FEC Systems for Aeronautical Telemetry

ERIK PERRINS, Senior Member, IEEE
University of Kansas

We consider the design of capacity-approaching forward error correction (FEC) systems for use in the aeronautical telemetry environment. The modulation format is the bandwidth-efficient telemetry-group version of shaped-offset quadrature phase-shift keying (SOQPSK-TG). The FEC codes are a low-density parity-check (LDPC) code and a serially concatenated convolutional code (SCCC). The block structure of the two types of FEC is designed such that the two code words are formatted as similarly as possible. This unified format allows the flexibility of the FEC-encoded signal to be received by legacy test-range assets that are not equipped with FEC—although no coding gain is realized in such a case. We also show how a wide range of near-optimal SOQPSK demodulators can be paired with the FEC decoders; this includes the most widely deployed SOQPSK demodulator (the so-called symbol-by-symbol (SxS) demodulator), which so far has not been considered for use in FEC applications. In all of our demodulator/decoder designs, we apply techniques that are simple and robust and that fully decouple the demodulator's tasks (e.g., synchronization, filtering) from the decoder's tasks (e.g., path metrics, soft outputs). For both LDPC and SCCC, we show that very attractive system designs can perform within 0.5 dB of optimum, while requiring only a fraction of the complexity of the optimum system.

Manuscript received September 21, 2012; revised January 10, 2013; released for publication January 21, 2013.

IEEE Log No. T-AES/49/4/944704.

Refereeing of this contribution was handled by M. Rice.

This paper was presented in part at the International Telemetering Conference (ITC'12), San Diego, CA, Oct. 2012.

Author's address: Department of Electrical Engineering and Computer Science, University of Kansas, 2020 Eaton Hall, 1520 West 15th St., Lawrence, KS 66045, E-mail: (esp@ieee.org).

I. INTRODUCTION

Forward error correction (FEC) codes are a technology that has become standard in nearly all digital communication settings. One arena where FEC has seen only limited application is aeronautical telemetry, although this is in the process of changing. A long period of relative quiet in the flight-test community—going back to the 1970s—reached an end in the late 1990s due a series of spectrum reallocations. The loss of spectrum sparked the development of two new "tiers" of bandwidth-efficient telemetry waveforms as replacements for the inefficient legacy waveform known as PCM/FM. As of 2004, these new waveforms were adopted in the IRIG 106 Standard [1]. Although the spectrum challenge is relatively new, one challenge that is ever present in flight testing is the strict set of size, weight, and power supply (SWaP) constraints that are placed on telemetry systems. Thus, a familiar characteristic of the new waveforms is that they are constant envelope, which allows their bandwidth efficiency to be preserved, even when used with nonlinear amplifiers.

Due to the continued scarcity of spectrum and an ever-increasing number of flight-test articles, the telemetry community is currently migrating away from an "always on" serial-streaming telemetry (SST) architecture and is in the process of developing a burst-mode packet-based architecture known as integrated network enhanced telemetry (iNET) [2]. Long-term plans call for the development of an orthogonal frequency division multiplexing (OFDM) type waveform, which presents formidable challenges when strict SWaP constraints are imposed. In order to mitigate risk, the Communication Link Standards Working Group (CLSWG) adopted the telemetry-group version of shaped-offset quadrature phase-shift keying (SOQPSK-TG) as the initial iNET physical-layer waveform [3]; SOQPSK-TG is one of the constant-envelope tier I SST waveforms in IRIG 106 [1].

The pairing of SOQPSK-TG with FEC codes was first studied in [4], where turbo product codes were considered for SST systems, mainly because of the commercial availability of decoding hardware. In [5], [6] a number of candidate coding schemes for SOQPSK-TG were considered for future use in iNET; out of this study, serially concatenated convolutional codes (SCCCs) emerged as a leading candidate. Concurrently, low-density parity-check (LDPC) codes paired with binary phase-shift keying (BPSK) were being developed at the Jet Propulsion Laboratory (JPL) for use in deep space applications [7]; these codes were deemed of interest to iNET mainly due to the maturity of hardware decoders for LDPC. Based on the available preliminary data, the iNET CLSWG adopted one of the JPL LDPC codes as its primary FEC selection and designated an SCCC as its alternate selection [3]. While these selections marked a clear step forward, many important questions have yet to be addressed, such as the performance of LDPC paired with SOQPSK-TG, the performance of the two coded system relative to channel capacity, and the design of reduced-complexity (i.e., realizable) demodulator/decoder systems.

In this paper we address questions such as these and give an in-depth, side-by-side study of these LDPC and SCCC systems for the aeronautical telemetry environment. We begin with a full description and specification of the coding schemes themselves; for the SCCC scheme in particular, this includes enhancements to the interleaving and puncturing schemes from [6] that yield a systematic code word with a rate of exactly 2/3, which matches the LDPC code word and allows greater interoperability with legacy test-range receiving equipment. Our numerical results include a discussion on channel capacity, where we demonstrate that these coding schemes perform within 1.0 dB of their respective theoretical limits.

After describing the air interface, we devote the remainder of the paper to demodulator and decoder design. In developing these systems we apply novel techniques that significantly reduce complexity and improve robustness in exchange for minimal losses in performance (i.e., we apply near-optimal techniques). All of our proposed systems have demodulators that are completely decoupled from the FEC decoders (i.e., the demodulators are completely outside the decoding "loop"), which makes them more straightforward to realize in practice. Also, none of the systems require estimation of additional nuisance parameters, such as signal-to-noise ratio (SNR), which further reduces their complexity. The significance of our demodulator/decoder systems is that they are simple, robust, and give telemetry system architects the flexibility of a "mix-and-match" design approach, with a wide variety of demodulators and decoders at their disposal. In particular we apply the OQPSK-type symbol-by-symbol (SxS) demodulator to FEC applications for the first time. In uncoded systems the suboptimal SxS demodulator is the most widely deployed (and most simple) SOQPSK technology and results in a loss of 1–2 dB depending on its implementation. We show that in coded systems, the SxS demodulator is suboptimal by only 0.5–0.8 dB, which makes it a very attractive design option given its simplicity.

The paper is organized as follows. In the next section we describe the transmitter model, including the two FEC codes and the SOQPSK signal model. In Section III we develop the individual demodulator and decoder modules, and in Section IV we show how these modules are combined into full systems. In Section V we give a comprehensive set of numerical results that includes the bit error rate (BER) performance of the near-optimal demodulators by themselves (i.e., the uncoded case), the performance of the coded systems relative to channel capacity, and the BER performance of the coded systems using the proposed near-optimal demodulator/decoder architectures.

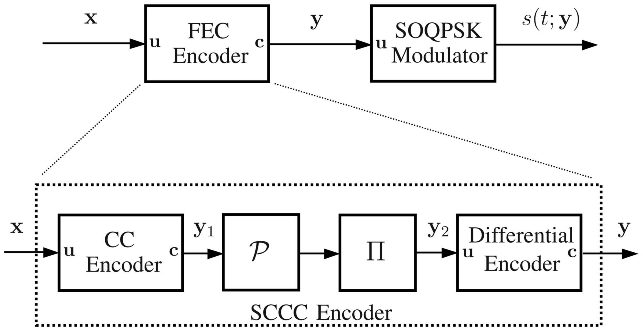

Fig. 1. Transmitter model.

II. TRANSMITTER MODEL

The transmitter model is shown in Fig. 1. The information word (sequence) is denoted as $\mathbf{x} \overset{\Delta}{=} \{x_i\}_{i=0}^{K-1}$, where K is the number of bits contained in the information word and each bit has a duration of T_b seconds. The FEC encoder accepts $\mathbf{x}$ as its input and returns the code word (sequence) $\mathbf{y} \overset{\Delta}{=} \{y_i\}_{i=0}^{N-1}$ as its output, where N is the number of symbols contained in the code word and each symbol has a duration of T_s seconds. The FEC encoder has a rate $R \overset{\Delta}{=} K/N$, and we have the relationship $T_s = RT_b$.

As mentioned above the iNET CLSWG adopted SOQPSK-TG as the initial physical-layer waveform, with FEC options of LDPC and SCCC. Based on analysis of link budget and throughput requirements, the CLSWG identified a coding rate of $R = 2/3$ and an information word length of $K = 4096$ bits (code word length of $N = 6144$ symbols) as attractive design choices [3]. Figure 1 reflects the SOQPSK-based transmitter design, where the code word $\mathbf{y}$ is fed to the SOQPSK modulator which then produces the transmitted signal $s(t;\mathbf{y})$. The two FEC options and the SOQPSK-TG waveform are now specified in greater detail.

Notation: As shown in Fig. 1, the FEC systems consist of various constituent encoders and a modulator, with the output of one possibly being connected to the input of another. The information and code words for the overall system are denoted as $\mathbf{x}$ and $\mathbf{y}$, respectively. Within a particular constituent encoder (or the modulator), the generic notation for the input word is $\mathbf{u}$ and the generic notation for the output word is $\mathbf{c}$. These symbols are reused multiple times and their separate meanings are clear because of context.

A. FEC Encoders

1) *LDPC Encoder*: The LDPC code is the $R = 2/3$, $K = 4096$ ($N = 6144$) code developed at

NASA's JPL [7, 8]. This is a quasi-cyclic code in the family known as "Accumulate Repeat-4 Jagged Accumulate" (AR4JA) codes and is fully described in [8]; suffice it to say that its quasi-cyclic structure permits a simplified encoder design. The encoder is systematic, meaning that the first 4096 positions in $\mathbf{y}$ contain the information word $\mathbf{x}$ and that the last 2048 positions in $\mathbf{y}$ contain the parity symbols.

When transmitting an LDPC code word, the SOQPSK modulator in Fig. 1 is operated without differential encoding and the modulator is viewed as being separate and distinct from the code (i.e., a pragmatic coding scheme), as we explain further in the section on decoding algorithms.

2) *SCCC Encoder*: SCCC coding schemes for aeronautical telemetry were originally explored at the University of Kansas [5, 6]. In this paper we present the final SCCC design, which contains certain design enhancements that minimize the format differences between the LDPC and SCCC code words. Specifically, the final SCCC design is systematic and has a rate of exactly 2/3.

The SCCC encoder consists of four submodules, as shown in the expanded view in the lower part of Fig. 1. The first of these submodules is a convolutional encoder that contains the $R = 1/2$, 4-state, systematic feedback (FB) convolutional code (CC) with a generator given by $(7,5)$ in the octal notation. The encoder has one output stream containing the information bits (without alteration) and one output stream containing the parity symbols. The encoder is initialized to the all-zeros state prior to encoding the information word. After the $K =$ 4096 information bits are encoded, two additional termination symbols are appended at the input; the purpose of which is to return the encoder to the all-zeros state. Thus, the CC encoder produces the intermediate code word $\mathbf{y}_1$, which consists of 8196 systematically encoded symbols.

The final coding rate of exactly 2/3 is achieved by puncturing the rate-1/2 code as follows. (The puncturing block is denoted by the symbol $\mathcal{P}$ in the expanded portion of Fig. 1.) A rate-2/3 puncturing pattern [9] is applied a total of 2042 times, puncturing one parity symbol each time; a rate-3/4 puncturing pattern [9] is applied a total of 4 times, puncturing two parity symbols each time; finally, the two parity symbols associated with the termination symbols are punctured entirely. The net result of this puncturing scheme is that an information word of exactly $K =$ 4096 bits gives rise to a code word of exactly $N =$ 6144 symbols.

The next portion of the SCCC encoder is the S-random interleaver [10], denoted by the symbol Π in the expanded portion of Fig. 1 and whose output is the intermediate code word $\mathbf{y}_2$. In our enhanced design the interleaver is constrained such that the first 4096 positions in $\mathbf{y}_2$ contain the information bits (in permuted order of course), and the last 2048 positions in $\mathbf{y}_2$ contain the 2046 parity symbols and the 2 termination symbols (also in permuted order). This "systematic" interleaving scheme has no detrimental impact on performance[1] or run-time complexity; this is demonstrated later in Section V-D and in Fig. 14. The motivation for this systematic design is that of system flexibility. Because test ranges are outfitted with a heterogenous mix of transmitters and receivers/decoders, it is possible to have a ground station that is not equipped with an FEC decoder. The common (and systematic) format of the two types of FEC codes makes them more friendly for this application environment. In the event that an SCCC (or LDPC) code word is received by a ground station without an FEC decoder, the systematic design makes it possible to easily extract the information bits (without realizing the coding gain of course).

The final SCCC encoding step is differential encoding, as shown in the expanded portion of Fig. 1. The differential encoder (DE) plays the role of the inner code (with $R = 1$) in the serially concatenated coding scheme [11, 12]. We drive the encoder with $\mathbf{u} = \mathbf{y}_2$ and the output is the final code word $\mathbf{y}$. The input/output relationship of this encoder is

$$c_i = u_i \oplus c_{i-2}, \qquad u_i, c_i \in \{0,1\} \tag{1}$$

where $\oplus$ denotes the logical XOR operation. This is the so-called double DE [13]. We define the even-indexed inputs as $\{u_i\}_{i\text{-even}}$ and the odd-indexed inputs as $\{u_i\}_{i\text{-odd}}$. The double DE can be thought of as two separate single DEs, where one encodes the even-indexed inputs and the other encodes the odd-indexed inputs. Each single DE has a two-state trellis, which is used later in the decoder.

Unlike the CC encoder, where we assumed the encoder was initialized to the all-zeros state, we assume here that the DE is initialized with $c_{-2} = 1$ and $c_{-1} = 0$, which is done for reasons of compatibility with the SOQPSK modulator that follows. Also, the DE is not terminated to a known state, which results in lower quality decoding estimates for the last few symbols; however, this uncertainty is widely distributed over the entire $N = 6144$ code word via the deinterleaver and thus has a negligible impact on the overall performance of the code.

B. SOQPSK Signal Model

SOQPSK is a special case of continuous phase modulation (CPM), where the input symbols undergo a precoding operation that permits the use of SxS OQPSK-type detectors at the receiver. In other words, the precoder converts a CPM modulator into an approximation of an OQPSK modulator, albeit one

[1] In fact we were able to find a systematic interleaver with $S =$ 57, which is slightly higher than the expected upper value of $\sqrt{6144/2} \approx 55.4$ [10].

that (advantageously) produces a constant-envelope signal. The output of the modulator is denoted as $s(t;\mathbf{y})$ because the demodulator's task is to undo the modulation and return an estimate of $\mathbf{y}$.

Within the modulator the input to the precoder is $\mathbf{u}$ and output of the precoder is the sequence $\boldsymbol{\alpha} \overset{\Delta}{=} \{\alpha_i\}_{i=0}^{N-1}$, with the precoder operation given by

$$\alpha_i = (-1)^{i+1}(2u_{i-1} - 1)(u_i - u_{i-2}) \tag{2}$$

where $\alpha_i \in \{-1, 0, +1\}$ and $u_i \in \{0, 1\}$. A given ternary symbol α_i has the same duration as an input symbol u_i, namely T_s.

The remainder of the operations of the SOQPSK modulator are those of a standard CPM modulator that is driven by $\boldsymbol{\alpha}$. The phase of the SOQPSK signal is

$$\phi(t;\mathbf{y}) = 2\pi h \sum_{i=0}^{N-1} \alpha_i q(t - iT_s). \tag{3}$$

We assume that the initial phase of the signal is zero, which corresponds to initial values of $u_{-2} = 1$ and $u_{-1} = 0$ in (2) (which is in agreement with the initialization of (1), according to [14, Fig. 4]). For all versions of SOQPSK, the modulation index h has a value of $h = 1/2$. The phase response $q(t)$ is the time-integral of the frequency pulse $g(t)$. The function $g_{\mathrm{TG}}(t)$ that is used to generate the SOQPSK-TG waveform is defined in [1] and $g_{\mathrm{TG}}(t)$ and $q_{\mathrm{TG}}(t)$ are plotted in [14, Fig. 1]. The final operation within the modulator is to form the complex-baseband SOQPSK waveform given by

$$s(t;\mathbf{y}) \overset{\Delta}{=} \exp\{j\phi(t;\mathbf{y})\}. \tag{4}$$

The optimal demodulator for SOQPSK makes use of the Viterbi algorithm (VA). This is because SOQPSK—like other CPMs—is a modulation with memory. Reduced-complexity near-optimal demodulators have also been proposed [14] that make use of the VA. However, the most widely used demodulation approach in practice is the simple OQPSK-type SxS demodulator. In this paper we study the effectiveness and performance of both major types of simplified SOQPSK demodulators (VA and SxS) when paired with FEC.

III. DEMODULATOR/DECODER MODULES

The received signal is modeled as

$$r(t) = \sqrt{\frac{E_s}{T_s}} s(t;\mathbf{y}) + w(t) \tag{5}$$

where E_s is the energy per symbol and $w(t)$ is complex-valued additive white Gaussian noise (AWGN) with zero mean and power spectral density N_0. The energy per symbol and energy per bit have the relationship $E_s = RE_b$. For the sake of simplicity, the received signal model in (5) omits synchronization variables (specifically, symbol timing and carrier phase offsets); however, because these are important problems in their own right, we point out existing synchronization solutions in the development below. Because SOQPSK is treated as part of the code in the case of SCCC, the demodulator then becomes part of the FEC decoding loop, which greatly complicates a hardware implementation. A key goal and contribution of this paper is to devise methods that "decouple" the demodulation and decoding operations of the FEC system; this results in "stand alone" demodulators and FEC decoders.

The demodulators and decoders presented below exchange soft inputs and soft output, meaning that information is not hard-quantized into "black and white" values, but instead has a sense of reliability or "shades of gray." Before specifying the demodulators and decoders, we first define the soft information formats that are employed herein.

A. Soft Information Formats

The log likelihood ratio (LLR), which applies to a binary random variable U_i, is defined as $\lambda_i = \ln(P_i(U = 1)/P_i(U = 0))$, where $P_i(U = 1)$ and $P_i(U = 0)$ are the two probabilities that make up the binary probability mass function (pmf) for U_i. The LLR has great intuitive meaning because its sign corresponds to a maximum likelihood hard decision on U_i and its magnitude indicates the reliability of this decision. The scaling of λ_i has a precise physical meaning also, because it preserves the actual probabilities when going back and forth from the log domain to the linear domain. If the precise scaling is not maintained, the intuitive meaning of the LLR is not affected. In a receiver, precise scaling of LLRs requires an accurate SNR estimate, which can be difficult to obtain in a dynamic wireless environment such as an aeronautical test range.

In our FEC systems we employ log-domain probabilities where the precise scaling is unknown; this appreciably simplifies the receiver at the expense of a few tenths of a dB in performance, as shown later in our numerical results in Section V. The two specific soft information formats that are of interest here are the following.

Scaled LLR, which applies to binary random variables and is defined as $\mu_i = \alpha\lambda_i$, where $\alpha > 0$ is an unknown constant. For a sequence of binary random variables $\mathbf{U}$, we have $\boldsymbol{\mu} = \{\mu_i\}$.

Scaled log likelihood, which applies to nonbinary random variables and is defined as the scaled log of the individual probabilities in the pmf. The random variable U_i has a pmf $P_i(U = u)$ and a scaled log likelihood $\rho_i(u) = \alpha \ln(P_i(U = u))$. For a random sequence $\mathbf{U}$, we have $\boldsymbol{\rho}(u) = \{\rho_i(u)\}$. For binary random variables, to convert back and forth between $\rho_i(u)$ and μ_i we have $\rho_i(1) = +\mu_i/2$ and $\rho_i(0) = -\mu_i/2$

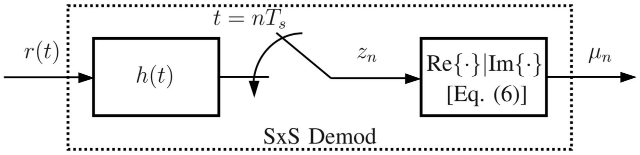

Fig. 2. Soft-output SxS demodulator.

to convert one direction, and $\mu_i = \rho_i(1) - \rho_i(0)$ to convert back.

B. Soft-Output SxS Demodulator

The SxS OQPSK-type demodulator is the simplest and most widely deployed SOQPSK demodulator. The contribution here is to demonstrate how this type of demodulator is used in the larger FEC-based system and also to more widely publish the optimal version of this demodulator.

The SxS demodulator is shown in Fig. 2 and consists of the following

1) a detection filter (DF) with impulse response $h(t)$, which is used in place of a matched filter (MF) because a true MF does not exist for an OQPSK interpretation of SOQPSK;
2) a sampler, which takes one complex-valued sample per symbol interval (i.e., the sample is taken at the instant $t = nT_s$) and whose output is z_n;
3) a block that extracts the real part when n is even and the imaginary part when n is odd (this alternating pattern is due to the OQPSK behavior).

Stated mathematically the soft output is given by

$$\mu_n = \begin{cases} \mathrm{Re}\{z_n\}, & n\text{-even} \\ \mathrm{Im}\{z_n\}, & n\text{-odd.} \end{cases} \tag{6}$$

We note that this soft output correspondes to the code symbols $\mathbf{y} = \{y_i\}_{i=0}^{N-1}$ in Fig. 1.

The BER performance of the SxS demodulator depends on the shape of the DF. In terms of computational complexity, the simplest DF is the integrate-and-dump (I&D) shape, where the impulse response is a rectangular pulse (i.e., non-return-to-zero or NRZ) with a duration of two symbol intervals and the computation of its output does not require any multiplications. The I&D response yields the worst BER performance. The half-sine (HS) response $[\sin(\pi(t/2T_s)),\ 0 \leq t \leq 2T_s]$ yields better performance at the expense of a few multiplications. In [15] a numerical optimization (NO) procedure is developed that uses a stochastic steepest descent technique [16] to find the DF shape that minimizes the BER. The resulting DF shape is plotted[2] in Fig. 3 for SOQPSK-TG (the plot also shows samples taken at $N = 2$ samples per T_s, which is the sample rate used for all demodulators in this study). The NO shape represents the best performing SxS option available. Regarding synchronization, symbol timing and carrier phase can be recovered with standard OQPSK-type techniques, e.g., [17], which are applicable for any DF shape.

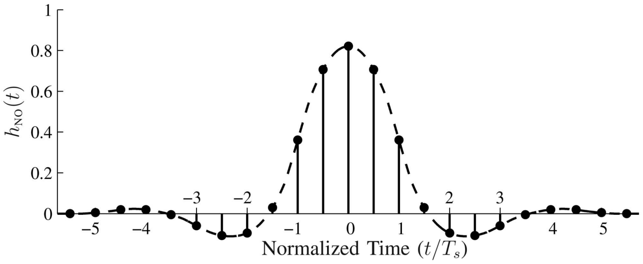

Fig. 3. Numerically optimized DF for SOQPSK-TG sampled at $N = 2$ samples per symbol time.

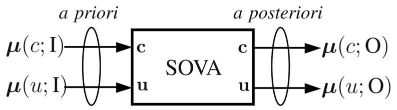
a priori
a posteriori
$\boldsymbol{\mu}(c;\mathrm{I})$
$\boldsymbol{\mu}(u;\mathrm{I})$
c
u
SOVA
c
u
$\boldsymbol{\mu}(c;\mathrm{O})$
$\boldsymbol{\mu}(u;\mathrm{O})$

Fig. 4. Block diagram of generic SOVA module.

C. Soft-Output VA

A generic soft-output Viterbi algorithm (SOVA) module is shown in Fig. 4, and the full algorithm description is found in [18]. On the input (I) side, it receives a priori probability distributions—in the form of scaled LLRs—for both **u** and **c**. The decoding algorithm generates a posteriori output (O) distributions based on the inputs and the code structure (i.e., trellis).[3] This output is always computed for **u**, but is not always needed for **c** (as shown later).

Because of the modulation memory, the optimal trellis for SOQPSK-TG has 512 states; however, near-optimal schemes were developed in [14] that make use of the 4-state time-varying trellis, shown in [14, Fig. 3], which has different sections for n-even and n-odd. We now extend the VA concepts from [14] into the present context, taking into account the SOVA and also the requirement for decoupled demodulation and decoding operations. For a given edge (branch) in the 4-state trellis e, there is an associated output symbol $c(e)$. The output symbol is itself a two-tuple $c(e) = [\theta(e), \alpha(e)]$, where $\theta(e) \in \{0, \pi/2, \pi, 3\pi/2\}$ is the phase state and $\alpha(e) \in \{-1, 0, +1\}$ is the edge symbol.

[2] Although the numerical optimization approach was proposed in [15] for SOQPSK, the resulting filter response for SOQPSK-TG was not published therein (one can assume this was for proprietary reasons). Thus, Fig. 3 is the first published instance of this filter shape.

[3] In addition to the a priori and a posteriori probabilities, there is a third kind known as the extrinsic probability; this is used when the output of one SOVA module is connected to the input of another (as is the case if we look ahead to the SCCC decoders in Figs. 8 and 9). The extrinsic probability is simply the a posteriori normalized by the a priori; in the log domain the extrinsic probability is then $\boldsymbol{\mu}(u;\mathrm{O}) - \boldsymbol{\mu}(u;\mathrm{I})$.

The metrics within the SOVA are computed using sampled MF outputs. As with the trellis the optimal MF bank is quite large and consists of 511 filters. Two very effective approaches for reducing the size of the MF bank were studied in [14]. The first method is called pulse truncation (PT) [19] and it achieves a complexity reduction by truncating the memory of the frequency pulse $g_{\mathrm{TG}}(t)$ down to a single symbol interval; this results in an MF bank with three filters, one for each ternary α. The second method is called pulse amplitude modulation (PAM) [20] and it achieves a complexity reduction by decomposing the SOQPSK-TG signal into many components and then exploiting the fact that nearly all of the signal energy is contained in the first two components; this results in an MF bank with two filters, the outputs of which are combined in three different ways, one for each ternary α. Thus, either implementation results in a complex-valued sample from the MF bank for each ternary α at each time step n, which we denote as $z_n(\alpha)$. In order to obtain $\{z_n(\alpha)\}$, synchronization can be achieved by the CPM-based joint carrier and timing synchronization scheme in [21], which has been extended to PAM and SOQPSK in [22]–[24]; these synchronization schemes are intertwined with the VA because they make use of tentative path survivors (i.e., they are decision-directed algorithms).

Because it is nonbinary $c(e)$ has a log likelihood representation. The scaled, a priori, log likelihood for $c(e)$ is

$$\rho_n(c(e);\mathrm{I}) = \mathrm{Re}\{\exp\{-j\theta(e)\}z_n(\alpha(e))\}. \tag{7}$$

Because $\boldsymbol{\rho}(c;\mathrm{I})$ is a sufficient statistic (as is $\{z_n(\alpha)\}$), then $r(t)$ can be considered demodulated once these variables are obtained and they can be used in place of $r(t)$ in any subsequent processing. As such we construct two specialized forms of the SOVA for SOQPSK.

In Fig. 5(a) we have the "SOQPSK demod," which has $r(t)$ as its upper input and its lower input is unconnected because no a priori input on $\mathbf{u}$ is initially available. The SOQPSK demod computes $\boldsymbol{\rho}(c;\mathrm{I})$, which it outputs for subsequent use, and its edge (branch) metric increment at time step n is

$$\gamma_n(e) = \rho_n(c(e);\mathrm{I}). \tag{8}$$

In Fig. 5(b) we have the "SOQPSK decoder," which does not have any MF or synchronization requirements because it makes use of $\boldsymbol{\rho}(c;\mathrm{I})$ as an input. Because an a priori input on $\mathbf{u}$ is available in this case (as explained later), its edge (branch) metric increment at time step n is

$$\gamma_n(e) = \rho_n(u(e);\mathrm{I}) + \rho_n(c(e);\mathrm{I}) \tag{9}$$

where $u(e)$ is the input symbol associated with a given edge e in the SOQPSK trellis. The computation of the upper soft output is not required for either of the two SOQPSK SOVAs.

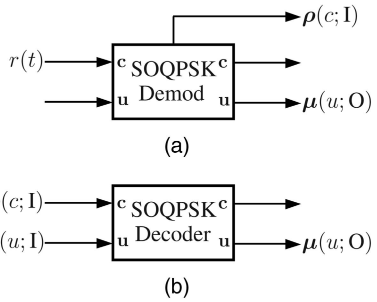
$\boldsymbol{\rho}(c;\mathrm{I})$
$r(t)$
c SOQPSK c
Demod
u u
$\boldsymbol{\mu}(u;\mathrm{O})$
(a)
$\boldsymbol{\rho}(c;\mathrm{I})$
$\boldsymbol{\mu}(u;\mathrm{I})$
c SOQPSK c
u Decoder u
$\boldsymbol{\mu}(u;\mathrm{O})$
(b)

Fig. 5. Block diagrams of two specialized SOVA modules for SOQPSK. The one in (a) performs all tasks typically associated with demodulator (such as synchronization and matched filtering) plus those of decoder (path metrics, soft outputs), while the one in (b) performs only tasks typically associated with decoder.

The SOVA for the CC encoder follows the generic form in Fig. 4. Each edge in the 4-state CC encoder trellis is labeled with an input symbol $u(e)$ and a pair of output symbols $cc(e)$. Here, the metric increment at time step n is

$$\gamma_n(e) = \rho_n(u(e);\mathrm{I}) + \rho_n(cc(e);\mathrm{I}). \tag{10}$$

A SOVA module for the DE in (1) is also of interest. The a priori distribution for the output symbols $\boldsymbol{\mu}(c;\mathrm{I})$ can be separated into even-indexed and odd-indexed sets $\{\mu_i(c;\mathrm{I})\}_{i\text{-even}}$ and $\{\mu_i(c;\mathrm{I})\}_{i\text{-odd}}$, respectively. The even- and odd-indexed sets are decoded separately (and in parallel if desired) by two DE SOVA modules, which operate on the 2-state trellis belonging to a single DE. Each edge in this 2-state trellis is labeled with an input symbol $u(e)$ and a code symbol $c(e)$, with a metric increment at time step n given by

$$\gamma_n(e) = \rho_n(u(e);\mathrm{I}) + \rho_n(c(e);\mathrm{I}). \tag{11}$$

D. Scaled-Min LDPC Decoder

The suboptimal LDPC decoder we employ in this paper is the "scaled-min" decoder [25], where the input $\boldsymbol{\mu}(c;\mathrm{I})$ can be a scaled LLR. The decoding algorithm is identical to that specified in [26, Algorithm 15.2], except that the check node update is computed as

$$\eta_{m,n}^{[l]} = -K\left(\prod_{j\in\mathcal{N}_{m,n}} \mathrm{sign}(-\mu_j^{[l-1]} + \eta_{m,j}^{[l-1]})\right) \times \min_{j\in\mathcal{N}_{m,n}} \{|-\mu_j^{[l-1]} + \eta_{m,j}^{[l-1]}|\} \tag{12}$$

where we have used the same notation established in [26, Algorithm 15.2]. We select the value of the scale factor as $K = 3/4$ for its good performance as well as its simplicity of implementation in hardware (i.e., bit shifting and adding). When computed in a sequential loop, the loop result is initialized as $\eta_{m,n}^{[l]} := +\infty$ (in pseudocode), the loop is executed over the indexes $j \in \mathcal{N}_{m,n}$ with the result updated by

$$\eta^{[l]}_{m,n} := f(\eta^{[l]}_{m,n}, -\mu^{[l-1]}_{j} + \eta^{[l-1]}_{m,j}) \qquad (13)$$

where the two-argument function $f(a,b)$ is defined as

$$f(a,b) = \text{sign}(a)\text{sign}(b)\min(|a|,|b|). \qquad (14)$$

Upon completion of the loop, the loop result is finalized as $\eta^{[l]}_{m,n} := -K\eta^{[l]}_{m,n}$. The algorithm iterates between the check node update and the bit node update (see [26, Algorithm 15.2]) for a total of N_{it} iterations. A motivating feature of (14) (which also applies to (12)) is that $f(\alpha a, \alpha b) = \alpha f(a,b)$ for $\alpha > 0$, which demonstrates that the performance of the scaled-min algorithm is insensitive to the level (scale) of its input.

IV. FEC DEMODULATOR/DECODER SYSTEMS

In the previous sections we have defined two types of soft-output demodulators (SxS and SOVA) and two types of FEC codes (LDPC and SCCC). These can be combined to form four distinct demodulator/decoder systems, the block diagrams of which are shown in Figs. 6–9. Before discussing how each system is different from the others, we first concentrate on the attributes they all have in common. In fact this is one of the main aims of this work: to design FEC systems that are as similar as possible. In particular,

1) Each system has a stand-alone demodulator that is completely decoupled from the FEC decoder.

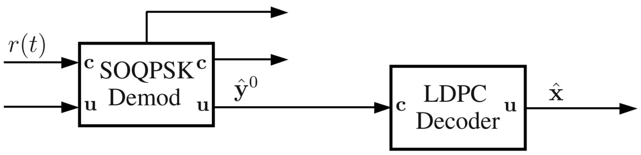

Fig. 6. Block diagram of LDPC system with OQPSK demod.

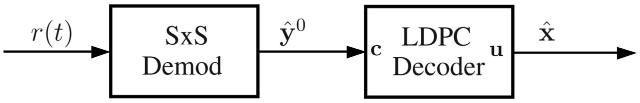

Fig. 7. Block diagram of LDPC system with SxS demod.

The demodulator operates on the received signal $r(t)$, performs all synchronization tasks, and returns an initial soft estimate $\hat{\mathbf{y}}^0$ of the transmitted code word.[4]

2) Each system has a stand-alone FEC decoder that is completely decoupled from the soft-output demodulator. The FEC decoder operates on the demodulator output and returns the estimated information word $\hat{\mathbf{x}}$.

3) Because the code words are systematically encoded and because FEC decoders are relatively costly and less ubiquitous on test ranges, in the event the receiver is not equipped with an FEC decoder, $\hat{\mathbf{x}}$ can be recovered from a hard-limited version of the

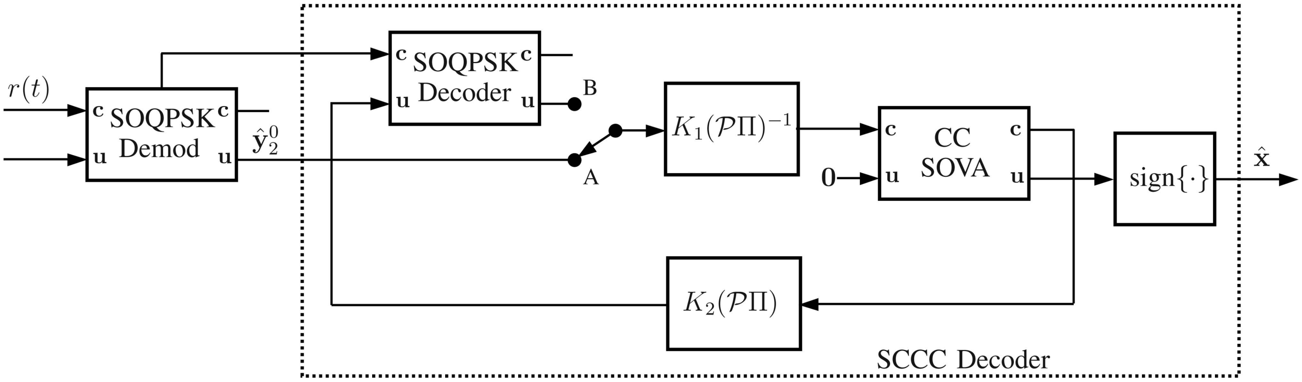

Fig. 8. Block diagram of SCCC system with SOQPSK demod and decoder.

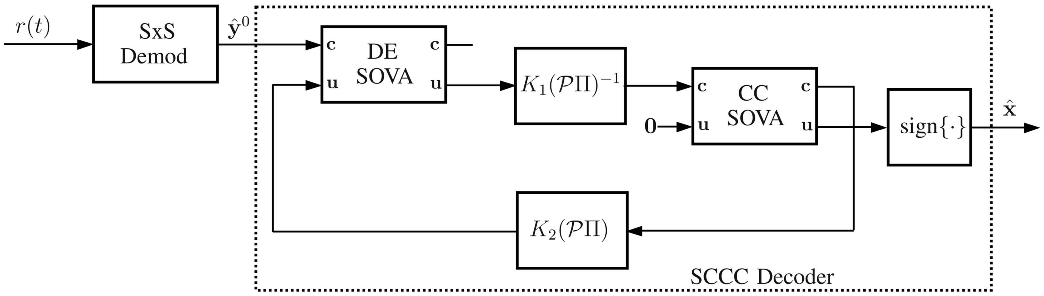

Fig. 9. Block diagram of SCCC system with SxS demod and DE SOVA.

[4] In the case of Fig. 8, the stand-alone demodulator actually returns $\hat{\mathbf{y}}_2^0$, which is a soft estimate of the intermediate code word $\hat{\mathbf{y}}_2$. Although this is slightly different from the other systems, it still achieves the design goals of this effort.

demodulator output with relative ease (but with no FEC performance gain, of course). For the LDPC systems in Figs. 6 and 7, $\hat{\mathbf{x}}$ occupies the first 4096 positions of $\hat{\mathbf{y}}^0$. For the SCCC system in Fig. 8, an interleaved $\hat{\mathbf{x}}$ occupies the first 4096 positions of $\hat{\mathbf{y}}_2^0$ and must therefore be deinterleaved. For the SCCC system in Fig. 9, an interleaved and differentially encoded $\hat{\mathbf{x}}$ occupies the first 4096 positions of $\hat{\mathbf{y}}^0$ and must therefore be differentially decoded and then deinterleaved.

A. LDPC Systems

Because the individual blocks in Figs. 6 and 7 have already been described in full detail above, no additional explanation is needed.

B. SCCC Systems

Because the "SCCC decoder" (i.e., the enclosed subdiagrams in Figs. 8 and 9) is a new entity, further explanation is needed. The specifics of the SCCC decoder depend on which demodulator is being used.

1) *SCCC System with the SOVA Demod*: When the SOVA demod is being used (Fig. 8), it does the first part of the SCCC decoding process. The switch in Fig. 8 is initially in position "A." The demodulator (extrinsic) output is fed to a block that scales it by a factor of $K_1 = 0.75$, deinterleaves it, and depunctures it; these operations are signified by $K_1(\mathcal{P}\Pi)^{-1}$. The CC SOVA does its processing next. Once it is finished its upper (extrinsic) output is scaled by $K_1 = 0.75$, reinterleaved, and repunctured $[K_2(\mathcal{P}\Pi)]$. The SCCC decoder is then ready to commence its second iteration. The switch is now placed in position "B." The SOQPSK decoder is now used from this point on, and the remainder of the processing proceeds as with the first iteration. When N_{it} iterations are finished, the lower output of the CC SOVA is hard limited to produce $\hat{\mathbf{x}}$. Although the SCCC decoding process is shown as a loop, it can also be implemented in a pipelined architecture to yield a much higher decoder throughput.

2) *SCCC System with the SxS Demodulator*: When the SxS demodulator is being used (Fig. 9), the demodulator does no part of the FEC decoding process. Here, the first SCCC decoding iteration begins with the DE SOVA, the lower input of which is set to $\mathbf{0}$ initially. The remainder of the SCCC decoding steps are as explained above. Because the actual inner code is best modeled by the DE plus the SOQPSK modulator, the system in Fig. 9 incurs a small performance penalty, as demonstrated below.

V. PERFORMANCE OF THE FEC SYSTEMS

In this section we provide numerical performance results for the FEC systems described above. In most instances we give performance data for four different demodulators: SOVA demod with PAM filters, SOVA demod with PT filters, SxS demod with NO filter, and SxS demod with I&D filter. Also, reference data is given for the optimal demodulator—maximum a posteriori (MAP) detection [27] with a 512-state trellis [14], which is implemented in the form of a soft-input soft-output (SISO) module [28].

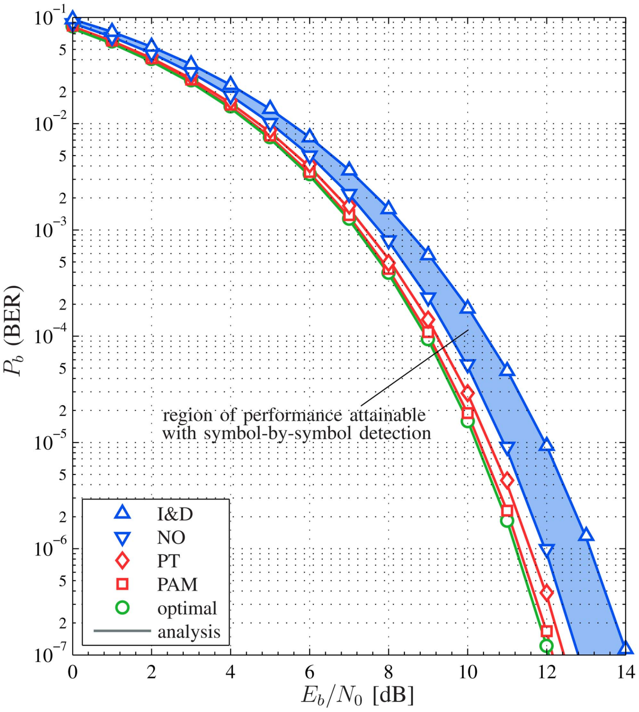

Fig. 10. Probability of bit error (P_b) and BER of uncoded SOQPSK-TG when paired with various near-optimal demodulators and without differential encoding. Discrete simulation points (BER) are plotted separately from analytical curves (solid lines, P_b), but in each case there is strong agreement between simulation and analysis.

D. Uncoded Systems

Figure 10 shows the performance of uncoded systems (without differential encoding), i.e., the stand-alone performance of the demodulators. In each case we show a solid curve for the analytical probability of bit error (P_b) and discrete BER simulation points plotted on top of these curves. For the optimal demodulator the analytical P_b curve is given by

$$P_b = \frac{1}{2}Q\left(\sqrt{1.60\frac{E_b}{N_0}}\right) + \frac{1}{2}Q\left(\sqrt{2.59\frac{E_b}{N_0}}\right). \quad (15)$$

With SOQPSK there are two "error modes." The first is where two competing data sequences deviate by 90° at some symbol interval, then remain apart by 90° during the following symbol interval, and then merge back together during the third symbol interval. This error mode has a normalized squared Euclidean distance (NSED) [29, ch. 3] of 1.60. The second error mode is where two competing data sequences deviate by 90° at some symbol interval, then each go 90° in the opposite direction during the following symbol interval (ending up 90° apart again), and then merge back together during the third symbol interval. This error mode has an NSED of 2.59. Both error modes have an equal number of transmit/receive sequences where they can occur (hence the 1/2 weights in (15)). At low E_b/N_0 both modes occur frequently; however, as E_b/N_0 increases, the second error mode quickly becomes rare.

As Fig. 10 shows the 4-state SOVA demods suffer very little performance loss relative to the optimal 512-state demod, as was shown in [14]. As expected the SOVA demods outperform the SxS demods. However, the SxS NO has very attractive performance given its simplicity; its loss relative to the SOVA cases is only approximately 0.5 dB.

B. Performance Relative to Channel Capacity

We are interested in answering the question of how good these particular LDPC and SCCC codes are, as compared with the theoretical limit given by the channel capacity. The true capacity of the AWGN channel constrained by the use of SOQPSK (or CPM in general) has not been determined, mainly because of the analytical difficulties presented by the nonlinear modulation. Thus, we use the so-called symmetric information rate[5] (SIR) as a practical substitute for the true capacity, as others have done in the literature, e.g., [30]–[32], and we use the term "capacity" synonymously with the term "SIR."

Because SCCCs use the modulation as an integral part of the overall channel code, their performance limit is given by the channel capacity. On the other hand, because LDPC codes place the modulation outside the decoding loop, their performance limit is given by the slightly lesser pragmatic capacity. These general concepts are more fully developed in [32], and were applied specifically to SOQPSK in [33], which resulted in capacity and pragmatic capacity curves for SOQPSK-TG.

Regarding the performance of the particular LDPC and SCCC coding schemes used herein, we consider for a moment the performance optimal counterparts to the demodulator/decoder systems in Figs. 6 and 8; these optimal systems consist of MAP demodulators/decoders that exchange true LLRs and operate on the full 512-state SOQPSK-TG trellis. Let $p_i = P_i(U = 1)$ be the a posteriori output of the decoder. We define the information rate of the code as the average mutual information between the information bits at the encoder input and the soft outputs of the optimal decoder, scaled by the code rate:

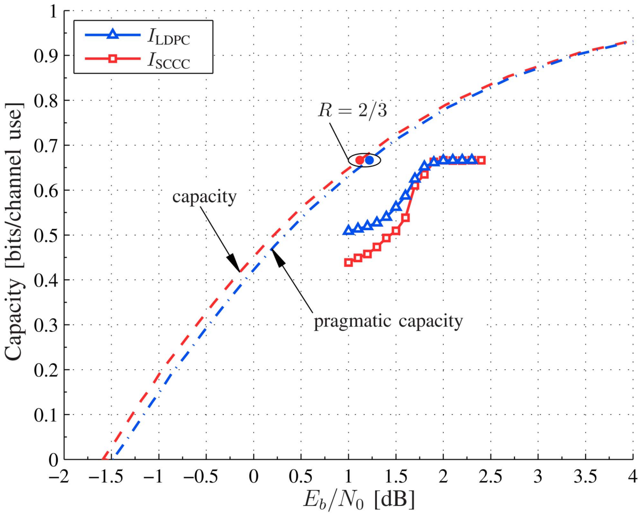

Fig. 11. Information rates of $R = 2/3$ LDPC and SCCC codes as function of E_b/N_0. Information rate of LDPC is approximately 0.8 dB away from its performance limit (pragmatic capacity), while information rate of SCCC is approximately 0.9 dB away from its performance limit (capacity).

$$I_{\text{CODE}} = R\overline{(1 - H(p_i))} \qquad (16)$$

where $H(p)$ is the entropy of a binary random variable (Bernoulli trial) with parameter p and $\overline{(\cdot)}$ denotes the average over time.

In Fig. 11 we plot (16) for LDPC and SCCC as a function of E_b/N_0, along with the capacity and pragmatic capacity reference curves from [33]. The information rates of both codes reach their asymptotic limit of $R = 2/3$ at $E_b/N_0 \approx 2.0$ dB, which places the LDPC code within 0.8 dB of its theoretical limit (pragmatic capacity) and the SCCC within 0.9 dB of its theoretical limit (capacity).

C. LDPC and SCCC with Various Demodulators

Figure 12 shows the BER performance of LDPC-encoded SOQPSK-TG with the four demodulator examples. The LDPC decoder performs a maximum of $N_{\text{it}} = 200$ iterations. A reference curve is also given for the optimal MAP/SISO demod paired with the original LDPC check node update in [26, Algorithm 15.2]. The dominant loss faced by the SOVA-based systems (PAM and PT)—0.2 dB—is actually due to the scaled-min LDPC check node update in (12); the SOVA's 4-state reduced trellis and the approximations in the filter banks have a lesser impact on the performance. The SxS demods incur additional losses because the trellis is omitted and the quality of the soft output is thus reduced. However, it is notable that the BER curves are more tightly bunched in the FEC case; that is to say, the losses in Fig. 12 are smaller for the progressively suboptimal demods than they are in Fig. 10. This result makes an

[5] The true capacity of a channel is the maximum of the mutual information between the transmitted and received symbols over all possible input distributions. The SIR is the mutual information between the transmitted and received symbols assuming that the input distribution is fixed as the uniform distribution.

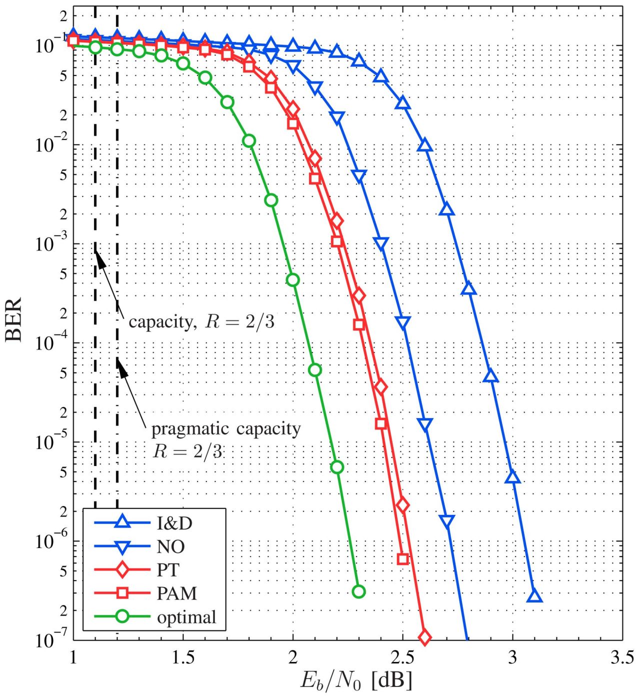

Fig. 12. BER results for LDPC when paired with four different near-optimal demodulator configurations. While relative ordering of performance is same as in Fig. 10, curves are more tightly bunched in this case. This makes simple SxS demodulators an attractive choice in FEC applications.

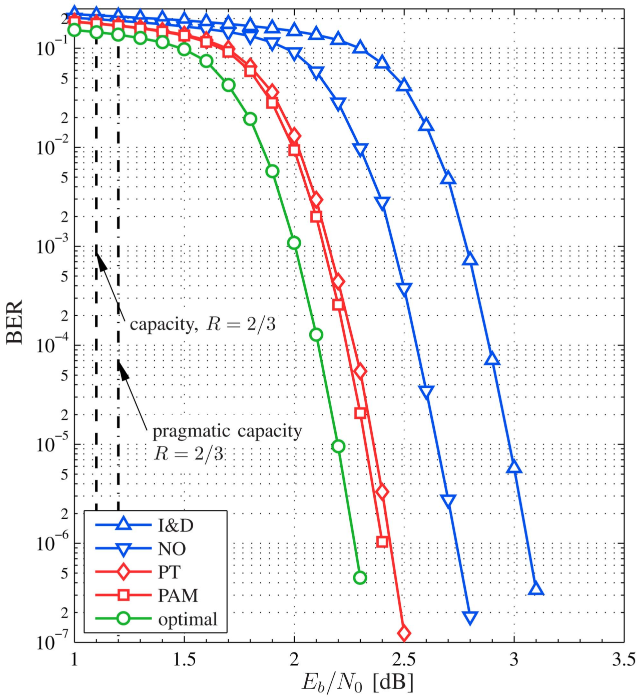

Fig. 13. BER results for SCCC when paired with four different near-optimal demodulator configurations. Here again, relative ordering is as expected but overall drop off in performance is quite small for SxS demodulators, making them an effective solution for FEC applications.

even stronger case for the use of the simple-yet-robust SxS demods: the 0.2 dB loss of the NO SxS demod relative to the SOVA demods is an attractive trade for the simplified implementation.

Figure 13 shows the performance of SCCC-encoded SOQPSK-TG with the four demodulator examples. The SCCC decoder performs $N_{\rm it} = 16$ iterations. A reference curve is also given for the optimal MAP/SISO demod/decoder used in place of the SOVA demod/decoder. As with LDPC the SCCC curves are more tightly bunched than in the uncoded case. Comparing the optimal cases LDPC has a slight advantage over SCCC. However, for the SOVA demods (PAM and PT), the SCCC systems have a slight advantage over LDPC for asymptotically large E_b/N_0 (low BER). Their relatively minor loss of 0.1–0.15 dB relative to optimum is due to the "max-log-MAP" approximation within the SOVA [18], which causes a loss here (but not with LDPC) because the decoding loop repeats the approximation over and over again.

The SxS demods have an additional minor loss component in the SCCC case. This is attributable to the fact that the symbol-energy-to-noise ratio E_s/N_0, at which the SxS demod in Fig. 9 operates, is low enough that both SOQPSK error modes occur frequently. The DE SOVA that follows in Fig. 9 does not have a large enough trellis to resolve these two error modes from each other as the decoding iterations progress. On the other hand, because the SOQPSK decoder in Fig. 8 is fed with 8 log likelihoods that preserve the ±90° transitions of the SOQPSK waveform, the PAM and PT cases are not affected by this additional loss.

Although the detailed reasons behind the losses vary between the LDPC and SCCC systems, the net losses relative to optimum are remarkably similar. This is especially true of the SxS demods, where the NO and I&D cases are suboptimum by 0.45 dB and 0.8 dB, respectively; once again, we conclude that the SxS demods have attractive performance given their simplicity.

Figures 12 and 13 also show reference lines for the capacity and pragmatic capacity of SOQPSK-TG with a coding rate of $R = 2/3$. As noted above the codes perform within around 1 dB of their respective theoretical limits.

D. Enhanced SCCC Design versus Conventional SCCC Design

One final yet important question is whether or not the SCCC enhancements described in Section II-A.2—FB CC, systematic interleaver, trellis termination/puncturing to achieve exactly $R = 2/3$—are worth the design effort, i.e., is the performance better, the same, or worse than the conventional SCCC system in [14]? For the sake of clarity, we note that the conventional system in [14] uses a nonsystematic, feed-forward (FF) CC encoder, a nonsystematic interleaver, and no trellis termination.

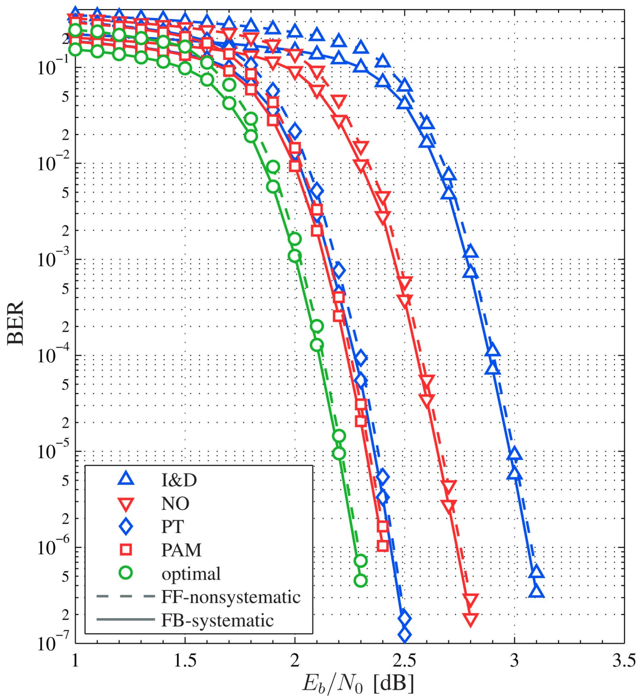

Fig. 14. BER results comparing proposed FB, systematic SCCC format, against conventional FF, nonsystematic SCCC format. A pair of curves is given for each of five demodulator cases. In each demodulator case BER of FF-nonsystematic format (dashed line) is approximately 1.5 times higher than FB-systematic format (solid line).

This question is answered in Fig. 14, where a pair of curves (one for FF-nonsystematic and one for FB-systematic) is plotted for each of the four demodulator examples that have been considered above (along with the optimal demodulator). For each demodulator case the BER of the FF-nonsystematic format (dashed line) is approximately 1.5 times higher than the FB-systematic format (solid line). Thus, at low E_b/N_0 (high BER), the proposed system has an observable performance advantage over the conventional system, and as E_b/N_0 becomes asymptotically large (low BER), the difference in performance becomes vanishingly small.

Of the three SCCC enhancements—FB CC, systematic interleaver, trellis termination/puncturing—we have isolated the FB CC as the dominant cause of the factor of 1.5 difference between the two BERs. Because the FF and FB encoders differ only in the mapping between information sequences and code words, we know that the two encoders have identical code word error rates (WERs). Therefore, the FB-systematic format has a more effective mapping between information sequences and code words in terms of BER, resulting in approximately 1.5 times fewer information bit errors (on average) per code word error event, relative to that of the FF-nonsystematic format.

Given the system design benefits (compatibility with non-FEC demods and greater compatibility between the LDPC and SCCC code words), no complexity disadvantages, and no performance disadvantages, the enhanced SCCC format in Section II-A.2 is an attractive design option.

VI. CONCLUSION

We have developed two types of FEC codes—LDPC and SCCC—for use with SOQPSK-TG in the aeronautical telemetry environment. We have demonstrated that both of these codes perform within 1.0 dB of the channel capacity of SOQPSK-TG. We have also developed two major types of near-optimal soft-output SOQPSK demodulators—SOVA and SxS—and have shown how each is properly implemented in FEC demodulator/decoder systems. The significance of the receiver systems is that they are simple, robust, and flexible. In particular we have demonstrated that a very simple SxS demodulator performs within 0.5 dB of optimum for both FEC codes, which is an attractive performance tradeoff given the minimal amount of implementation complexity.

REFERENCES

[1] Range Commanders Council Telemetry Group, Range Commanders Council
IRIG Standard 106-04: *Telemetry Standards*.
Range Commanders Council Telemetry Group, Range Commanders Council, White Sands Missile Range, New Mexico, 2004. [Online], available: http://www.ntia.doc.gov/osmhome/106.pdf.

[2] Central Test and Evaluation Investment Program (CTEIP)
iNET System Architecture, version 2007.1.
Central Test and Evaluation Investment Program (CTEIP), July 2007.

[3] Integrated Network Enhanced Telemetry (iNET) Communication Link Standards Working Group (CLSWG)
Meeting of the Integrated Network Enhanced Telemetry (iNET) Communication Link Standards Working Group (CLSWG).
Integrated Network Enhanced Telemetry (iNET) Communication Link Standards Working Group (CLSWG), Pasadena, CA, Mar. 17–18, 2008.

[4] Geoghegan, M.
Experimental results for PCM/FM, Tier 1 SOQPSK, and Tier II multi-h CPM with turbo product codes.
Proceedings of the International Telemetering Conference, Oct. 2003.

[5] Damodaran, K. and Perrins, E.
Serially concatenated codes for aeronautical telemetry.
Proceedings of the IEEE Military Communications Conference (MILCOM 2008), San Diego, CA, Nov. 2008, pp. 1–6.

[6] Damodaran, K.
Serially concatenated coded continuous phase modulation for aeronautical telemetry.
Master's thesis, Dept. of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, Dec. 2008.

[7] Andrews, K. S., et al.
The development of turbo and LDPC codes for deep-space applications.
Proceedings of the IEEE, **95** (Nov. 2007), 2142–2156.

[8] Consultative Committee for Space Data Systems (CCSDS)
Low density parity check codes for use in near-Earth and deep space applications.
Consultative Committee for Space Data Systems (CCSDS), 131.1-O-2 Orange Book, Sept. 2007.

[9] Yasuda, Y., Kashiki, K., and Hirata, Y.
High-rate punctured convolutional codes for soft decision Viterbi decoding.
IEEE Transactions on Communications, **COM-32** (Mar. 1984), 315–319.

[10] Divsalar, D. and Pollara, F.
Multiple turbo codes for deep-space communications.
Telecommunications and Data Acquisition Progress Report, May 1995.
[Online], available: http://tmo.jpl.nasa.gov/tmo/progress report/42-121/121T.pdf.

[11] Benedetto, S., et al.
Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding.
IEEE Transactions on Information Theory, **44** (May 1998), 909–926.

[12] Narayanan, K. and Stüber, G.
A serial concatenation approach to iterative demodulation and decoding.
IEEE Transactions on Communications, **47** (July 1999), 956–961.

[13] Simon, M. K.
Multiple-bit differential detection of offset QPSK.
IEEE Transactions on Communications, **51** (June 2003), 1004–1011.

[14] Perrins, E. and Rice, M.
Reduced-complexity approach to iterative detection of coded SOQPSK.
IEEE Transactions on Communications, **55** (July 2007), 1354–1362.

[15] Geoghegan, M.
Optimal linear detection of SOQPSK.
Proceedings of the International Telemetering Conference, San Diego, CA, Oct. 2002.

[16] Widrow, B. and McCool, J.
A comparison of adaptive algorithms based on the methods of steepest descent and random search.
IEEE Transactions on Antennas and Propagation, **AP-24** (Sept. 1976), 615–637.

[17] Rice, M.
Digital Communications: A Discrete-Time Approach.
Upper Saddle River, NJ: Prentice-Hall, 2009.

[18] Fossorier, M. P. C., et al.
On the equivalence between SOVA and max-log-MAP decodings.
IEEE Communications Letters, **2** (May 1998), 137–139.

[19] Svensson, A., Sundberg, C-E., and Aulin, T.
A class of reduced-complexity Viterbi detectors for partial response continuous phase modulation.
IEEE Transactions on Communications, **COM-32** (Oct. 1984), 1079–1087.

[20] Perrins, E. and Rice, M.
PAM representation of ternary CPM.
IEEE Transactions on Communications, **56** (Dec. 2008), 2020–2024.

[21] Morelli, M., Mengali, U., and Vitetta, G. M.
Joint phase and timing recovery with CPM signals.
IEEE Transactions on Communications, **45** (July 1997), 867–876.

[22] Colavolpe, G. and Raheli, R.
Reduced-complexity detection and phase synchronization of CPM signals.
IEEE Transactions on Communications, **45** (Sept. 1997), 1070–1079.

[23] Perrins, E., Bose, S., and Wylie-Green, M. P.
PAM-based timing synchronization for ARTM modulations.
Proceedings of the International Telemetering Conference, San Diego, CA, Oct. 2008.

[24] Chandran, P. and Perrins, E.
Decision-directed symbol timing recovery for SOQPSK.
IEEE Transactions on Aerospace and Electronic Systems, **45** (Apr. 2009), 781–789.

[25] Chen, J., et al.
Reduced-complexity decoding of LDPC codes.
IEEE Transactions on Communications, **53** (Aug. 2005), 1288–1299.

[26] Moon, T. K.
Error Correction Coding: Mathematical Methods and Algorithms.
Hoboken, NJ: Wiley-Interscience, 2005.

[27] Bahl, L., et al.
Optimal decoding of linear codes for minimizing symbol error rate.
IEEE Transactions on Information Theory, **IT-20** (Mar. 1974), 284–287.

[28] Benedetto, S., et al.
A soft-input soft-output APP module for iterative decoding of concatenated codes.
IEEE Communications Letters, **1** (Jan. 1997), 22–24.

[29] Anderson, J. B., Aulin, T., and Sundberg, C-E.
Digital Phase Modulation.
New York: Plenum Press, 1986.

[30] Padmanabhan, K., et al.
General CPM and its capacity.
Proceedings of the International Symposium on Information Theory (ISIT 2005), Adelaide, Australia, Sept. 2005, pp. 750–754.

[31] Arnold, D., et al.
Simulation-based computation of information rates for channels with memory.
IEEE Transactions on Information Theory, **52** (Aug. 2006), 3498–3508.

[32] Perotti, A., et al.
Capacity-achieving CPM schemes.
IEEE Transactions on Information Theory, **56** (Apr. 2010), 1521–1541.

[33] Şahin, C. and Perrins, E.
The capacity of SOQPSK-TG.
Proceedings of the IEEE Military Communications Conference, Baltimore, MD, Nov. 2011, pp. 555–560.

Erik Perrins (S'96—M'05—SM'06) received his B.S. (magna cum laude), M.S., and Ph.D. degrees from Brigham Young University, Provo, UT in 1997, 1998, and 2005, respectively, all in electrical engineering.

From 1998 to 2004 he was with Motorola, Inc., Schaumburg, IL, where he was engaged in research on land mobile radio products. Since August 2005 he has been with the Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, where he is currently an associate professor. Since 2004 he has also been an industry consultant on problems such as reduced-complexity receiver design and receiver synchronization. His current research interests include digital communication theory, synchronization, channel coding, power line communications, and complexity reduction in receivers.

Dr. Perrins is the area editor for *Modulation & Signal Design* for the *IEEE Transactions on Communications*. He is a Member of the IEEE Communications Society and is currently serving as the Secretary of the Communication Theory Technical Committee (CTTC) within ComSoc.