Finally, the number of block transmissions per slot is 1 in both protocols, so that the average number of transmissions per successful transmission is simply given by \(1/\eta\).

6.2 Numerical Results

We expect our probing protocols to have a smaller throughput than the classic protocols, but consume less energy.

As an example, Fig. 2 shows the throughput, \(\eta\), vs. the fading margin, \(F\), for some values of the probing interval, \(t\), and of the round trip delay, \(m\). It can be seen that, as expected, a larger \(t\) yields a smaller throughput, since the protocol samples the channel less frequently, and therefore on the average it takes more time for the system to become aware that the channel is ON. Also, the throughput is degraded if \(m\) is increased, as happens also in classic Go-Back-N, since a failure involves the retransmission of all pending packets regardless of their being correctly received.

The energy consumption characteristic of the protocols is depicted in Fig. 3, which shows the average number of transmissions per correctly received packet, \(N\), vs. the fading margin, \(F\). \(N\) is given by \(1/\eta\) for classic Go-Back-N whereas, for the probing protocol, two different quantities are to be considered: the average number of block transmissions, \(N_b\), and the average number of probe transmissions, \(N_p\). Fig. 3 shows how in harsh fading conditions (or in the presence of severe interference) the probing protocol greatly reduces the number of block transmissions, \(N_b\). The number of probe packet transmissions, \(N_p\), decreases as \(t\) increases, since probing the channel less frequently leads to a reduced energy consumption. On the other hand, \(N_b\) is determined only by the transitions from states 2 and 2’ (this can be seen directly from (14), (17), and (18)), and is therefore independent of \(t\).

The correct parameter to measure the energy consumption in the probing protocol is a weighed combination of the two, i.e., \(N = N_b + wN_p\), which is the average number of equivalent block transmissions per success. Here, the energy ratio between probe packets and data blocks is taken to be

![Fig. 2](image-url)  
![Fig. 3](image-url)
$w = 0.2$, which is reasonable in view of the fact that the probe packets are just the block overhead (note that for ATM cells $w$ would be about 0.1, leading to even better performance).

An interesting graph that highlights the trade-off discussed above is shown in Fig. 4, where the average number of equivalent block transmissions, $N$, is plotted vs. the throughput $\eta$. The curves for probing ARQ are parametrized by the probing interval, $t$, where increasing $t$ corresponds to smaller values of both $\eta$ and $N$, i.e., to moving left along the curves. On the other hand, the performance of classic ARQ GBN is represented by a single point. It is clear that the gain in transmission energy saving can be very significant, especially for $F$ not very high (less than 10 dB). Of course, for larger fading margins, the classic ARQ will perform just as well, since most of the packets need not be retransmitted, and $1/\eta$ is already close to 1. The probing protocol, on the other hand, can be used in environments where the impairments caused by propagation and interference are a dominant factor.

The effect of the mobile speed is investigated in Fig. 5, where similar curves are compared for $F = 10$ dB and for $f_d \tau = 0.002, 0.02$, and 0.2. It can be clearly seen that, as expected, a slowly varying channel yields better performance, whereas a fast varying channel leads to poor throughput performance. Also, when $t$ is increased in the slow fading channel case, the throughput loss is very small, so that the energy efficiency can be improved at almost no throughput cost. On the other hand, for a fast fading channel, increasing $t$ is not useful from the energy point of view, since the channel is almost independent and the probing is not effective, whereas the throughput performance is significantly decreased because of the large number of good slots which are left unused (e.g., $(1 - e)\tau$ per probing period on average, for iid packet errors).

Note that the curves in Fig. 5 can also be used to study the dependence of the performance on the carrier frequency (which directly affects $f_d$) and on the time between consecutive packets, $\tau$, which depends on the packet size and on the framing structure used (if any). In addition, the packet length may also affect the fading margin, as studied in [3].

Finally, the results obtained for the selective probing protocol described in Section 5 can be compared with ARQ Selective-Repeat. The results, not shown here, exhibit a behavior similar to those of Fig. 4, and reveal that some further improvement can be achieved, with respect to the previous case, even though, at least for the channel under study, the improvement does not justify the additional complexity.

7 Modeling Energy-Aware Evolutions

The technique described so far is adequate for finding the steady-state averages of the quantities of interest, but does not track the detailed information needed to model the dynamic behavior of batteries. In particular, the transition probabilities studied so far are independent of the status of
the battery. Such transitions correspond to a battery whose output power is a constant as long as there is some charge in it, and drops to zero when the battery is discharged. However, the step function may not always be an adequate description of the relationship between output power and battery charge.

In order to capture these features, we expand the model in the following way. Consider the transition diagram of the Markov model described above, and add another dimension, \( u \), which represents the units of energy that the battery has available. The resulting transition diagram is composed of a number of stages, 0 through \( U \), where \( U \) represents the number of energy unit stored in the battery when it is completely charged. Stage 0 is composed of one state, which is an absorbing state (the battery is completely discharged). We assume for simplicity that \( w = 1/K \), with \( K \) an integer, i.e., the energy required for a block transmission is an integer multiple of that for a probe transmission. In this case, the battery charge can be measured in terms of probe packet transmissions in our model. This can be generalized to the case in which \( w \) is any rational number, but a larger \( U \), although still finite, is needed. Let

\[
\mu = \frac{1}{w} T = K T^{(b)} + T^{(p)}
\]

be the matrix of consumed energy per transition, measured in probe packet units.

The transition structure of the Markov model is still maintained, except for the fact that the appropriate stage for the destination must be identified. In particular, a transition from \( i \) to \( j \) (which occurs with probability \( P_{ij} \)) in the Markov chain gives rise to transitions from \( i(u) \) to \( i(u - \mu_j) \) in the flow graph (\( \mu_j \) energy units are consumed). Also, some transitions for the stages close to 0 have to be modified, since there are no stages for \( u < 0 \) (actually, one can think of all “negative” stages as collapsed into state 0). This allows us to write recursive equations for the quantities of interest. For example, let \( \xi(u, \rho) \) be the probability that \( \rho \) rewards will be earned in the remaining lifetime, given that the system evolves from state \( i(u) \) to absorption. Then, conditioning on the destination of the first transition from \( i(u) \), a general recursive relationship can be written as follows:

\[
\xi_i(u, \rho) = \sum_{j \in \Omega} P_{ij} \xi_j(u - \mu_j, \rho - R_{ij})
\]

where \( R_{ij} \) is the number of rewards earned in the \( ij \)th transition. Note that this relationship is valid only for the first protocol introduced in this paper, where the \( R_{ij} \) are deterministic. If \( R_{ij} \) is an r.v., additional terms must appear, along with the probability mass distribution of \( R_{ij} \).

Similar relationships are found for other quantities, by replacing \( R_{ij} \) with the appropriate values (\( T^{(b)}_{ij} \), \( T^{(p)}_{ij} \), or \( \mu_j \) for the various number of transmissions, and \( D_i \), for the number of time slots). From these relationships, one can find the probability distribution of the number of time slots, the number of transmissions, and the number of successfully received blocks given that the system started in stage \( i(U) \) (possibly chosen at random according to the steady-state distribution, \( \pi \), even though the effect of this choice will vanish as \( U \) is large enough). A possible use of this information is in determining the average quantities, and in particular the throughput performance and the average number of transmissions per successfully received packet, which will be equal to the quantities computed through the steady-state Markov analysis, for \( U \) sufficiently large.

Specifically, we have the following recursive relationship for the mean and the second moment of the number of rewards, \( \rho(u) \), starting from state \( i \) in stage \( u \) to absorption:

\[
\overline{\rho}_i(u) = \sum_{j \in \Omega} P_{ij} [\overline{\rho}_j(u - \mu_j) + R_{ij}]
\]

(25)

\[
\rho^2_i(u) = \sum_{j \in \Omega} P_{ij} [\rho^2_j(u - \mu_j) + 2\rho_j(u - \mu_j)R_{ij} + R_{ij}^2]
\]

(26)

From (25) and (26), it is possible to find mean and variance by averaging over \( i \). Note that if \( U \) is large enough, and the stage-dependent parameters vary slowly with \( u \), the steady-state distribution \( \pi \) can be used.

This approach is powerful, not only because it gives the complete statistical information, but also because it allows us to essentially have all quantities dependent on the energy state, \( u \). For example, it is possible to incorporate performance degradation due to battery discharge by choosing \( P_{ij} = P_{ij}(u) \). Although this approach is a little tedious, it preserves enough modularity in the transition structure to allow efficient computations once the transition branches have been specified. In particular, when the quantities involved are functions of \( u \), recursive equations such as (24) still hold.

7.1 Battery Recovery

As mentioned before, there is evidence [6] that the capacity of a cell depends on the manner in which it is discharged, and that parameters such as the frequency and the duty-cycle of a pulsed discharge regime have an impact on the battery life. The analysis presented here could incorporate such recovery features with some additional complexity.

If we assume that a battery recovers \( \alpha \) units per slot on the average then, recognizing that an energy unit can be used only when completely recovered, an energy upgrade occurs every \( 1/\alpha \) slots. This can be modeled by associating an energy upgrade with probability \( D_\alpha \) with transition \( ij \), where \( D_i \) is the number of slots associated with transitions originating from state \( i \). A generalization that associates different degrees of energy upgrades with silent slots, block transmissions and probe transmissions can also be accommodated by including arbitrary transition-dependent upgrade probabilities, \( \alpha_{ij} \), instead of \( D_i/\alpha \). The steady-state analysis can be extended by introducing an appropriate “recovery” matrix to be subtracted from \( T \) (or \( \mu \)).

In the flow graph model, the recursive relationships must be modified to account for charge recovery. For example, (24) can be rewritten as

\[
\xi_i(u, \rho) = \sum_{j \in \Omega} P_{ij} \left[ \alpha_{ij} \xi_j(u - \mu_j + 1, \rho - R_{ij}) + \left(1 - \alpha_{ij}\right) \xi_j(u - \mu_j, \rho - R_{ij}) \right]
\]

(27)

and all other relationships can be generalized similarly.
8 Conclusions

In this paper, we have addressed the problem of the energy efficiency of ARQ error control protocols over Markov channels. We have shown that classic ARQ strategies could lead to a considerable waste of energy, due to the high number of transmissions. An adaptive scheme, which slows down the transmission rate when the channel is impaired, saves energy without a significant loss in throughput. A modified scheme, which yields slightly better performance but requires some additional complexity, is also studied.

The proposed protocols can be successfully employed for reliable data transmissions on mobile radio links. Finally, an analytical approach that can track more complex models for energy sources, including dynamic charge recovery, is discussed. This detailed approach could be computationally burdensome but is capable of accommodating the performance of error control protocols under more elaborate models for battery performance.

Future work based on a comprehensive understanding of battery characteristics in the wireless environment remains to be undertaken. We anticipate the possibility of developing new protocols that optimize the energy utilization and the battery life.

Appendix A: Computation of $R_E$ and $R_C$

As mentioned before, after exiting state 2 or 2', m data blocks are transmitted before the transmitter becomes aware of the failure and starts transmitting probe packets. We number these blocks from 1 to m (with block number 1 the first block to be transmitted). Let $a_i = 1$ if the $i$th block was successful, and $a_i = 0$ otherwise, $i = 1, \ldots, m$. With this notation, the number of rewards associated with these $m$ pending blocks is given by

$$R = \sum_{i=1}^{m} a_i .$$

We have

$$R_E = R(a_1 = 0) = \sum_{i=2}^{m} a_i \text{ given } a_1 = 0,$$

and

$$R_C = R(a_1 = 1) = 1 + \sum_{i=2}^{m} a_i \text{ given } a_1 = 1.$$  \hfill (28)

Let $\mathbf{a} = (a_1, a_2, \ldots, a_m)$. The probability of a given configuration of the success/failures of the pending packets, given $a_1$, is given by

$$P[\mathbf{a}_E] = P[a_{m} | a_{m-1} P[a_{m-1} | a_{m-2}] \ldots P[a_2 | a_1] P[a_1 | 0].$$

Finally, the average rewards are found as

$$R_E = \sum_{\mathbf{a}_E} P[\mathbf{a}_E],$$

$$R_C = \sum_{\mathbf{a}_C} P[\mathbf{a}_C].$$

Note that, with $\mathbf{a}' = (a_3, a_4, \ldots, a_m),$

$$R_E = \sum_{\mathbf{a}_E} P[\mathbf{a}_E] 0 \sum_{i=3}^{m} P[a_i | a_{i-1}] \sum_{j=2}^{m} a_j,$$

$$R_C = \sum_{\mathbf{a}_C} P[\mathbf{a}_C] 1 \sum_{i=3}^{m} P[a_i | a_{i-1}] \left(1 + \sum_{j=2}^{m} a_j \right).$$

If we let

$$\chi(a_2) = \sum_{i=3}^{m} P[a_i | a_{i-1}] \sum_{j=2}^{m} a_j,$$

we find a compact representation of the result:

$$\frac{R}{R_E} = \frac{1}{\chi(0)} + \left(\frac{p}{r} \cdot \frac{q}{s}\right) \chi(1).$$

Appendix B: List of Notation

- $M_F(x) = \left(\begin{array}{cc} p(x) & q(x) \\ r(x) & s(x) \end{array}\right)$: transition matrix for the forward channel;
- $M_B(x) = \left(\begin{array}{cc} a(x) & b(x) \\ c(x) & d(x) \end{array}\right)$: transition matrix for the backward channel;
- $F$: fading margin;
- $\epsilon = \frac{a}{r+s}$: average block error rate;
- $\rho = J_0(2\pi f_D \tau)$: Gaussian correlation coefficient of two samples of the fading process;
- $f_D$: Doppler frequency;
- $\tau$: time displacement of two channel samples;
- $J_0(\cdot)$: Bessel function of the first kind and zeroth order;
- $Q(\cdot)$: Marcum Q function;
- $m$: round-trip delay (in slots);
- $t$: probing period (in slots);
- $P$: transition matrix of the chain;
- $P_{ij}$: transition probability from state $i$ to state $j$;
- $\Omega$: state space of the chain;
- $\pi$: steady-state distribution of the chain;
- $D_i$: delay (in slot) associated with transition $ij$;
- $D_i = \sum_{j=1}^{\Omega} P_{ij} D_j$: delay (in slot) associated with state $i$ (averaged over the destination);
- $\overline{D_i} = \sum_{j=1}^{\Omega} \pi_{ij} D_j$: mean recurrence time of state $i$;
- $R_i$: reward (in blocks) associated with transition $ij$;
- $R_i = \sum_{j=1}^{\Omega} P_{ij} \overline{R_j}$: reward (in blocks) associated with state $i$ (averaged over the destination);
\[
\tilde{R}_w = \sum_{i \in \pi_m} \tilde{r}_i - \sum_{i \in \pi_m} \tilde{r}_i : \text{mean reward earned between two consecutive visits of state } i;
\]
\[
R(\zeta) := \text{total reward earned in } [0, \zeta];
\]
\[
\eta_i : \text{average throughput in blocks per slot};
\]
\[
\mathbf{T} = [\mathbf{T}^{(b)} + w\mathbf{T}^{(p)}] : \text{matrix of total transmissions};
\]
\[
w : \text{ratio between energy consumed by a probing packet transmission and energy consumed by a block transmission};
\]
\[
N_p : \text{average number of block transmissions per successful transmission};
\]
\[
N_{ij}^{(p)} : \text{average number of probing packet transmissions per successful transmission};
\]
\[
\mathbf{T} = [\mathbf{T}^{(b)} + w\mathbf{T}^{(p)}] : \text{matrix of total transmissions};
\]
\[
\tilde{r}_i : \text{reward corresponding to a sequence of } m \text{ blocks whose first block was correct (for the selective probing protocol)};
\]
\[
\tilde{r}_i : \text{reward corresponding to a sequence of } m \text{ blocks whose first block was incorrect (for the selective probing protocol)};
\]
\[
\tilde{R}_E = E[R_E];
\]
\[
\tilde{R}_c = E[R_c];
\]
\[
K = 1/w : \text{ratio between energy consumed by a block transmission and energy consumed by a probing packet transmission};
\]
\[
U_i : \text{total battery charge};
\]
\[
\mu_{ij} : \text{energy consumption associated with transition } ij;
\]
\[
\mu = \frac{1}{2} \mathbf{T} : \text{energy consumption matrix};
\]
\[
\rho_i(u) : \text{number of rewards starting from state } i \text{ in stage } u;
\]
\[
\tilde{\rho}_i(u) : \text{expected value of } \rho_i(u);
\]
\[
\tilde{\rho}_i^2(u) : \text{expected value of } \rho_i^2(u);
\]
\[
\alpha : \text{average number of energy units recovered per slot};
\]
\[
\alpha_{ij} : \text{average number of energy units recovered in transition } ij;
\]
\[
\alpha_{ij} = (a_1, a_2, \ldots, a_m);
\]
\[
\alpha_{ij} = (0, a_2, \ldots, a_m);
\]
\[
\alpha_{ij} = (1, a_2, \ldots, a_m);
\]
\[
\alpha' : \sum_{i=1}^{m} \prod_{j=1}^{m} P[a_i | a_{i-1}] = \sum_{j=2}^{m} a_j .
\]

**ACKEOWLEDGMENT**

This paper was presented in part at the 34th Annual Allerton Conference on Communications, Control, and Computing, Allerton House, Monticello, Illinois, Oct. 2-4, 1996.

**REFERENCES**


**Michele Zorzi** (S’89, M’95) received the Laurea degree and the PhD in electrical engineering from the University of Padova, Italy, in 1990 and 1994, respectively. During the academic year 1992/93, he was on leave at the University of California, San Diego (UCSD), attending graduate courses and doing research on multiple access in mobile radio networks. In 1993, he joined the faculty of the Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy. He is currently with the Center for Wireless Communications at UCSD. His present research interests involve performance evaluation in mobile communications systems, and random access in mobile radio networks. Dr. Zorzi is a member of the IEEE and AEI.

**Ramesh R. Rao** (SM’80) received his Honours Bachelor’s degree in electrical and electronics engineering from the University of Madras in 1980. He did his graduate work at the University of Maryland, College Park, Maryland, receiving the MS degree in 1982 and the PhD degree in 1984. Since then, he has been a member of the faculty of the Department of Electrical and Computer Engineering at the University of California, San Diego. His research interests include architectures, protocols and performance analysis of computer and communication networks.