Firing the Genie: Two-Phase Short-blocklength Convolutional Coding with Feedback

Adam R. Williamson, Tsung-Yi Chen and Richard D. Wesel
Department of Electrical Engineering, University of California, Los Angeles, Los Angeles, California 90095
Email: adamroyce@ucla.edu; tychen@ee.ucla.edu; wesel@ee.ucla.edu

***Abstract*—In an effort to account for the latency cost of error detection at short blocklengths, we simulate a two-phase feedback-based incremental redundancy scheme. This scheme consists of communication and confirmation phases, as used in the error exponent literature, and allows messages to be decoded with high reliability. Simulation results of tail-biting convolutional codes on the AWGN channel are shown, which demonstrate that the two-phase scheme can deliver throughput surpassing the random coding lower bound on variable-length feedback (VLF) code achievability. A comparison with simulation of CRC-based error detection is also presented.**

I. Introduction

Polyanskiy et al. [1] tightly characterized the backoff from capacity at short blocklengths without feedback by providing achievability and converse bounds on the maximum rate, along with a normal approximation of the channel dispersion (i.e., the stochastic variation of the channel) that closely approximates both bounds.

In [2] Polyanskiy et al. similarly characterized performance at short blocklengths when noiseless feedback is available. In contrast to the no-feedback case, when feedback is present there is a large gap between the lower and upper bounds on maximum rate at short blocklengths.

In [2], Polyanskiy et al. studied two distinct settings of variable length coding with feedback. One setting is the variable-length feedback (VLF) setting in which variable-length codes are employed at the transmitter potentially taking advantage of full noiseless feedback of the noisy received symbols. With VLF, the receiver decides when to make a final decision (and end the transmission of symbols for that message) based on the noisy received symbols.

The second setting is variable-length feedback with termination (VLFT). In this setting, variable-length codes again are employed at the transmitter as before, but the transmitter also has the ability to send one noiseless termination symbol for each message. The receiver makes a final decision using the noisy symbols it has received before receiving the noiseless termination symbol.

For VLFT in the context of full noiseless feedback of the received symbols, the transmitter sends the noiseless termination as soon as it sees that the receiver will decode to the correct message. From a simulation perspective, this is equivalent to having a genie at the receiver that informs the decoder when it has received a sufficient number of noisy symbols to decode correctly.

This research was supported by the National Science Foundation through grant CIF CCF 1162501.

In [3], we presented a VLFT scheme with limited decoding times (corresponding to "packets"), using variable-length codes with finite maximum length formed by the puncturing of tail-biting convolutional codes. We also provided a rate-compatible sphere-packing (RCSP) analysis of the scheme and showed that the RCSP predictions closely matched the performance of simulated tail-biting convolutional codes. When limited to the same decoding times, the VLFT random coding lower bound on throughput of [2] was exceeded by the throughput predicted by RCSP and that achieved with tail-biting convolutional codes.

The goal of the present work is to characterize how well punctured convolutional codes can perform when there is no noiseless termination symbol (i.e., no genie) to declare when the receiver will decode correctly. This is the VLF setting of [2], in which the receiver must decide to terminate transmission based on a desired probability of error ϵ. We adopt a two-phase approach following Burnashev [4] to increase the reliability of the decoder's decisions in the absence of a genie. Based on feedback from the receiver after the first phase, in the second phase the transmitter sends a confirmation message to confirm or reject the decoder's tentative decision.

In Sec. II we extend the incremental redundancy (IR) scheme of [3] to incorporate the second phase and derive the expected throughput and latency. Rate-compatible sphere-packing analysis is used to determine the transmission lengths used in each phase-1 incremental transmission and the length of the confirmation messages used in phase 2. We show two-phase simulation results for the AWGN channel in Sec. III, using maximum-likelihood (ML) decoding of tail-biting convolutional codes. We also compare with a one-phase scheme in which cyclic redundancy checks (CRCs) are used for error detection and discuss the shortcomings of a CRC-based approach. Sec. V concludes the paper

II. Feedback without Noiseless Termination

As mentioned above, the VLFT approach presented in [2] includes a special use-once, noise-free termination symbol that is communicated on a separate feedforward control channel. This facilitates zero-error communication at short blocklengths; the transmitter determines when the receiver has decoded correctly (based on noiseless feedback) and sends the special termination symbol. This termination symbol models

practical systems which have a separate and highly-reliable control channel that can effectively be considered noise-free. This allows the message-communication and termination aspects to be considered separately.

Since not all systems have access to separate control channels, Polyanskiy et al. [2] also considered the VLF setting, which does not include the noiseless termination symbol. Consider the following one-phase and two-phase approaches for feedback in this setting:

In the one phase approach, the receiver decides when it has decoded with sufficient reliability based on its noisy received symbols that encode the message. The receiver uses feedback only to inform the transmitter to stop transmitting symbols for the current message.

In the two-phase approach, the transmitter sends a first phase of symbols that encode the primary information message. The receiver attempts decoding and uses feedback to communicate to the transmitter either all the noisy received symbols or, alternatively, its decoded message. The transmitter compares the transmitted codeword with the result of decoding the received symbols and informs the receiver in a second phase of symbols that encode a confirmation message whether it has correctly decoded.

For both the one-phase and two-phase schemes described above, zero-error communication is no longer possible at short blocklengths.

In the one phase scheme, the receiver will sometimes (unknowingly) decode incorrectly but still decide that it has decoded with sufficient reliability and conclude communication related to that message. All error detection mechanisms, such as a CRC included within the information bits [5], a bounded-distance maximum-likelihood or bounded-angle maximum-likelihood decoder [6], [7], or an erasure decoding rule [8] have residual undetected errors.

In the two-phase scheme with full feedback, the transmitter does have full knowledge of when errors have occurred, but the second phase will sometimes be decoded incorrectly leading the receiver to incorrectly conclude communication related to that message, which leads to undetected errors at the receiver.

A. Two-Phase Scheme

Inpired by the error exponent literature for memoryless channels with feedback (e.g., see [4], [9] and the references therein), this paper focuses on the two-phase scheme. The canonical two-phase scheme assumes that causal, noiseless feedback is available and consists of a *communication* phase and a *confirmation* phase.

In the communication phase, coded symbols are transmitted and decoded at the receiver. Noiseless feedback informs the transmitter of the decoding result or provides the received symbols so that the transmitter can replicate the decoding result. In the confirmation phase, the transmitter sends a coded ACK/NACK on the forward channel depending on whether decoding was successful. The receiver informs the transmitter through noiseless feedback as to which confirmation message (ACK or NACK) it decoded.

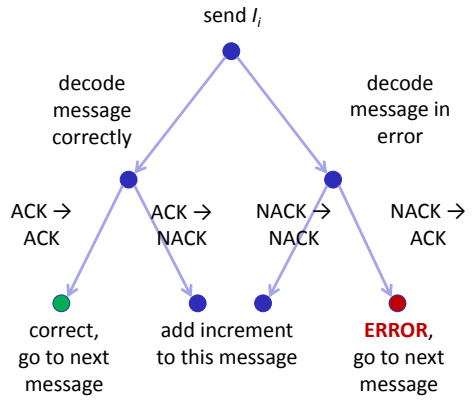

Fig. 1. An illustration of the events that can lead to correct decoding, undetected errors, or additional retransmissions.

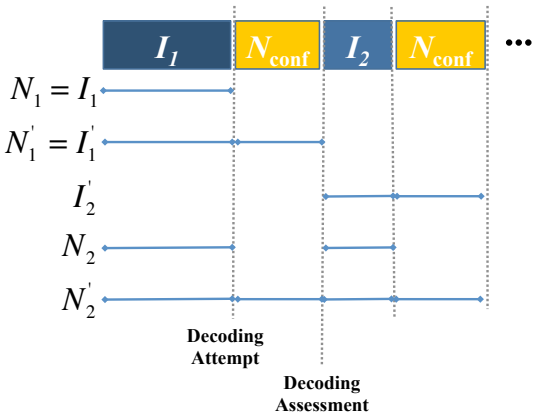

Fig. 2. A notional diagram of the two-phase communication scheme. One two-phase cycle has a communication-phase transmission with I_i symbols and a confirmation-phase transmission with N_{conf} symbols. Thus the number of symbols in the ith cycles is $I_i' = I_i + N_{\text{conf}}$. N_i is the number of communication-phase phase symbols transmitted by the end of the ith cycle. N_i' is the total number of communication-phase phase symbols transmitted by the end of the ith cycle. The dashed vertical lines indicate the communication phase decoding attempts and confirmation phase ACK/NACK decoding assessments.

If the receiver decodes an ACK, it will proceed to the next message. If the receiver decodes a NACK, the two phases are repeated until an ACK is decoded at the receiver in the confirmation phase.

As shown in Fig. 1, message decoding errors only occur when a forward NACK is decoded as an ACK (occurring with probability $p_{n\to a}$), in which case the receiver is unaware that it has decoded incorrectly (i.e., all message errors are undetected errors). When forward ACKs are decoded as NACKs (occurring with probability $p_{a\to n}$), the message incurs additional latency due to retransmission.

B. Two-Phase Scheme with Incremental Redundancy

Fig. 2 shows a diagram of the proposed two-phase incremental redundancy scheme. I_1 symbols are transmitted in the first communication phase and decoded with blocklength N_1. Next, N_{conf} symbols communicating ACK or NACK are transmitted in the confirmation phase. The total number of

symbols in the ith two-phase cycle is $I'_i = I_i + N_{\text{conf}}$. Transmission stops when the receiver has decoded an ACK in the confirmation phase, even if the transmitter actually sent a NACK. If a NACK is decoded, another round of communication and confirmation occurs; I_2 symbols are transmitted and decoded using the cumulative blocklength $N_2 = N_1 + I_2$, and then another N_{conf} symbols are transmitted.

The communication-phase transmission length varies with the transmission index; in the ith transmission it is I_i, and the decoding blocklength is $N_i = N_{i-1} + I_i$. The confirmation-phase transmission length is always N_{conf}. In our scheme, the ith confirmation message is decoded independently of the previous confirmation blocks.

If the receiver still has not decoded an ACK by the mth transmission, the scheme disregards all earlier transmissions and starts over with I_1 communication symbols and N_{conf} confirmation symbols. This is analogous to the protocol of our VLFT implementation in [3]. We note that the concept of m maximum transmissions before repetition also appears in, for example, [10], [11].

The total number of channel uses at the ith decoding attempt, N'_i, is

$$N'_i = \begin{cases} \sum_{j=1}^{i} I'_j\,, & 1 \le i \le m \\ \ell N'_m + \sum_{j=1}^{t} I'_j\,, & i = \ell m + t \end{cases}. \tag{1}$$

This scheme reduces to simple ARQ, but with a separate confirmation phase, when the maximum number of transmissions m is 1.

C. Throughput Analysis of Two-Phase Scheme

We denote the probability of decoding incorrectly in decoding attempt i (i.e., when decoding with blocklength N_i) as $P(\zeta_i)$ and the probability that the decoder picks the correct codeword as $P(\zeta_i^c) = 1 - P(\zeta_i)$. We assume that successful decoding of confirmation messages occurs with equal probability regardless of whether ACK or NACK was sent, i.e., $p_{n\to a} = p_{a\to n}$.

For the Gaussian channel with a simple repetition BPSK code for the confirmation messages and SNR η, we have $p_{n\to a} = Q(\sqrt{N_{\text{conf}}\eta})$, where $Q(u) = \frac{1}{\sqrt{2\pi}}\int_u^\infty \exp\frac{-t^2}{2}dt$ is the tail probability of the standard normal distribution. We denote the probability of success in the confirmation phase as $p_{a\to a} = p_{n\to n} = 1 - p_{n\to a}$.

Let $\mathcal{N}_i$ be the event that the receiver decodes the ith confirmation message as a NACK, which has the following probability:

$$P(\mathcal{N}_i) = \begin{cases} P(\zeta_1)p_{n\to n} + P(\zeta_1^c)p_{a\to n}, & i = 1 \\ P(\mathcal{N}_{i-1})P(\zeta_i|\mathcal{N}_{i-1})p_{n\to n} & \\ \quad + P(\mathcal{N}_{i-1})P(\zeta_i^c|\mathcal{N}_{i-1})p_{a\to n}, & 2 \le i \le m \\ P(\mathcal{N}_m)^\ell P(\mathcal{N}_t), & i = \ell m + t \end{cases}. \tag{2}$$

For convenience we also define $P(\mathcal{N}_0) = 1$. With probability $P(\mathcal{N}_i)$, the transmitter sends increment $i+1$ with length I_{i+1} and another N_{conf} confirmation symbols.

For $2 \le i \le m$, the expression for $P(\mathcal{N}_i)$ in (2) can be expanded into terms similar in form to the two terms in (2) for the $i = 1$ case. The number of these terms grows exponentially with i, but the overall probability is tightly upper bounded by the two dominant terms as follows:

$$\begin{aligned} P(\mathcal{N}_i) <& P(\zeta_1, \zeta_2, \ldots, \zeta_i)p_{n\to n}^i & (3)\\ &+ P(\zeta_1, \zeta_2, \ldots, \zeta_{i-1}, \zeta_i^c)p_{n\to n}^{i-1}p_{a\to n}\,. & (4)\end{aligned}$$

Using the tight approximation $P(\zeta_1, \zeta_2, \ldots, \zeta_i) \approx P(\zeta_i)$, we have the following approximation for $2 \le i \le m$:

$$P(\mathcal{N}_i) \approx P(\zeta_i)p_{n\to n}^i + (P(\zeta_{i-1}) - P(\zeta_i))p_{n\to n}^{i-1}p_{a\to n}\,. \tag{5}$$

Similarly, the probability of undetected error in the ith transmission is well approximated by $P(\text{UE}_i) \approx P(\zeta_i)p_{n\to n}^{i-1}p_{n\to a}$ for $i = 1, \ldots, m$ and $P(\text{UE}_{\ell m+t}) = P(\mathcal{N}_m)^\ell P(\text{UE}_t)$ for integers $\ell \ge 1$ and $1 \le t \le m$. The overall probability of (undetected) error for a message is given by

$$P(\text{UE}) = (1 - P(\mathcal{N}_m))^{-1}\, P(\text{UE}_1^m), \tag{6}$$

where

$$P(\text{UE}_1^m) = \sum_{i=1}^{m} P(\text{UE}_i). \tag{7}$$

Denote the overall transmission length of both phases of the ith increment as $I'_i = I_i + N_{\text{conf}}$. The two-phase scheme's expected latency $\lambda^{(\text{two-phase})}$ (i.e., the average number of channel uses before an ACK) and throughput $R_t^{(\text{two-phase})}$ are computed as follows:

$$\lambda^{(\text{two-phase})} = (1 - P(\mathcal{N}_m))^{-1} \sum_{i=1}^{m} I'_i P(\mathcal{N}_{i-1}), \tag{8}$$

$$\begin{aligned} R_t^{(\text{two-phase})} &= \frac{k}{\lambda^{(2)}}(1 - P(\text{UE})) & (9)\\ &= \frac{k(1 - P(\mathcal{N}_m))(1 - P(\text{UE}))}{\sum_{i=1}^{m} I'_i P(\mathcal{N}_{i-1})} & (10)\\ &= \frac{k(1 - P(\mathcal{N}_m) - P(\text{UE}_1^m))}{\sum_{i=1}^{m} I'_i P(\mathcal{N}_{i-1})}, & (11)\end{aligned}$$

where k is the number of information bits in each attempted message. The expression in (9) includes the factor $(1-P(\text{UE}))$ so that $R_t^{(\text{two-phase})}$ excludes undetected errors and thus only counts messages that are decoded successfully at the receiver.

Rate decreases as additional increments are transmitted. The instantaneous rate at the completion of the ith transmission is $R_i = k/N'_i$.

When considering complexity it is useful to compute the expected number of message decoding attempts D as follows:

$$D = (1 - P(\mathcal{N}_m))^{-1} \sum_{i=1}^{m} P(\mathcal{N}_{i-1})\,. \tag{12}$$

In this work we focus on the average number of channel uses, ignoring the decoding delay and the delay due to round-trip propagation time t_{RTT}. The expected round-trip delay associated with transmission of one message is $2Dt_{\text{RTT}}$.

D. Rate-compatible-Sphere-packing Blocklength Optimization

The expressions (1-12) given in Sec. II-C are general and may be applied to any error correction code and any channel model. In this section, we use the rate-compatible sphere-packing (RCSP) of [3], [12] to approximate the two-phase VLF performance possible for finite-length codes on the additive white Gaussian noise (AWGN) channel with SNR η. The RCSP analysis provides a framework in which $\{I_i\}$ and N_{conf} can be optimized to maximize throughput $R_t^{(\text{two-phase})}$.

RCSP assumes that the code achieves a (geometrically impossible) perfect packing of decoding spheres at each of the jth transmissions ($j = 1, \ldots, m$) and uses a bounded-distance decoder. Even though perfect sphere-packing codes are not possible, ML decoding performance of convolutional codes at short blocklengths can approximate the RCSP analysis [3].

The squared sphere-packing decoding radius corresponding to blocklength N_j is $r_j^2 = N_j(1+\eta)\ 2^{-2k/N_j}$. For a bounded-distance decoder, errors occur when the noise power is larger than the squared decoding radius. The sphere-packing probability of decoding error $P_{\text{SP}}(\zeta_j)$ associated with radius r_j is

$$P_{\text{SP}}(\zeta_j) = P\left(\sum_{\ell=1}^{N_j} z_\ell^2 > r_j^2\right) = 1 - F_{\chi^2_{N_j}}(r_j^2), \qquad (13)$$

where the $z_\ell \sim \mathcal{N}(0,1)$ and $F_{\chi^2_{N_j}}(u)$ is the CDF of a chi-square with N_j degrees of freedom. Note that (13) is the marginal probability of error without conditioning on decoding errors in the $(j-1)$th decoding attempt.

Using the performance of RCSP, $\{I_i\}$ and N_{conf} are optimized to maximize throughput $R_t^{(\text{two-phase})}$ under a specified constraint on $P(\text{UE})$ as follows:

$$\{I_i\}^*, N_{\text{conf}}^* = \arg \max_{\{I_i\}, N_{\text{conf}}} R_t \ \text{ s.t. } \ P(\text{UE}) \le \epsilon. \qquad (14)$$

III. Simulation Results

Fig. 3 shows throughput vs. latency when the overall probability of undetected error is constrained to be less than ϵ=10^{-4}. The curves include VLF converse and random-coding lower bounds following [2], the constrained two-phase RCSP analysis, simulated tail-biting convolutional codes (TBCCs) with m=5 and $P_{\text{UE}} \le 10^{-4}$, the one-phase scheme with CRCs used for error detection, and the theoretical performance limit of finite-length block codes with no feedback from [1].

A. Rate-Compatible Sphere-packing Curve

Table I provides the optimal values $\{I_i\}^*$ and N_{conf}^* according to the RCSP analysis for a variety of k values for $\epsilon = 10^{-4}$. In some cases, $\{I_i\}^*$ and N_{conf}^* were also found for lower values of ϵ as a guide to transmission length selection for the TBCC simulations. RCSP throughput performance is shown in Fig. 3 using the transmission lengths designed to satisfy (14) for $\epsilon = 10^{-4}$ including the $\epsilon = 10^{-4}$ rows of Table I.

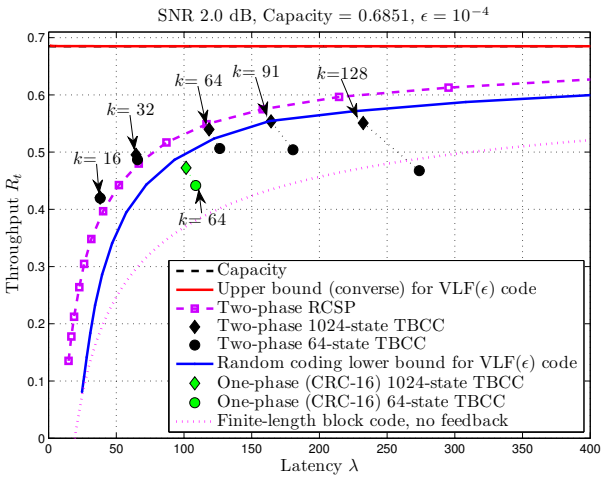

Fig. 3. Throughput vs. latency for VLF converse and random-coding lower bounds following [2], the constrained two-phase RCSP analysis, and simulated tail-biting convolutional codes (TBCCs) with m=5 and $P_{\text{UE}} \le 10^{-4}$. Results of the one-phase scheme with CRCs used for error detection are shown for comparison. Also shown for comparison is the theoretical performance limit of finite-length block codes with no feedback from [1].

TABLE I
Optimal transmission lengths $\{I_i\}^*$ and N_{conf}^* using RCSP, as in (14), with $m = 5$, and SNR $\eta = 2$ dB.

Info. Bits k	Transmission Lengths $\{I_1^*, I_2^*, I_3^*, I_4^*, I_5^*\}$	Confirmation Block N_{conf}^*	Target Error ϵ
16	(30, 8, 6, 7, 9)	7	10^{-4}
32	(50, 10, 8, 8, 14)	8	10^{-4}
64	(98, 12, 10, 12, 18)	8	10^{-4}
64	(101, 12, 10, 12, 18)	9	3.33×10^{-5}
91	(139, 14, 14, 14, 22)	8	10^{-4}
91	(133, 18, 14, 14, 22)	10	2.50×10^{-5}
128	(192, 18, 22, 14, 30)	8	10^{-4}
128	(192, 18, 14, 18, 26)	10	2.50×10^{-5}

B. Two-Phase Scheme with Tail-Biting Convolutional Codes

The TBCC simulations use $\{I_i\}$ and N_{conf} values that are slightly different from those in Table I. The values that were used in the TBCC simulations are shown in Table II. Only convolutional code simulations achieving $P(\text{UE}) \le \epsilon = 10^{-4}$ are shown in Fig. 3.

We restrict our attention to tail-biting implementations of these convolutional codes because the throughput efficiency advantage is important for the relatively small blocklengths we consider. A simple repetition BPSK code is used in the confirmation phase, which only communicates a single bit (ACK or NACK).

Table III, taken from [13, Table 12.1], lists the rate-1/3 convolutional codes that were used as the mother codes for our simulations. Each has the optimum free distance d_{free}. The higher-rate codewords used for the rate-compatible initial and subsequent transmissions are created by pseudorandom rate-compatible puncturing of the rate 1/3 mother codes.

We note that the simulations use the binary-input AWGN (BI-AWGN) channel with soft-decision decoding. In contrast, the VLF(ϵ) converse and random-coding lower bound shown in this paper apply to the full (real-valued) AWGN channel, which has capacity $C_{\text{AWGN}} = \frac{1}{2}\log(1+\eta)$. However at SNR 2.0 dB the restriction to binary inputs is not a significant factor.

C. Information Theoretic Curves

Fig. 3 also includes information-theoretic limits on the maximum rate attainable at short blocklengths, both with and without feedback. The "Finite-length block code, no feedback" curve uses the Gaussian channel dispersion to compute the maximum rate, which tightly approximates both the achievability and converse bounds when there is no feedback [1].

The "Upper bound (converse) for VLF(ϵ) code" and "Random coding lower bound for VLF(ϵ) code" curves are converse (upper) and achievability (lower) bounds for variable-length feedback codes from [2, Theorem 4] and [2, Theorem 3], respectively, particularized to the AWGN channel. Computational details of these bounds for AWGN can be found in [14].

Fig. 3 demonstrates that convolutional codes using the two-phase scheme can deliver throughput at least as high as the VLF(ϵ) random-coding lower bound at low latencies. We observe that the m=5 two-phase scheme beats the VLF(ϵ) random-coding achievability bound despite the fact that the random-coding bound allows the decoder to terminate after transmission of any individual symbol.

At the shortest blocklengths ($k \in \{16, 32\}$), Fig. 3 shows that ML-decoded convolutional codes provide slightly better error performance than the bounded-distance RCSP prediction, which results in superior throughput. As k increases, however, the convolutional code performance begins to fall far short of the RCSP throughput. Eventually, even the 1024-state convolutional code performance lags that of the VLF lower bound.

The poor performance of the convolutional codes for larger values of k is expected because the free distance of the convolutional codes does not improve once the blocklength of the mother code has exceeded the analytic traceback depth L_D [15]. In contrast, RCSP and the VLF random-coding lower bound both have code performance continuing to improve as blocklength increases. It is an interesting area of future investigation to identify codes that perform well, and perhaps

TABLE II
TRANSMISSION LENGTHS $\{I_i\}$, N_{CONF} AND SIMULATED P(UE) FOR SIMULATIONS WITH $m = 5$ AND SNR $\eta = 2$ DB. ONLY SIMULATIONS ACHIEVING $P(\text{UE}) \le \epsilon = 10^{-4}$ ARE SHOWN IN FIG.3.

k	$\{I_1, I_2, I_3, I_4, I_5\}$	N_{conf}	Simulated P(UE)	
			64-state	1024-state
16	(29, 7, 7, 7, 9)	7	7.90×10^{-5}	5.71×10^{-5}
32	(50, 10, 8, 8, 12)	8	6.80×10^{-5}	7.34×10^{-5}
64	(95, 14, 12, 12, 18)	8	2.17×10^{-4}	2.27×10^{-4}
64	(98, 12, 12, 12, 18)	9	6.30×10^{-5}	5.48×10^{-5}
91	(143, 14, 12, 14, 22)	9	8.00×10^{-5}	4.46×10^{-5}
128	(192, 18, 14, 14, 26)	8	3.97×10^{-4}	2.50×10^{-4}
128	(192, 18, 14, 14, 26)	10	6.00×10^{-5}	4.55×10^{-5}

TABLE III
GENERATOR POLYNOMIALS g_1, g_2, AND g_3 CORRESPONDING TO THE RATE 1/3 CONVOLUTIONAL CODES USED IN SIMULATIONS. d_{FREE} IS THE FREE DISTANCE, $A_{d_{\text{FREE}}}$ IS THE NUMBER OF CODEWORDS WITH WEIGHT d_{FREE}, AND L_D IS THE ANALYTIC TRACEBACK DEPTH.

# memory elements, ν	# states, $s = 2^\nu$	(g_1, g_2, g_3)	d_{free}	$A_{d_{\text{free}}}$	L_D
6	64	(117, 127, 155)	15	3	21
10	1024	(2325, 2731, 3747)	22	7	34

TABLE IV
GENERATOR POLYNOMIALS FOR FOUR "GOOD" A-BIT CRCS FROM [5]. THE GENERATOR NOTATION IS HEXADECIMAL. FOR EXAMPLE, 0XCD INDICATES A POLYNOMIAL OF $x^8 + x^7 + x^4 + x^3 + x + 1$. SIMULATED PROBABILITIES OF UNDETECTED ERROR IN THE ONE-PHASE IR SCHEME FOR THE 2 DB AWGN CHANNEL WITH m=5 AND k=64 ARE SHOWN, CORRESPONDING TO POINTS IN FIG. 3.

Generator Polynomial	# States in Conv. Code	Simulated P(UE)	Throughput $R_t^{(\text{CRC})}$	Latency $\lambda^{(\text{one-phase})}$
0x9 (A=4)	64	9.34×10^{-2}	0.5140	105.8
0x9 (A=4)	1024	8.38×10^{-2}	0.5502	99.9
0xcd (A=8)	64	7.05×10^{-3}	0.5125	108.5
0xcd (A=8)	1024	5.31×10^{-3}	0.5486	101.5
0xc07 (A=12)	64	4.01×10^{-4}	0.4783	108.7
0xc07 (A=12)	1024	2.86×10^{-4}	0.5097	102.0
0x8810 (A=16)	64	6.20×10^{-5}	0.4417	108.7
0x8810 (A=16)	1024	6.25×10^{-5}	0.4725	101.6

exceed the VLF(ϵ) random-coding lower bound, for latencies between 200 and 600 symbols.

D. Error Detection with CRCs

In contrast with the two-phase scheme, we now discuss results of simulated convolutional codes using the optimal transmission lengths $\{I_i\}$ identified in the original one-phase scheme (assuming perfect sphere-packing) [3] with the addition of CRCs used for error detection (in place of the genie). The CRC polynomials used are from [5] and are listed in Table IV, along with the probabilities of undetected error when used in the m=5 one-phase IR scheme with k=64 information bits. Of the CRCs simulated, only the 16-bit CRC yielded an error probability below ϵ=10^{-4}. Fig. 3 shows the throughput corresponding to the 16-bit CRC used for error detection.[1] The throughput plotted in Fig. 3 counts only the non-CRC information bits that are passed through the channel error free. For an A-bit CRC and k input bits, the throughput of the one-phase scheme is reduced according to $R_t^{(\text{CRC})} = R_t^{(\text{one-phase})}(k-A)/k = (k-A)/\lambda^{(\text{one-phase})}$, where the latency $\lambda^{(\text{one-phase})}$ for the one-phase scheme is

$$\lambda^{(\text{one-phase})} = \frac{I_1 + \sum_{i=2}^{m} I_i P\left(\bigcap_{j=1}^{i-1} \zeta_j\right)}{1 - P\left(\bigcap_{j=1}^{m} \zeta_j\right)}. \tag{15}$$

CRCs are widely used in industry, are simple to implement, and allow reasonable probabilities of undetected error when

[1]Shorter CRC lengths yield higher throughput but are not included in the plot because the probability of error failed to meet the ϵ=10^{-4} constraint.

chosen appropriately. However, the optimal error detection capabilities are not generally well-understood [5]. When used in conjunction with error correction codes, the error analysis becomes more complicated[2].

For ML-decoded convolutional codes, the decoder is constrained to choose a valid codeword. Thus bit errors in a decoded convolutional codeword cannot be accurately modeled as coming from an i.i.d. binary symmetric channel. This significantly complicates the analysis of CRC performance.

As such, selecting an appropriate CRC length and polynomial to be used in an IR scheme is difficult. System designers may be tempted to be conservative in the choice of CRCs (picking long CRCs), such that the probability of undetected error is extremely small, but this will significantly reduce the throughput, as shown in Fig. 3 for the 16-bit CRC.

Thus, we conclude that for the case of m=5 transmissions and k=64 information bits, the two-phase scheme provides better throughput with a smaller probability of undetected error than the one-phase scheme with CRCs. At larger average blocklengths (e.g., $\sim$ 300 bits), the throughput penalty induced by moderate-length CRCs may be small enough that the two-phase scheme is not required.

A further distinction between the two-phase scheme and the CRC method of error detection is that the two-phase approach requires $(k+1)$ bits of feedback per two-phase increment, whereas the CRC approach requires only 1 bit. In both cases, the decoder autonomously determines when to terminate, though only the CRC approach is a stop-feedback code such that the encoded bits X^n are not a function of the previous channel outputs Y^{n-1} [2]. The two-phase IR scheme improves the reliability precisely because the transmitter uses noiseless feedback to confirm or deny the decoder's decisions.

IV. Future Work

We further investigate the rates achievable at short blocklengths with stop-feedback codes in [14], in which the receiver uses the reliability output Viterbi algorithm (ROVA) [17] to evaluate a stopping rule based on the target error probability ϵ. As soon as the probability that the ML codeword is correct is at least $(1-\epsilon)$, the receiver terminates transmission. This one-phase IR scheme is similar to the CRC-based approach, except that the error constraint is guaranteed to be satisfied (given that the blocklength of the mother code is sufficiently long). As such, the problem of choosing the appropriate CRC length is avoided. At short blocklengths, the ROVA approach also delivers rates surpassing the VLF lower bound.

The two phase scheme presented in this paper may be thought of as a rudimentary form of active hypothesis testing [18], [19]. Better performance may well be possible with more advanced active hypothesis testing implementations in which the transmitter not only confirms or rejects the decoding decision but actively helps to refine it based on the current state of the decoder [18], [19].

[2]See, e.g., [16] for a discussion of error detection codes used in ARQ schemes.

V. Conclusion

We have presented a two-phase incremental redundancy scheme which permits high rates to be achieved with low latency. The two-phase scheme uses forward ACK/NACK messages to confirm/reject the receiver's tentative decoding decision. Rate-compatible sphere-packing analysis provided optimized incremental transmission lengths. Using these transmission lengths as a guide, the two-phase scheme using tail-biting convolutional codes exceeded the VLF random-coding lower bound on throughput for short blocklengths. We also simulated tail-biting convolutional codes with CRCs used for error detection in a traditional one-phase scheme, but they could not match the two-phase performance.

References

[1] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307 –2359, May 2010.

[2] ——, "Feedback in the non-asymptotic regime," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 4903–4925, Aug. 2011.

[3] A. R. Williamson, T.-Y. Chen, and R. D. Wesel, "A rate-compatible sphere-packing analysis of feedback coding with limited retransmissions," in *Proc. 2012 IEEE Int. Symp. Inf. Theory (ISIT)*, Cambridge, MA, USA, July 2012.

[4] M. V. Burnashev, "Data transmission over a discrete channel with feedback. Random transmission time," *Probl. Inf. Transm.*, vol. 12, no. 4, pp. 10–30, 1976.

[5] P. Koopman and T. Chakravarty, "Cyclic redundancy code (CRC) polynomial selection for embedded networks," in *2004 IEEE Int. Conf. Dependable Systems and Networks (DSN)*, July 2004, pp. 145 – 154.

[6] S. Dolinar, K. Andrews, F. Pollara, and D. Divsalar, "Bounds on error probability of block codes with bounded-angle maximum-likelihood incomplete decoding," in *Proc. 2008 IEEE Int. Symp. Inf. Theory and its Applications (ISITA)*, Dec. 2008.

[7] ——, "Bounded angle iterative decoding of LDPC codes," in *Proc. 2008 IEEE Military Commun. Conf. (MILCOM)*, Nov. 2008.

[8] E. Hof, I. Sason, and S. Shamai, "Performance bounds for erasure, list, and decision feedback schemes with linear block codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 8, pp. 3754–3778, Aug. 2010.

[9] P. Berlin, B. Nakiboglu, B. Rimoldi, and E. Telatar, "A simple converse of Burnashev's reliability function," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3074–3080, Jul. 2009.

[10] E. Visotsky, Y. Sun, V. Tripathi, M. Honig, and R. Peterson, "Reliability-based incremental redundancy with convolutional codes," *IEEE Trans. Commun.*, vol. 53, no. 6, pp. 987– 997, June 2005.

[11] E. Uhlemann, L. Rasmussen, A. Grant, and P.-A. Wiberg, "Optimal incremental-redundancy strategy for type-II hybrid ARQ," in *Proc. 2003 IEEE Int. Symp. Inf. Theory (ISIT)*, July 2003, p. 448.

[12] T.-Y. Chen, D. Divsalar, and R. D. Wesel, "Chernoff bounds for analysis of rate-compatible sphere-packing with numerous transmissions," in *Proc. 2012 IEEE Inf. Theory Workshop (ITW)*, Lausanne, Switzerland, Sep. 2012.

[13] S. Lin and D. J. Costello, *Error Control Coding, Second Edition*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2004.

[14] A. R. Williamson, T.-Y. Chen, and R. D. Wesel, "Reliability-based error detection for feedback communication with low latency," submitted.

[15] J. Anderson and K. Balachandran, "Decision depths of convolutional codes," *IEEE Trans. Inf. Theory*, vol. 35, no. 2, pp. 455–459, Mar. 1989.

[16] S. Lin, D. Costello, and M. Miller, "Automatic-repeat-request error-control schemes," *IEEE Commun. Mag.*, vol. 22, no. 12, pp. 5–17, Dec. 1984.

[17] A. Raghavan and C. Baum, "A reliability output Viterbi algorithm with applications to hybrid ARQ," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 1214–1216, May 1998.

[18] M. Naghshvar and T. Javidi, "Active sequential hypothesis testing," *arXiv*, 2012. Available: http://arxiv.org/abs/1203.4626.

[19] ——, "Extrinsic jensenshannon divergence with application in active hypothesis testing." in *Proc. 2012 IEEE Int. Symp. Inf. Theory (ISIT)*, Cambridge, MA, USA, July 2012, pp. 2191–2195.