Throughput Performance of an Unslotted Direct-Sequence SSMA Packet Radio Network

JAMES S. STOREY, MEMBER, IEEE, AND FOUAD A. TOBAGI, FELLOW, IEEE

Abstract—A spread-spectrum multiple-access (SSMA) packet radio network model is presented. The topology is a fully connected network with identical message generation processes at all radios. Packet lengths are exponentially distributed, and packets are generated from a Poisson process, resulting in a Markovian model. This network model accounts for the availability of idle receivers in a finite population network. The model also allows for the performance of the radio channel to be specified in detail. The channel considered is a BPSK direct-sequence SSMA radio channel with hard decision Viterbi decoding. An analysis of the Viterbi decoder leads to a bound on the decoder performance which is valid for a system with a varying probability of error, as is the case for the network under consideration. The approximate analysis yields lower bounds on throughput and probability of successful packet transmission. Results are given which show the effects on throughput of the received energy-to-noise density ratio, the number of chips per symbol, and the number of radios, as well as the improvement due to error control coding.

I. INTRODUCTION

FOR many years, various forms of spread-spectrum signaling have been used in secure communications systems. Though primarily implemented in military systems because of their anti-jam and anti- intercept properties, spread-spectrum techniques also provide a means for multiple access of a single radio channel, referred to as spread-spectrum multiple-access (SSMA). By using different spreading sequences, a number of transmissions can simultaneously occupy the channel, and each will suffer only slight degradation in signal quality compared to the performance of a single transmission. Furthermore, in packet radio networks, a single bit error causes the entire received packet to be unsuccessful, so the use of error control coding often leads to improved performance. In this paper, we investigate the performance of a packet radio (PR) network which uses SSMA with error control coding.

In SSMA networks with a large spreading factor, a low density of radios, or powerful error control coding, the multi-user interference is small compared to the desired signal. Under these conditions, one could assume that both the noise and the interfering transmissions of other radios have no effect on the quality of the channel between any pair of radios. Some authors (e.g., [1]) have modeled PR networks under this assumption, referred to as the perfect capture assumption. These models primarily account for the states of the radios (busy or idle) at the time of a packet reception. For a given data rate, larger spreading factors require a proportionally larger bandwidth. In other words, for a fixed bandwidth and data rate, the multi-user interference will become significant as the number of simultaneous transmissions increases. Since a greater overall throughput may be possible with more transmissions, we are interested in the performance of networks in the range for which the multiuser interference cannot be ignored, so perfect capture is not a valid assumption.

The early throughput analyses of SSMA packet radio networks, such as the work by Raychaudhuri [2] and by Pursley [3], considered time slotted systems, in which the number of interfering transmissions is constant throughout the entire packet. An unslotted network with varying multiuser interference was analyzed by Musser and Daigle [4], who assumed a simple approximate function for the probability of successfully receiving a packet, namely, 1 if the number of simultaneous transmissions does not exceed some cutoff L, and 0 if the number does exceed L. Pursley undertook a more exact analysis of an unslotted network, in the context of frequency-hopped SSMA [5]. By deriving expressions for the probability distributions of the maximum and minimum number of simultaneous transmissions occurring during the reception of a packet, he found upper and lower bounds on the probability of success for a fixed length packet, which in turn gave bounds on the throughput. Pursley and Taipale [6] analyzed the performance of a Viterbi decoder and derived a bound on the probability of a packet being successful for a direct-sequence spread-spectrum system using convolutional coding. However, the derivation of this bound assumes a constant error rate throughout the entire packet. Following the approach used in [5], they then found bounds on network throughput.

With the exception of the bounding approach used in [5] and [6], most analyses have used a simple model of the packet error process, either by assuming perfect capture (e.g., [1]), by making a step function approximation to the probability of success (e.g., [4]), or by considering a slotted system with a constant number of interferers throughout the entire packet. The work that did include an accurate model of the packet error process did not consider a variety of other factors which influence the success or failure of a packet transmission, in particular, the effect of the destination radio being busy in a finite population network.

In papers by Brazio and Tobagi [7] and also by the authors [8], it was shown that a Markovian network model could be extended to account for the state of the radio channel. In this paper, we combine this Markovian network model with a model of a direct sequence binary phase shift keying (DS-BPSK) SSMA radio channel using convolutional coding with Viterbi decoding, accounting for the variation over time of the multi-user interference. Two main contributions of this work allow us to do so. First, the network model incorporates a probability of error which is a function of the number of interfering transmissions. Second, the decoder performance is evaluated for the case when the error rate varies during the reception of a packet. This is in contrast to standard analysis of error control decoder performance, in which it is assumed that the raw channel errors are independent and occur with a constant probability from bit to bit.

With this approach, we are able to accurately, though not exactly, model a PR network at many levels. The performance
measures that are derived are close approximations to lower bounds on the throughput and the probability of a packet being successful. These results are found as a function of radio channel parameters, such as the received energy-to-noise density ratio $E_b/N_0$ and chips per symbol $N$, and of network parameters, such as the number of radios $M$ and offered traffic rate $\lambda$.

It has been shown (e.g., [19]) that the probability of error is a function of both the number of interferers and the distribution of the received signal powers, and cannot be accurately determined by considering only the total interfering power. Nevertheless, in this paper, we will consider only the case of equal-power interferers. Furthermore, in the current work, we focus on the effect of channel errors, and assume that preamble synchronization is instantaneous and perfect. In a multiple-access environment without time slotting, the combination of interference and noise will result in some packets being lost due to failure to acquire the spreading sequence. Nevertheless, as a first step, it is of interest to study the effect of channel errors on the network performance in isolation from the effects of synchronization. A detailed and realistic model of the effect of imperfect synchronization is beyond the scope of this paper, but has been examined in [10].

In the next section, the system under consideration is described, and the necessary assumptions and approximations are discussed. The analysis can be broken down into three parts, which are treated separately. Section III gives the Markovian model which constitutes the network model and which determines the evolution over time of the random process $X(t)$, the number of transmissions at time $t$. Section IV outlines the models used for the DS-BPSK radio channel. The Viterbi decoding process is analyzed in Section V. In Section VI, the expression for network throughput is derived, and numerical results are presented in Section VII.

II. SYSTEM MODEL

In this section, we present a description of the elements and organization of the packet radio network being modeled. A description of other aspects of PR networks can be found in [11]. For a tutorial on spread spectrum, the reader is referred to [12].

We use the following terminology. Bits refer to the unencoded information stream generated by the attached device. Symbols refer to the output of the convolutional encoder. Chips refer to the product of the data sequence and the spread-spectrum sequence. $N$ is the number of chips per symbol.

A. System Description

The network under consideration consists of $M$ packet radio units (PRU’s) sharing a single radio channel. Each PRU contains a transmitter and a receiver, and is connected to a terminal. A radio can be transmitting, receiving, or idle, but cannot transmit and receive at the same time. Traffic in the network consists of packets that are generated at one terminal, transmitted by the associated PRU, received by the destination PRU, and finally output to the destination terminal. The packets are of varying length, and the network is unslotted. We only consider the case of a fully connected network.

The channel access protocol is disciplined ALOHA [13]. With this protocol, each radio generates potential transmission times, known as the scheduling points, according to a random process, the scheduling process. These scheduling points dictate the times at which a radio might begin transmission; however, an actual transmission will only begin if the radio is not already transmitting or receiving a packet from another radio, and if the radio has a packet available for transmission. Thus, a radio will not interrupt a partially received packet in order to begin a packet transmission.

We consider spreading sequences that are random sequences of infinite length, independent from chip to chip. These sequences lead to an accurate model for a limited set of SSMA systems, in which there is little cross-correlation between spreading sequences used by different transmitters. This is the case for systems in which spreading is used primarily for security. Furthermore, it is a close approximation for a system using structured sequences, with the restriction that a different sequence is used for every transmitted symbol (i.e., bit-by-bit code changing), and that there are enough sequences that no sequence is ever re-used during the transmission of a packet.

The modulation format is direct-sequence binary phase-shift keying (DS-BPSK). We consider convolutional coding with Viterbi decoding.

B. Modeling Assumptions

In order to derive a model which leads to a tractable solution, we make a number of assumptions and approximations. These are described below, with a discussion of the motivation and validity of some of the modeling choices.

The synchronization process is perfect, so the receiver will always lock onto a packet destined to it, and will not be affected by packets destined to other radios. We ignore the overhead needed for Viterbi decoder synchronization and the CRC or parity bits used for error detection. We also ignore the effect of acknowledgements, assuming that a perfect instantaneous acknowledgment channel is available. Chip symbol, and carrier tracking is perfect, and error detection does not occur until the end of a packet. Thus, it the beginning of a packet is acquired, the receiver remains locked onto the packet until the end of transmission, regardless of whether or not bit errors occur.

To reduce the number of variables that must be considered, we assume that all radios are identical. Thus, all $M(M-1)$ source-destination pairs have equal traffic requirements, equal scheduling process rates and identical packet length distributions. Also, received signals from every radio have identical energy-to-noise density ratios. We assume zero propagation delay, so the network state as seen by every radio is identical.

In this paper, we derive the maximum achievable throughput. We do not treat delay, but assume the heavy traffic condition. For this condition, the output packet buffer has infinite capacity, and is full, so there is a packet available for transmission at every scheduling point. Packets to be transmitted are selected randomly from this buffer, so the lengths of successive packets transmitted by a PRU are independent. Also, packets are never lost at the generating PRU due to buffer overflow, so we do not need to distinguish between radios that are backlogged and those that are not. This heavy traffic assumption has often been used in previous analyses, and gives an upper limit to the achievable throughput.

The Markovian network model requires a Poisson process for the scheduling process, and requires that the transmitted packet lengths are drawn independently from an exponential distribution. The rate of the Poisson process is $\lambda$, referred to as the rate of offered traffic, and the mean packet length is $1/\mu$, giving a normalized rate of offered traffic $g = N/\mu$. It should be noted that the exponential distribution for packet length is the distribution of the lengths of transmitted packets, which includes both new packets and retransmissions. The distribution of lengths of successful packets, which are the packets supplied by the terminals, is skewed from this exponential. Actual networks may have distributions that are not exponential. A network with fixed length packets has been simulated, and the results given in Section VII indicate that the performance is not strongly dependent upon the packet length distribution.

III. NETWORK MODEL

The network model is a continuous time Markov chain. Because all radios are identical, we only need to keep track of
the number of radios in a given state rather than the exact state of each individual radio. Radios are inhibited from transmitting when they are receiving a packet, so the state of the network at time $t$, $Z(t)$, must include the number of PRU’s receiving packets $R$ as well as the number transmitting, $X$. Therefore, $\Omega$ the state space of $Z(t)$, is $\{(X, R) : 0 \leq X \leq M, 0 \leq R \leq X, X + R \leq M\}$. Brazio has shown that this Markov chain is irreducible and homogeneous [7], so $\Omega$ has a limiting state probability distribution $\{\pi_0(X, R), (X, R) \in \Omega\}$.

We use the following notation. $P_{\text{symbol}}(X, R)$ is the probability that a specific PRU is idle, conditioned on the network state $Z(t) = (X, R)$. $P_{\text{symbol}}(X, R)$ is the probability that a specific destination is idle given that the source is idle. The state-change rates for $Z(t)$ are determined as follows. For state $(X, R)$, $P_{\text{symbol}}(X, R) = (M - X - R)/M$, and new transmissions occur at the aggregate rate of $\lambda M P_{\text{symbol}}(X, R) = (M - X - R)\lambda$. The destination is idle with probability $P_{\text{symbol}}(X, R) = 1 - [(X + R)/(M - 1)]$. With perfect synchronization, if the source and destination are idle at a scheduling point the packet will be received, so there will be a transition to state $(X + 1, R + 1)$. This occurs at rate $\lambda M P_{\text{symbol}}(X, R)P_{\text{symbol}}(X, R)$. If the destination is busy, the packet transmission will correspond to a transition to state $(X + 1, R)$. These transitions occur at rate $\lambda M P_{\text{symbol}}(X, R)(1 - P_{\text{symbol}}(X, R))$. Of the $X$ packets being transmitted, $R$ are being received and $X - R$ are not. Thus, due to the completion of packet transmissions, there is a transition to state $(X - 1, R - 1)$ at rate $R\mu$, and to state $(X - 1, R)$ at rate $(X - R)\mu$.

We denote the rate transition matrix of the Markov chain by $Q$. We find the steady-state probabilities $\pi_0(X, R)$ of the Markov chain $Z(t)$ by solving the conservation equations,

$$\pi_0 Q = 0 \quad \text{and} \quad \sum_{(X,R) \in \Omega} \pi_0(X, R) = 1.$$  

(1)

IV. SPREAD-SPECTRUM RADIO CHANNEL MODEL

The model used for the direct-sequence BPSK radio channel is that of Pursley and Taipale, described in [6]. Because the network state changes slowly compared to the duration of a symbol, the interference $X - 1$ will be constant over this duration with high probability, and will change by no more than 1 with very high probability. Therefore, we can make the approximation that the interference is constant during a single symbol. The received $E_s/N_0$ is identical for all transmitter-receiver pairs.

In [6], it is proven that the worst case interference occurs when all interfering signals are phase and are bit aligned with the desired signal. Thus, the model assumes phase coherence of all received signals (i.e., a synchronous system), but gives a lower bound on the performance of an asynchronous system. This worst case leads to symbol errors that are conditionally independent given the number of interfering transmissions (this conditional independence of symbol errors is used in proving the bound on the performance of the Viterbi Decoder, in Section V). The channel model yields an upper bound on $P_{\text{symbol}}(X)$, the mean probability of symbol error for a symbol which is received in the presence of $X - 1$ interfering transmissions and Gaussian noise, as

$$P_{\text{symbol}}(X) \leq \sum_{i=0}^{N(X-1)} \binom{N(X-1)}{i} 2^{-N(X-1)} p(i)$$  

(2)

where

$$p(i) = \frac{\sqrt{2E_s/N_0}}{N} \left( 1 + \frac{2i - N(X-1)}{N} \right)$$  

(3)

and $Q(\xi)$ is

$$Q(\xi) = \frac{1}{\sqrt{2\pi}} \int_{-\xi}^{\infty} e^{-\gamma^2/2} d\gamma.$$  

(4)

An inequality is used because the expression found is a worst case bound on the probability of symbol error for an asynchronous system.

Using a Gaussian approximation for the binomial distribution of interfering chips, as in [12, vol. 2] gives

$$P_{\text{symbol}}(X) \approx Q(\sqrt{SNR(X)}),$$  

(5)

where

$$SNR(X) = \frac{N(X-1)}{N_0/2E_s}.$$  

The approximation of (5) was found to be accurate to within 1 percent of the results from (2) when $N(X - 1)$ is greater than 512 and $P_{\text{symbol}}(X) > 10^{-10}$. For very small $P_{\text{symbol}}(X)$, the relative accuracy becomes poor. However, error rates this low have virtually no impact on the network performance. For large values of $N(X - 1)$, the approximation requires significantly less computing time than the exact expression. Therefore, we calculated $P_{\text{symbol}}(X)$ using (2) when $N(X - 1) \leq 512$ and using (5) when $N(X - 1) > 512$.

V. VITERBI DECODER PERFORMANCE

In this section, we analyze the probability that a packet of length $L$ bits is correctly decoded in a system with convolutional coding, using hard decision Viterbi decoding. We derive a lower bound on the probability that the packet is successful $P_S$ which is a product over $j$ of a bound on $P_e(j)$ where $P_e(j)$ is the probability that no decoder error is made at a bit position $j$ given that no decoder error was made at all previous bit positions. This bound is of the same form as the bound obtained by Pursley and Taipale in [6], but this new derivation shows that the form is valid even when the symbol error rate varies during the reception of a packet (due to the varying number of interferers over time).

The resulting product form of the bound on $P_S$ allows the error process to be pessimistically modeled by a process that is independent from bit to bit, conditioned on the history of the number of interfering transmission (determined from the network states). In order to incorporate the error process modeling the decoder performance into the Markovian network, an approximation is made which is based on two observations. First, empirical studies of the viterbi decoder show that the performance of a decoder with a finite memory equal to $5K$ bits where $K$ is the constraint length of the code, will be virtually identical to the performance of the theoretical decoder with an infinite length memory. Second, for the operating ranges of interest, the network state governing the number of interferers changes slowly over this interval of $5K$ bits (see Section V-C below). Consequently, the state throughout the memory of the decoder can be approximated as being equal to the current network state. This second approximation is validated by simulation, and the resulting inaccuracy is shown to be less than 5 percent in the cases of interest. This approximation leads to a model of the error process which is independent from bit to bit and memoryless.

A. Viterbi Decoder Analysis

The following terminology is used. Bits refer to the uncoded data stream input from the attached device. Symbols refer to the binary sequence output from the encoder (for a rate 1/2 code, there are two symbols for every bit). We will use the word position to refer to the bit or symbol position in the data sequence, and generally reserve the word bit or symbol to mean the value (0 or 1) occurring at a given position. We denote the position by $j$. The analysis is given in terms of a rate
1/2 code. This is easily extended to the case of a rate 1/m code.

We consider the hard-decision decoding of a packet of length $\mathcal{L} + K - 1$ bits, which consists of $\mathcal{L}$ data bits and a known trailer of $K - 1$ bits. The encoder is preset to a known state at the start of each packet. We characterize the channel by a set of states $X_0$, and we let $\mathcal{C}$ be the vector of channel states $(X_0, X_1, X_2, \ldots, X_{\mathcal{L}+K-1})$. The channel is a memoryless binary symmetric channel, for which $P_e$ the probability of symbol error at position $j$, is a function of the current channel state $X_j$. In addition, the symbol errors are conditionally independent given $\mathcal{C}$. Throughout this analysis, all probabilities are distributions over $\mathcal{C}$. For notational simplicity, we will not explicitly write out this condition. Because the channel errors are symmetric, and convolutional codes are group codes, $P_c$ will be the same for any transmitted codeword [14], so we choose the data bits, trailer, and starting state to be all zeros.

$D_0$ is a sequence of $\mathcal{L} + K - 1$ bits where the first $\mathcal{L}$ bits are arbitrary and the last $K - 1$ bits are the known trailer bits, and $\alpha \in \{0, 2^2 - 1\}$. $D(j)$ is the subsequence consisting of the first $j$ bits of the sequence $D_0$, $C_{\alpha}$ is the sequence of $2(\mathcal{L} + K - 1)$ symbols output from the encoder (with a rate 1/2 code), for an encoder input of $D_0$. Thus, $\{C_{\alpha}\}, \alpha = 0, 2^2 - 1$ is the set of all possible codewords for a packet with $\mathcal{L}$ data bits. $D(j)$ is the subsequence consisting of the first $j$ symbols of the codeword $C_{\alpha}$. $C_{\alpha}(j)$ is the $j$th symbol of $C_{\alpha}$. The all zero data sequence is $D_0$, and the corresponding codeword, which is the transmitted codeword, is $C_0$. $C_0$ is the Hamming distance between $C_{\alpha}$ and $C_0$. $C_\alpha$ is the sequence of $2(\mathcal{L} + K - 1)$ received symbols, which is equal to the sequence of channel errors. $R$ is the subsequence consisting of the first $j$ received symbols, and $R(j)$ is the $j$th symbol of $R$. We denote an arbitrary encoder state by $\varepsilon$. $S(j)$ is the survivor path for state $\varepsilon$ at position $j$. $P_c$ is the probability that the packet in $\mathcal{C}$. For the sake of brevity, we will say that a codeword $C_{\alpha}$ is in state $\varepsilon$ at position $j$ to mean that $\varepsilon$ is the state of position $j$ of the encoder which produced $C_{\alpha}$.

The packet is successfully decoded if and only if $S(\mathcal{L} + K - 1)$, the survivor path for the zero state, is equal to the all zero path $D_0$, so

$$P_c = \prod_{j=1}^{\mathcal{L}+K-1} \Pr(S(j) = \mathcal{D}(j)|S(j-1) = \mathcal{D}(j-1)).$$

We define $P_c(j|1, 2, \ldots, j-1)$ to be the conditional probability

$$P_c(j|1, 2, \ldots, j-1) = \Pr(S(j) = \mathcal{D}(j)|S(j-1) = \mathcal{D}(j-1)).$$

To find $P_c(j|1, 2, \ldots, j-1)$, we only have to consider those codewords which return to the zero state at position $j$. Furthermore, the condition implies that any of these codewords which have previously returned to the zero state have already been eliminated, so the only codewords that need to be considered are those which return to the zero state for the first time at position $j$, and remain in the zero state from $j + 1$ to $\mathcal{L} + K - 1$. We denote these codewords by the set $\psi(j)$. $\psi$ denotes the union over $j = K, \mathcal{L} + K, \ldots$ of $\psi(j)$ (note that $\psi(j)$ is always empty for $j < K$). Any codeword $C_{\alpha} \in \psi(j)$ is identically zero for positions after $j$, so $|R(j) - \varepsilon(j)| = |R(j) - \varepsilon(j)| = |\varepsilon(j) - C_{\alpha}| - |\varepsilon(j) - C_{\alpha}|$, which we denote by $\delta_l(C_{\alpha})$.

In order to simplify the analysis in the case when there is a tie for the minimum distance path at some position $j$, we introduce $\mathcal{F}_j$, the event that $C_{\alpha}$ is favored over $C_0$, as follows. If $\delta_l(C_{\alpha}) < 0$ then with probability one, $C_{\alpha}$ is not favored, and if $\delta_l(C_{\alpha}) > 0$ then with probability one, $C_{\alpha}$ is favored. If $\delta_l(C_{\alpha}) = 0$, then $C_{\alpha}$ is favored with probability 1/2 where the choice is made independently of all other codewords. The complementary event $\mathcal{F}_j$ is the event that $\mathcal{F}_j$ does not occur.

At position $j$, if there is not a tie for the minimum distance codeword, then a decoder error will be made if and only if one or more codewords in $\psi(j)$ is favored over $C_0$. In the case of a tie between $C_0$ and $j$ codewords in $\psi(j)$, the probability that no codeword is favored over $C_0$ is $(1/2)^j$. However, if no decoder error has occurred up to position $j$, then the probability of a decoder errors $1/2$, since only two survivor paths are compared for each state ($i - 1$ of the tied code words will already have been eliminated). Thus,

$$P_c(j|1, 2, \ldots, j-1) \geq \Pr \left( \bigcap_{\alpha \in \psi(j)} \mathcal{F}_j \mid \text{no error at } 1, 2, \ldots, j-1 \right).$$

$$P_c \geq \Pr \left( \bigcap_{\alpha \in \psi(j)} \mathcal{F}_j \right).$$

In Appendix A, we prove that

$$\Pr(\mathcal{F}_j | \mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_{j-1}) \geq \Pr(\mathcal{F}_j)$$

for any codeword $C_{\alpha}$ and disjoint set of $\mathcal{C}$ codewords $C_0 \neq C_{\alpha}$, where the codewords are any of the $2^2$ codewords of length $\mathcal{L} + K - 1$. This proof is valid even when the probability of symbol error varies during the packet reception. Thus,

$$P_c \geq \prod_{j=K}^{\mathcal{L}+K-1} \Pr(\mathcal{F}_j).$$

This lower bound on the probability of successful packet reception has a product form, in which unconditional probabilities are multiplied to give the total probability. Because the bound takes this form, we can model the highly correlated output errors from the decoder by a simple process of errors that are independent from bit to bit, and the resulting solution will give a bound on the actual performance.

B. Numerical Evaluation of $P_c$

Unfortunately, the bound given in (10) is intractable even for very short packets, as the size of the sets $\psi(j)$ become very large even for numerically of $j$. Nevertheless, with a close approximation, we arrive at an expression that can be evaluated numerically.

The sets $\psi(j)$ are different for each position $j$. However, we can consider the supersets $\psi'(j)$ which include all codewords for which the first return to zero is at position $j$, even though codewords that were not in the zero state at positions $-K + 2$ to $0$. These supersets are equivalent for all $j$. By shifting the pattern of symbol errors occurring in positions $1$ to $j$ to the positions $2(\mathcal{L} + K - j) - 1$ to $2(\mathcal{L} + K - 1)$, we find a lower bound on the probability of correct decoder decision that is a function of the pattern of symbol errors, independent of the position $j$.

$$\prod_{\alpha \in \psi(j)} \Pr(\mathcal{F}_{\alpha}) \geq \prod_{\alpha \notin \psi(\mathcal{L} + K - 1)} \Pr(\mathcal{F}_{\alpha})$$

for the shifted sequence of symbol errors.

We now introduce an empirical result which allows us to reduce the number of codewords that must be considered. It has been found that with very high probability, the survivor paths merge within 4 or 5 constraint lengths in the past [15], [16]. In other words at any position $j$, all survivor paths $S(j)$ will be equal from positions $1$ to $j - i$ where $i$ is a random variable that depends upon the error pattern occurring in the
received sequence, but is less than 5 constraint lengths (35 bits for \( K = 7 \)) with very high probability. This implies that if a decoder error is made at position \( j \), then with very high probability, the chosen survivor path will be identically zero throughout positions 1 to \( j - 5K \). This allows us to limit \( \psi' = (\mathcal{L} + K - 1) \) to those codewords which are in the zero state at all positions except for positions \( \mathcal{L} + K - 5K \) to \( \mathcal{L} + K - 1 \). Thus, to a very close approximation

\[
\prod_{\sigma \in \psi} \Pr(\mathcal{F}_\sigma) = \prod_{\alpha \in \psi} \Pr(\mathcal{F}_\alpha) \tag{12}
\]

where \( \psi' \) are all those codewords in \( \psi' = (\mathcal{L} + K - 1) \) that are identically zero except for positions \( \mathcal{L} + K - 5K \) to \( \mathcal{L} + K - 1 \). We will use \( \bar{p}_C(p_{\bar{K}-k-1}, \ldots, p_{\bar{K}-1, k-1}) \) to denote the right-hand side of (12), which is a function of the shifted sequence of symbol errors \( (p_{\bar{K}-k-1}, \ldots, p_{\bar{K}-1, k-1}) \).

\( \hat{p}_C \) is a monotonic function of the probabilities of symbol error. Therefore, we can bracket the value of \( \hat{p}_C \) by considering the maximum and minimum probability of symbol error to occur during the interval from bit positions \( \mathcal{L} + K - 5K \) to \( \mathcal{L} + K - 1 \). This reduces the problem to one of evaluating \( \hat{p}_C \) with a constant probability of symbol error \( \rho \). For simplicity, we will denote this by \( \hat{p}_C(\rho) \).

We find \( \hat{p}_C(\rho) \) by considering the weight structure of the code being used. For a constant probability of symbol error \( \rho \), the probability that a codeword \( \mathcal{E}_i \) of weight \( i \) is favored is equal to \( \hat{p}_C(\rho) \), given by Viterbi in [17] as

\[
\hat{p}_C(\rho) = \begin{cases} 
\frac{1}{2} \left( \frac{i}{2} \right) \rho^{i/2} (1 - \rho)_{i/2}, & \text{if } i \text{ odd;} \\
\sum_{\sigma \in \psi} \left( \frac{i}{2} \right) \rho^{i/2} (1 - \rho)_{i/2}, & \text{if } i \text{ even.}
\end{cases}
\tag{13}
\]

If \( a_i \) is the number of codewords of weight \( i \) in \( \psi' \), and \( W_{\text{max}} \) is the maximum weight of any codeword in \( \psi' \), then

\[
\hat{p}_C(\rho) = \prod_{i=1}^{W_{\text{max}}} (1 - \rho^{i/2})^{a_i} \tag{14}
\]

As described in [17], we can find \( a_i \), the number of codewords of weight \( i \) in \( \psi' \), from a signal flow graph representation of the encoder. One of the authors [10] has determined the values of \( a_i \) up to \( i = 70 \) for the rate 1/2, \( K = 7 \) code of Odendalder [14]. For this code, there are 70 symbols during which the codewords in \( \psi' \) can be nonzero, so \( W_{\text{max}} \) cannot be greater than 70. Thus, numerical values for \( \hat{p}_C(\rho) \) can be calculated from these coefficients for this code.

Let \( \hat{p}_E \) denote the overall bound on the probability of success, given by the right-hand side of (10), and let \( \hat{p}_{\text{s}}(\rho) \) be the maximum probability of symbol error occurring during bit positions \( j - 5K + 1 \) to \( j \), and \( \hat{p}_{\text{s}}(\rho) \) be the minimum. Because the probabilities \( \hat{p}_{\text{s}}(\rho) \) are monotonically decreasing with increasing probability of symbol error

\[
\prod_{j=K}^{2K-1} \hat{p}_C(p_{\text{s}}) \geq \prod_{j=K}^{2K-1} \hat{p}_C(\rho_{\text{s}}) \tag{15}
\]

C. Memoryless Approximation

With the Markovian network model, the time between network state transitions is exponentially distributed. The rate is equal to the sum of the rates of the transitions leaving a given state, which is \( (M - X, R, X) \). From the numerical results, it is found that the maximum throughput is achieved with an offered traffic rate \( g \) less than 1.0, so for every state \( (X, R) \), the transition rate is bounded by \( M \). For a mean packet length \( 1/\mu = 1000 \) bits, this gives a mean holding time of at least 33 bits for \( M = 30 \). Furthermore, when the performance is limited by channel errors, which is the only case for which the approximations would have any effect, the throughput is maximized by a \( g < 1.0 \). For \( g \) in this range, the mean holding time is greater than \( 5K = 35 \) bits for all states with a large steady-state probability even for networks as large as \( M = 60 \). Thus, we can make the approximation

\[
\hat{p}_C(\rho_{\text{s}}(\rho)) \approx \prod_{j=K}^{2K-1} \hat{p}_C(p_{\text{s}}(\rho)) \tag{16}
\]

where \( p_{\text{s}}(\rho) \) is the current value of the probability of symbol error at bit position \( j \).

To the extent of the approximation that the survivor paths merge within 5 constraint lengths, a true lower bound on \( P_s \) is found by using \( P_{\text{s}}(\rho) \). We have written a discrete event simulation of the network, including memory over \( 5K \) bits, to determine the values of \( \prod_{j=K}^{2K-1} \hat{p}_C(p_{\text{s}}(\rho)) \) and \( \prod_{j=K}^{2K-1} \hat{p}_C(p_{\text{s}}(\rho)) \). The results from these simulations show that the approximate value \( \hat{p}_E \) is within 5 percent of these bounds over the ranges of interest, thus validating the approximation.

For a mean packet length of 1000 bits, we can ignore the effect of the small offset of \( K - 1 = 6 \) bits, and consider the product from \( j = 1 \) to \( j = \mathcal{L} \). Furthermore, because the holding times in the network states \( (X, R) \) are long compared to the duration of a single bit, when considering the network performance, this error process can be closely approximated by a continuous time process. The resulting process is Poisson, with arrivals corresponding to the occurrence of a decoder error. The rates \( \epsilon(X) \) of the auxiliary Markov chain are such that the probability of an arrival during a bit is equal to the probability of decoder error. Denoting the duration of one bit by \( b \) gives

\[
e^{-\epsilon b} = 1 - \hat{p}_E(\rho_{\text{symbol}}(\epsilon(X))) = \frac{\ln(1 - \hat{p}_E(\rho_{\text{symbol}}(\epsilon(X)))}{b} \tag{17}
\]

VI. THROUGHPUT ANALYSIS

We define the throughput for radio \( u \), \( S_u \), as

\[
S_u = \lim_{T \to \infty} \frac{1}{T} \text{(Time in interval } T \text{ spent by radio } u \text{ successfully transmitting data).} \tag{18}
\]

A common technique used for evaluating throughput is to find the probability of success for packet and multiply this by the average length of successful packets. For systems with fixed length packets, and also for perfect capture systems, the average length of successful packets is equal to the average length of transmitted packets. However, in the system being considered, the probability of success depends on the packet length, so the average length of successful packets is not equal to \( 1/\mu \). Therefore, the probability of success does not directly give the throughput. We instead find the throughput by introducing an auxiliary Markov chain in a method which was first found by Buzao in [18].

We consider \( T_u \), the contribution to throughput of a single
packet. If the packet is successful, then \( T_e \) is equal to the length of the packet, otherwise it is equal to 0. We denote by \( T_e(X, R) \) the expected contribution to throughput of a packet given that the network is in state \((X, R)\) just prior to the start of transmission, and the source and destination are idle. This expectation is over all future evolutions of the network, which takes into account all combinations of symbol error rate that might be encountered by a packet starting \((X, R)\). It should be emphasized that the evaluation of \( T_e(X, R) \) does not assume that the network state is \((X, R)\) throughout the entire packet. We denote the probability of success of a packet of length \( \tau \) given the same conditions by \( P_{S}(X,R,\tau) \) and the average of this over all packet lengths by \( \bar{P}_{S}(X,R) \). We can state the user throughput as the product of conditional probabilities,

\[
S_u = \sum_{(X,R) \in R} \lambda \pi_0(X,R) P_{S}(X,R) \bar{P}_{D}(X,R) T_e(X,R).
\]

(19)

Because all radios are identical, the network throughput is \( M \) times \( S_u \), which is

\[
S = \sum_{(X,R) \in R} \lambda \pi_0(X,R) (M - X - R) \cdot \left(1 - \frac{X + R}{M - 1}\right) T_e(X,R).
\]

(20)

A. Auxiliary Markov Chain

We find \( T_e(X, R) \) and \( P_{S}(X,R) \) by considering the states of the network during the transmission by \( u \) of a specific packet, referred to as the tagged packet. We model this with an auxiliary Markov chain, \( Z_{\text{aux}}(t) \). The state space of \( Z_{\text{aux}}(t) \) is \( \Omega_{\text{aux}} \), which consists of all the states that could be visited by the tagged packet, and two absorbing states, Success and Failure. We denote the nonabsorbing states of \( \Omega_{\text{aux}} \) and \( \Omega_{\text{aux}}^* \), which includes all those states of \( \Omega \) for which \( X \neq 0 \). For the uncoded channel, the state Failure is entered when a channel error occurs. For the channel with error control coding, the state Failure is entered when a decoder error occurs. If the packet is completed without entering the state Failure the transmission is successful, so the state Success is entered.

The approximate error process was shown in the previous section to be Poisson, with errors occurring at a rate \( \epsilon(X) \) which is a function of the number of transmissions \( X \). Because the process is Poisson, the time until the occurrence of an error is exponentially distributed. Thus, the occurrence of decoder errors, or channel errors in the case of no error control coding, can be modeled as transitions to the state Failure from each state \((X, R)\) at rate \( \epsilon(X) \). Also, the time to completion of a given packet is exponentially distributed with rate \( \mu \), independent of the network state. This is modeled as transitions from every state except the state Failure to the absorbing state Success at rate \( \mu \).

We denote the transition rate matrix of the auxiliary Markov chain by \( Q_{\text{aux}} \). The state-transition rates for \( Z_{\text{aux}}(t) \) are similar to those of \( Z(t) \), however, transitions from \((X, R)\) to \((X - 1, R - 1)\) are at a rate \( (R - 1) \mu \), and there are the additional transitions to the states Success and Failure. The nonabsorbing states \((X, R) \in \Omega_{\text{aux}}^*\) can be indexed with linear ordering, \( j = 1, 2, \cdots, L, \) the state Success indexed as state \( L + 1 \), and Failure as state \( L + 2 \). The matrix \( Q_{\text{aux}} \) can be expressed as

\[
Q_{\text{aux}} = \begin{pmatrix} Q_{\text{aux}}^* & \mu I^T & \phi^T \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}
\]

where \( \phi^T \) is the \( L \times 1 \) vector corresponding to transitions to the state Failure [Determined by \( \epsilon(X) \)], \( I^T \) is the \( L \times 1 \) vector of ones, \( \theta \) is the \( 1 \times L \) vector of zeros, and \( Q_{\text{aux}}^* \) is the \( L \times L \) submatrix corresponding to transitions between the states \( \{1, 2, \cdots, L\} \).

B. Network Throughput

\[
T_e(X, R) \quad \text{and} \quad P_{S}(X,R) \quad \text{are found from the transition rate matrix of the auxiliary Markov chain} \quad Q_{\text{aux}} \quad \text{as follows.}
\]

Integrating over all packet lengths gives

\[
T_e(X, R) = \int_0^\infty \tau P_{S}(X,R,\tau) f(\tau) \, d\tau
\]

(22)

\[
P_{S}(X,R) = \int_0^\infty P_{S}(X,R,\tau) f(\tau) \, d\tau
\]

where \( f(\tau) \) is the pdf of the packet lengths, \( f(\tau) = \mu e^{-\mu \tau} \). The product \( f(\tau)P_{S}(X,R,\tau) \) is equal to the rate at which \( Z_{\text{aux}}(t) \) enters the state Success at time \( \tau \) given that \( Z_{\text{aux}}(0^-) = (X + 1, R + 1) \). Conditioning on the state at time \( \tau^- \), we find

\[
f(\tau)P_{S}(X,R,\tau) = \sum_{(X', R') \in \Omega_{\text{aux}}} \mu_{R'}(Z_{\text{aux}}(\tau^-))
\]

(23)

\[
= (X', R') \cdot Z_{\text{aux}}(0^-) = (X + 1, R + 1)
\]

since there is a transition from every nonabsorbing state to the state Success at a rate \( \mu \), independent of \( \tau \).

We use \( [A]_{i,j} \) to denote the \( j \), \( i \) element of a matrix \( A \), and \( [v]_j \) to denote the \( j \)th element of a vector \( v \). For a homogeneous Markov chain with transition rate matrix \( Q \), the probability of being in state \( i \) at time \( \tau \) given starting state \( j \) at time 0 is \( e^{Q \tau} \). We let \( \phi_j \) correspond to the state \((X + 1, R + 1)\), so (23) becomes

\[
f(\tau)P_{S}(X,R,\tau) = \sum_{i=1}^L [e^{Q_{\text{aux}} \tau}]_{i,j}
\]

(24)

Only the upper left corner of \( e^{Q_{\text{aux}} \tau} \) is needed, which is equal to \( e^{Q_{\text{aux}}} \), so

\[
f(\tau)P_{S}(X,R,\tau) = \mu [e^{Q_{\text{aux}}} I^T]_j
\]

(25)

where \( I^T \) is the \( L \times 1 \) vector of ones as in Section III. This gives

\[
T_e(X, R) = \int_0^\infty \tau \mu [e^{Q_{\text{aux}}} I^T]_j \, d\tau
\]

(26)

\[
= \mu \left[ \int_0^\infty \tau e^{Q_{\text{aux}}} I^T \, d\tau \right]_j
\]

\[
= \mu \left[ (Q_{\text{aux}}^{-1})^T I^T \right]_j
\]

(27)

where \( j \) is the index corresponding to state \((X + 1, R + 1)\). In a similar fashion,

\[
P_{S}(X,R) = \int_0^\infty \mu [e^{Q_{\text{aux}}} I^T]_j \, d\tau
\]

(28)

\[
= \mu \left[ (Q_{\text{aux}}^{-1})^T I^T \right]_j.
\]

VII. RESULTS AND CONCLUSIONS

A. Results

Because we give results for both coded and uncoded channels, we will state the received energy-to-noise density ratio in terms of \( E_b/N_0 \) for both cases. All results are for a
mean packet length $1/\mu$ of 1000 bits, and results for the coded channel are for the rate $1/2$, constraint length 7 code given in [14].

In Fig. 1 we plot the throughput $S$ as a function of the offered traffic rate $g$ for the coded channel with $E_b/N_0 = 8.0$ dB, for a network size $M = 20$. The compression of the curves for $N$ greater than 128, indicates that the channel becomes essentially perfect for all levels of interference, so the performance is limited by the availability of receivers rather than by the channel errors. This is the range of operation for which the perfect capture assumption is valid.

Also shown are results from the simulation, which account for the maximum and minimum value of $P_{\text{symbol}}$ occurring over the previous 35 bits. As discussed in Section V, these results bracket the true lower bound. The resulting throughputs are denoted by $P^*$ and $P^\dagger$. The analytical results are within 5 percent of all of the simulation points. We have simulated a number of other points, and found that for the values of $g$ which give the maximum throughput, the maximum error is never greater than 5 percent. Thus, for the numerical results given in this section, the memoryless approximation gives an accuracy of less than 5 percent.

Fig. 2 shows the maximum throughput $S_{\text{max}}$ versus $N$ for several values of $E_b/N_0$, where $S_{\text{max}}$ is the throughput obtained by maximizing $S$ over $g$. Several effects are indicated by these curves. At low values of $N$, maximum throughput is limited by the combination of interference and thermal noise, i.e., the channel errors. At higher $N$, the effect of interference becomes small. Indeed, because the network size, and thus $X$, is finite, in the limit as $N \to \infty$, the symbol error rate for all values of $X$ will approach $P_{\text{symbol}}(1)$, the performance with no interference for the given $E_b/N_0$. We have calculated this asymptotic maximum throughput, and found it to be 2.10 for $E_b/N_0 = 5.0$ dB, and 3.41 for $E_b/N_0 = 6.0$ dB. For $E_b/N_0 \geq 8.0$, with large $N$, the maximum throughput is limited by the receiver availability rather than by the effect of channel errors, so the asymptotic maximum throughput is equal to the same value of 3.50 for all $E_b/N_0 \geq 8.0$ dB. Furthermore, this limiting value is reached at moderate values of $N$, as can be seen in the figure.

Fig. 2 also includes results from the simulation of a network with fixed length packets. Although there is some difference, especially for low $E_b/N_0$, the results for fixed length and for exponential length packets generally agree. This indicates that the results, and particularly, the sensitivities to the parameters under investigation, are not strongly dependent on the packet length distribution.

Fig. 3 shows the improvement due to error control coding, for the specific code considered. $S_{\text{max}}$ versus $E_b/N_0$ is plotted for the coded channel with 32 and 256 chips per symbol, and for the uncoded channel with 64 and 512 chips per bit. The different spreading factors were chosen to give the same number of chips per bit, denoted by $W$. At the higher spreading factors, the use of coding give a power savings of about 3.0 dB. At lower spreading factors, even with 3.0 dB higher power, the uncoded channel cannot support as high a throughput as the coded channel. For the limited case when the thermal noise density $N_0 = 0$, with $W = 64$, the maximum throughput is 2.23 for the uncoded channel and is 3.25 for the coded channel. The use of error control coding results in a significant improvement in performance, except when the performance of the uncoded system is limited by receiver availability.

We show the effect of receiver availability explicitly in Fig. 4 for the coded channel, with $E_b/N_0 = 8.0$ dB. At small values of $N$, the maximum throughput increases nearly linearly with $N$. For larger $N$, a limiting value is reached. In Fig. 5, we plot maximum network throughput $S_{\text{max}}$ and maximum user throughput $S_{\text{max}}/M$ versus $M$ for the coded channel, $E_b/N_0 = 8.0$ dB, for $N = 32$ and $N = 256$. For smaller $N$, the performance becomes limited by the channel at low values of $M$. The maximum network throughput flattens out above $M = 15$, so the maximum user throughput goes roughly as $1/M$. With a greater amount of spreading, the system is not limited by the channel errors except for $M = 60$. In this case, with perfect scheduling the maximum user throughput would be 0.5, since each radio would spend half the time transmitting and half the time receiving. However, for the random access system, we find that for large $M$ and large $N$, the maximum user throughput is equal to about
shown for any codeword \( C_0 \) and any set of \( M \) codewords \( \{C_{\omega_j}\} \) where the codewords are of length \( 2 (L + K - 1) \) symbols. All events are implicitly conditioned on the channel state for each symbol, \( X \). The following approach is used. We identify a set of random variables \( \{d_i\} \) which is a sufficient statistic for determining the probability of \( \Phi_\beta \). We condition on the \( d_i \) and find a recursive expression for the sum over all values of \( d_i \) for a given \( i \). Summing over all \( d_i \) for every \( i \), we find a bound on the total probability.

\[
\Pr \left( \Phi_\beta \mid \Phi_{\omega_1}, \ldots, \Phi_{\omega_M} \right) \geq \Pr \left( \Phi_\beta \right). \tag{A.1}
\]

In this section, the word position will refer to symbol position, not bit position. Consider the ordered collection of codewords \( C_0, C_{\omega_1}, \ldots, C_{\omega_M} \), each of length \( 2 (L + K - 1) \) symbols. At each position \( j \), we can find the pattern \( \Phi_j \) formed by the ordered sequence of symbols \( \{C_j(j), C_{\omega_1}(j), \ldots, C_{\omega_M}(j)\} \). Let the value \( \nu(\Phi_j) \) be the number whose binary representation is the pattern \( \Phi_j \) i.e.,

\[
\nu(\Phi_j) = \sum_{j=0}^{M-1} C_{\omega_j}(j) \times 2^j.
\]

\[
\nu(\Phi_j) = C_{\omega_j}(j) \times 2^0 + C_{\omega_{j-1}}(j) \times 2^1 + \cdots + C_0(j) \times 2^{M-1} + C_0(j) \times 2^M. \tag{A.2}
\]

Denote by \( \mathcal{Y}(i) \) the set of all positions \( j \) for which \( \nu(\Phi_j) = i \). Denote by \( \mathcal{N}(i) \) the cardinality of the set \( \mathcal{Y}(i) \). Furthermore, let \( \mathcal{Y}(i) \) be the subset of \( \mathcal{Y}(i) \) of the symbols of the received sequence that is equal to 1. Let \( n_i = \mathcal{N}(i) + 2^k \), \( d_i = \mathcal{N}(i) + 2^k \), \( N = 2^k - 1 \), and let \( e_i \) refer to a specific value of the random variable \( d_i \). Thus, \( d_i \) is the number of positions for which the received sequence is one, \( e_i \) is one, and all of the \( e_i \)'s are zero, and \( d_i \) is the number of positions for which all of these are one. Define \( \tilde{d}_i \) to be the vector \( (d_{i,0}, d_{i,1}, \ldots, d_{i,t}) \) and similarly for \( \tilde{e}_i \).

Given the values \( \mathcal{N}(i) \), an event \( \Phi_j \) is conditionally independent of both the received sequence \( \mathcal{X} \) and all other events \( \Phi_{\omega_k} \), where \( \gamma_1 \) and \( \gamma_2 \) can be \( \beta \) or one of the \( \omega \). The numbers \( \mathcal{N}(i) \) are determined by the set of codewords \( C_0 \) and \( \{C_{\omega_j}\} \). The numbers \( \mathcal{N}(i) \) are random variables, since they are functions of the random channel errors. Also, the \( \mathcal{N}(i) \) for different patterns \( i \) are independent, since they depend upon the symbol errors occurring during disjoint sets of positions, and the symbol errors are independent conditioned on \( \mathcal{X} \) (where \( \mathcal{X} \) is the vector of channel states). Furthermore, \( \mathcal{N}(i) \) and consequently \( \Pr (\Phi_j) \) does not depend upon \( R(j) \) at positions \( j \) for which \( C(j) = 0 \), so \( \Phi_j \) is independent of \( \mathcal{N}(i) \) for \( i < 2^k \). As a result,

\[
\Pr \left( \Phi_j \mid \Phi_{\omega_1}, \ldots, \Phi_{\omega_M} \right) = \Pr \left( \Phi_j \mid \tilde{d}_N = \tilde{e}_N \right). \tag{A.3}
\]

We will use the shorthand \( U \) for the event \( \Phi_j \) and \( V \) for the intersection of events \( \Phi_{\omega_1}, \Phi_{\omega_2}, \ldots, \Phi_{\omega_M} \). As long as the probability of symbol error is always less that 1, the probabilities \( \Pr (U) \) and \( \Pr (V) \) will be nonzero, as will the probability of \( \tilde{d}_N = \tilde{e}_N \) for all values of \( \tilde{e}_N \). However, for some values of \( \tilde{d}_N \), the joint probability \( \Pr (\tilde{d}_N = \tilde{e}_N, U, V) \) may equal 0. Therefore, we define \( m(e, \mathcal{N}) \) to be the largest value of \( e_i \) for which \( \Pr (\tilde{d}_N = \tilde{e}_N, U, V) > 0 \). We use the shorthand notation \( m_i \) to refer to \( m(e_i, \mathcal{N}) \). Thus, in the following, \( \Pr (U \mid \tilde{d}_N = \tilde{e}_N) = \Pr (U \mid \tilde{e}_N) \), \( \Pr (d_i = e_i) \), and \( \Pr (V) \) are nonzero for all \( e_i \leq m_i \). We can calculate

\[
\Pr \left( \Phi_j \mid \Phi_{\omega_1}, \ldots, \Phi_{\omega_M} \right) = \Pr (U \mid V) = \sum_{e_0 = 0}^{m_0} \sum_{e_1 = 0}^{m_1} \cdots \sum_{e_N = 0}^{m_N} \cdot \Pr (U \mid \tilde{d}_N = \tilde{e}_N) \Pr (\tilde{d}_N = \tilde{e}_N \mid V). \tag{A.4}
\]

**APPENDIX**

**PROOF OF BOUND ON CONDITIONAL PROBABILITY**

In this Appendix, we prove that the conditional probability of \( \Phi_j \) given the set of constraints \( \{\Phi_{\omega_1}, \Phi_{\omega_2}, \ldots, \Phi_{\omega_M}\} \) is greater than the unconditional probability of \( \Phi_j \). This is

0.175, or 35 percent of the ideal 0.5. Similar asymptotic results have been derived by Sousa and Silvester in [19].

**B. Conclusions**

In this paper, we have combined a detailed Markovian network model and an accurate model of a DS-BPSK SSMA radio channel with convolutional coding. To achieve this, it was necessary to show that the bound on decoder performance of Pursley and Taipale [6] is valid in the case of a varying probability of symbol error. Furthermore, several approximations were made to allow numerical results to be derived. This allowed us to find the throughput of an SSMA network with an imperfect radio channel. We have presented numerical results that demonstrate various characteristics of the performance of such networks.

In a multiple-access spread-spectrum radio network, to determine the optimum values for the parameters of the radio channel such as \( E_b/N_0 \) and chips per symbol \( N \), one must take into account the effect of network parameters such as the density of radios and the rates of offered traffic, which determine the amount of interference that is experienced during the reception of a packet. We hope that the model and results presented here have helped to provide insight into these effects.
\[
\text{Pr}(U|V) = \sum_{e_i} \text{Pr}(U|d_i = e_i) \Delta_i(\bar{e})
\]

\[
= \sum_{e_i} \sum_{e_N} \text{Pr}(U|d_N = e_N, d_{N-1} = e_{N-1}) \cdot \text{Pr}(d_N = e_N|\bar{e}_{N-1}) \\times \left( \text{Pr}(d_N = e_N, d_{N-1} = e_{N-1}|V) - \text{Pr}(d_N = e_N, d_{N-1} = e_{N-1}) \right).
\]

(A.5)

It is shown in [10] that \(\text{Pr}(V|d_s = e_{s-1})\) is monotonically decreasing with \(e_s\), so

\[
\text{Pr}(V|d_s = e_s, d_{s-1} = e_{s-1}) \geq \text{Pr}(V|d_s = e_s + 1, d_{s-1} = e_{s-1}). \quad (A.6)
\]

Dividing both sides by \(\text{Pr}(V)\) gives

\[
\frac{\text{Pr}(d_s = e_s, d_{s-1} = e_{s-1}|V)}{\text{Pr}(d_s = e_s + 1, d_{s-1} = e_{s-1}|V)} \geq \frac{\text{Pr}(d_s = e_s + 1, d_{s-1} = e_{s-1}|V)}{\text{Pr}(d_s = e_s, d_{s-1} = e_{s-1}|V)} \cdot \frac{\text{Pr}(d_s = e_s, d_{s-1} = e_{s-1}|V)}{\text{Pr}(d_s = e_s + 1, d_{s-1} = e_{s-1}|V)}.
\]

(A.7)

Since this ratio is monotonically decreasing with \(e_s\), there are three possibilities

\[
\frac{\text{Pr}(d_s = e_s, d_{s-1} = e_{s-1}|V)}{\text{Pr}(d_s = e_s + 1, d_{s-1} = e_{s-1}|V)} \begin{cases} = 1 & 0 \leq e_s \leq m_i; \\ < 1 & 0 \leq e_s < m_i; \\ \text{crosses 1} & 0 \leq e_s \leq m_i \end{cases}
\]

(A.8)

where by crosses 1 we mean that it is \(\geq 1\) for \(e_s = c_i\) and is \(< 1\) for \(e_s > c_i\). We define the term \(\Delta_i(\bar{e})\) to be

\[
\Delta_i(\bar{e}) = \text{Pr}(d_s = e_s|V) - \text{Pr}(d_s = e_s). \quad (A.9)
\]

Because the probabilities are nonnegative, we find inequalities for \(\Delta_i(\bar{e})\) that correspond to cases in (A.8)

\[
\Delta_i(\bar{e}) \begin{cases} \geq 1 & 0 \leq e_s \leq m_i; \\ < 1 & 0 \leq e_s \leq m_i; \\ \text{crosses 0} & 0 \leq e_s \leq m_i \end{cases}
\]

(A.10)

Thus, either there is an integer \(c_i\) in the range \(0 \leq c_i \leq m_i\) such that \(\Delta_i(\bar{e}) \geq 0\) for \(e_s \leq c_i\) and \(\Delta_i(\bar{e}) < 0\) for \(e_s > c_i\), or \(\Delta_i(\bar{e}) < 0\) for all \(e_s\).

In order to derive a recursion expression, we consider the general form of the inner summation of (A.5),

\[
\sum_{e_i} \text{Pr}(U|d_i = e_i) \left( \text{Pr}(d_i = e_i|V) - \text{Pr}(d_i) \right) \Delta_i(\bar{e})
\]

(A.11)

In the case \(\Delta_i(\bar{e}) \geq 0\) for some \(e_i \in (0, 1, \cdots, m_i)\), we split

\[
\sum_{e_i} \text{Pr}(U|d_i = e_i, d_{i-1} = e_{i-1}) \Delta_i(\bar{e})
\]

(A.12)

For \(e_i \leq c_i\), \(\Delta_i(\bar{e}) \geq 0\) by definition. Equation (A.6) also holds for \(U_j\), so for \(e_i \leq c_i\),

\[
\text{Pr}(U|d_i = e_i, d_{i-1} = e_{i-1}) \geq \text{Pr}(U|d_i = c_i, d_{i-1} = e_{i-1}). \quad (A.13)
\]

Similarly, for \(e_i > c_i\),

\[
\text{Pr}(U|d_i = e_i, d_{i-1} = e_{i-1}) \geq \text{Pr}(U|d_i = c_i, d_{i-1} = e_{i-1}) \quad \text{for} \ e_i > c_i \quad (A.14)
\]

and \(\Delta_i(\bar{e}) < 0\), so the inequality reverses to the desired direction. Thus, both parts of the sum give the same inequality, so

\[
\sum_{e_i=0}^{m_i} \text{Pr}(U|d_i = e_i, d_{i-1} = e_{i-1}) \Delta_i(\bar{e})
\]

(A.15)

In the case \(\Delta_i(\bar{e}) < 0\) for all \(e_i \in (0, 1, \cdots, m_i)\), (A.15) and (A.16) hold for all \(e_i\), if we let \(c_i = 0\). We can find this sum over \(e_i\) of \(\Delta_i\) as

\[
\sum_{e_i=0}^{m_i} \text{Pr}(U|d_i = e_i, d_{i-1} = e_{i-1}) \Delta_i(\bar{e}) = \sum_{e_i=0}^{m_i} \sum_{e_i=0}^{m_i} \text{Pr}(d_i = e_i, d_{i-1} = e_{i-1}|V) \Delta_i(\bar{e})
\]

(A.17)

Thus,

\[
\sum_{e_i=0}^{m_i} \text{Pr}(U|d_i = e_i, d_{i-1} = e_{i-1}) \Delta_i(\bar{e}) \geq \text{Pr}(U|d_i = c_i, d_{i-1} = e_{i-1}) \Delta_i(\bar{e})
\]

(A.18)
Now, (A.5) can be evaluated as

\[
\Pr(U | V) - \Pr(U) = \sum_{\vec{d}_{N-1} \epsilon \mathcal{E}_N} \Pr(U | d_N = \vec{c}_N, \vec{d}_{N-1} = \vec{e}_{N-1}) \Delta_N(\vec{e}_N)
\]

\[
\geq \sum_{\vec{e}_{N-1}} \Pr(U | d_N = \vec{c}_N, \vec{d}_{N-1} = \vec{e}_{N-1}) \Delta_N(\vec{e}_{N-1})
\]

\[
\geq \sum_{\vec{e}_{N-2}} \Pr(U | d_N = \vec{c}_N, \vec{d}_{N-1} = \vec{e}_{N-2}, \vec{d}_{N-2} = \vec{e}_{N-2}) \Delta_{N-2}(\vec{e}_{N-2})
\]

\[\vdots\]

\[
\geq \sum_{\vec{e}_0} \Pr(U | \vec{d}_N = \vec{c}_0) \Delta_0(\vec{e}_0)
\]

\[
\geq \Pr(U | \vec{d}_N = \vec{c}_0) \sum_{\vec{e}_0} (\Pr(d_0 = \vec{e}_0 | V) - \Pr(d_0 = \vec{e}_0)).
\]

(A.19)

However,

\[
\sum_{\vec{e}_0} \Pr(d_0 = \vec{e}_0 | V) = \sum_{\vec{e}_0} \Pr(d_0 = \vec{e}_0 | V) = 1
\]

so the difference in the last line of (A.19) is 0, giving \( \Pr(U | V) - \Pr(U) \geq 0 \), or

\[
\Pr(\vec{F}_\beta | \vec{F}_\alpha, \ldots, \vec{F}_{\alpha_k}) \geq \Pr(\vec{F}_\beta).
\]

(A.20)

REFERENCES


