INCOMPATIBILITY OF TRELLIS-BASED NONCOHERENT SOQPSK DEMODULATORS FOR USE IN FEC APPLICATIONS

Erik Perrins
Department of Electrical Engineering & Computer Science
University of Kansas
Lawrence, KS 66045
esp@ieee.org

INTRODUCTION

This paper examines the compatibility/incompatibility of trellis-based noncoherent shaped offset quadrature phase shift keying (SOQPSK) demodulators for use in forward error correction (FEC) applications. The noncoherent demodulators are of the type given in [1, 2]. These demodulators do not *explicitly* estimate and track the phase offset in the received signal, as is done by a coherent demodulator. Instead, they have *implicit* phase estimators associated with each state (survivor) in the trellis. We show, however, that these implicit phase estimators still "lock" onto the carrier phase, in a manner similar to that of a coherent demodulator. Furthermore, because of the extremely low signal-to-noise ratio (SNR) in FEC applications, the phase lock sometimes slips by 180 degrees, which causes the demodulator output symbols to be inverted. We show that this is a relatively minor problem for a serially concatenated convolutional code (SCCC), mainly because the SCCC system uses *differential encoding* and this encoding is immune to 180 degree phase shifts. On the other hand, a low density parity check (LDPC) coded system does *not* use differential encoding. Thus, in LDPC systems the 180 degree phase shift proves to be quite catastrophic.

The solution to this problem is to adjust the "forgetting factor" parameter of the noncoherent detector to a value that yields a tighter phase lock. However, this diminishes the noncoherent nature of the demodulator. Therefore, another way of stating our conclusion is that *coherent demodulators* should be used in FEC applications, especially those using LDPC codes. In the end, while disappointing, this conclusion does not come as a major surprise, because one would not expect noncoherent demodulation to be applicable when differential encoding is not used.

SYSTEM MODEL

The transmitter and receiver models are shown in Figures 1 and 2, respectively. The transmitter consists of a FEC encoder followed by the SOQPSK modulator. The receiver consists of a *soft-output* SOQPSK demodulator followed by a FEC decoder. All of these modules are fully described in our related paper [3], except for the noncoherent demodulator that we now describe.

The *coherent* demodulator uses trellis metrics given by

$$\mathrm{Re}\left\{ e^{-j\hat{\phi}} e^{-j\theta_{n-1}} \int_{nT_s}^{(n+1)T_s} r(t) e^{-j\pi\alpha_n q(t-nT_s)} dt \right\}. \tag{1}$$

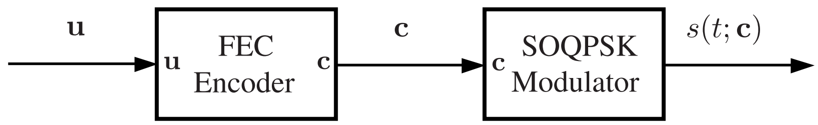

Figure 1: Transmitter Model.

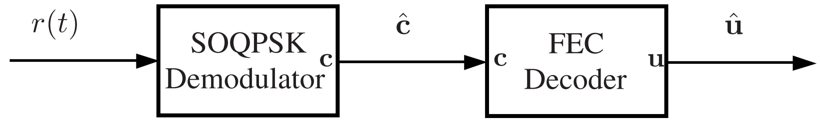

Figure 2: Receiver Model.

The received signal, $r(t)$, contains an unknown phase offset, ϕ. Although the above expression is quite intricate, it consists of three main parts: the matched filtering (MF) operation (i.e. the integral), the phase state correction (i.e. the factor $e^{-j\theta_{n-1}}$), and the correction for the carrier phase offset (i.e. the factor $e^{-j\hat{\phi}}$), where $\hat{\phi}$ is the estimated phase offset. The coherent demodulator explicitly estimates this phase via some means, such as a phase-locked loop (PLL). In the case of perfect phase correction we have $\hat{\phi} = \phi$.

In the case of the noncoherent demodulator, a complex-valued *phase reference*, Q_n, is associated with each state in the trellis at time step n. This *implicit* phase estimate is used to correct the phase of the metrics that branch out from state s in the trellis. Thus the trellis metrics in the noncoherent demodulator become

$$\mathrm{Re}\left\{ Q_{n-1}^{*} e^{-j\theta_{n-1}} \int_{nT_s}^{(n+1)T_s} r(t) e^{-j\pi\alpha_n q(t-nT_s)} dt \right\} \tag{2}$$

where $(\cdot)^{*}$ denotes the complex conjugate. When the soft-ouput Viterbi algorithm (SOVA) determines the surviving branch at each ending state , it then updates implicit phase reference via the operation

$$Q_n = aQ_{n-1} + (1-a)\left(e^{-j\theta_{n-1}} \int_{nT_s}^{(n+1)T_s} r(t) e^{-j\pi\alpha_n q(t-nT_s)} dt \right) \tag{3}$$

where $0 \le a \le 1$ is the *forgetting factor*. The "ingredients" in this update are the past information, scaled by a, and the information from the current time step, scaled by $1-a$. When a is small, the phase reference update places more weight on the current time step (because the factor $1-a$ is large) and places less weight on (i.e. forgets) the past reference $Q_{n-1}(s^{\mathrm{S}}(e^1))$. As $a \to 1$, the phase reference update essentially ignores the information from the current time step and relies almost entirely on the past information. When $a = 1$, the phase reference is no longer updated, and (2) is essentially the same as the coherent case (assuming perfect phase correction).

IMPLICIT PHASE LOCK AND 180 DEGREE PHASE SLIP

Although the noncoherent receiver does not explicitly produce a phase estimate, $\hat{\phi}$, a carrier phase estimate and carrier phase tracking is implicit in Q_n. During the signal acquisition process, the receiver must resolve the time boundary between the transmitted symbols (i.e., symbol timing recovery). Then, for SOQPSK, the receiver imposes its own starting point for even/odd symbol indexes. Once this is

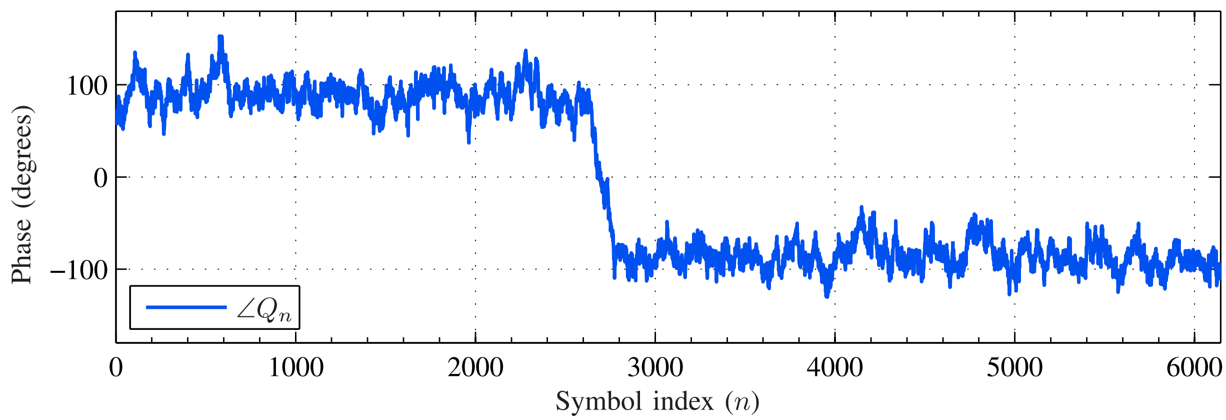

Figure 3: Angle of the noncoherent phase reference over the duration of one code word. In the middle of the code word, the phase reference slips from one lock point to the other.

accomplished, there are two possible phase alignments with SOQPSK—one being 180 degrees apart from the other—and the phase estimate of the receiver will lock onto one of these phase alignments. All of this is true whether the carrier phase is implicitly or explicitly estimated.

We consider the noncoherent demodulator with $a = 7/8$, operating with a rate $2/3$ FEC code and a bit-energy-to-noise ratio in the range of $E_b/N_0 = 2$ to 4 dB. Because of the coding rate, the symbol-energy-to-noise ratio, E_s/N_0, is diminished by $10 \log_{10}(2/3) = -1.76$ dB, and thus we have $E_s/N_0 = 0.24$ to 2.24 dB. The demodulator has the task of synchronizing with respect to the coded *symbols*, and thus the low value of E_s/N_0 makes synchronization more challenging.

Figure 3 shows the angle of the implicit noncoherent phase reference (in degrees) over the duration of one 6144-symbol code word. Initially, the phase reference is locked onto a value of $+90$ degrees. Near the middle of the code word, the noise causes the phase reference to slip to the opposite alignment point of -90 degrees. This causes the bits before and after the phase slip to be inverted with respect to each other. At these low SNRs, the phase slips occur a few percent of the time.

If the FEC code is LDPC, the decoder has no way to deal with the inverted bits—other than to treat them as errors—and thus the decoder is overwhelmed. This is evident by the LDPC BER curve shown in Figure 4. Because the rate of occurrence for the phase slips decreases slowly as E_b/N_0 increases, the BER curve is dominated by errors arising from the phase slips and does not exhibit the desired “waterfall” characteristic that is typical of LDPC codes.

For SCCC, the phase slips do not present much of a problem because they are corrected by the differential decoder with only a minor hiccup (i.e. two errors). Thus, the SCCC BER curve in Figure 4 drops off quickly in the waterfall fashion.

The performance of the two FEC decoders is illustrated in a different way in Figure 5. This figure shows the percentage of phase slips that ultimately lead to decoded bit errors. In the case of LDPC, essentially all phase slips result in decoded bit errors; the only non-error phase slips are those rare instances where the phase slip occurs at the end of the transmitted code word and the number of inverted bits is small enough that the LDPC decoder is not completely overwhelmed.

The situation is very different for SCCC. For $E_b/N_0 < 2.5$ dB, essentially all phase slips result in decoded bit errors. For $2.5 < E_b/N_0 < 3.5$ dB, the percentage drops because the SCCC decoder is better able to correct the two additional errors induced by the phase shift. For $E_b/N_0 > 3.5$ dB, the two

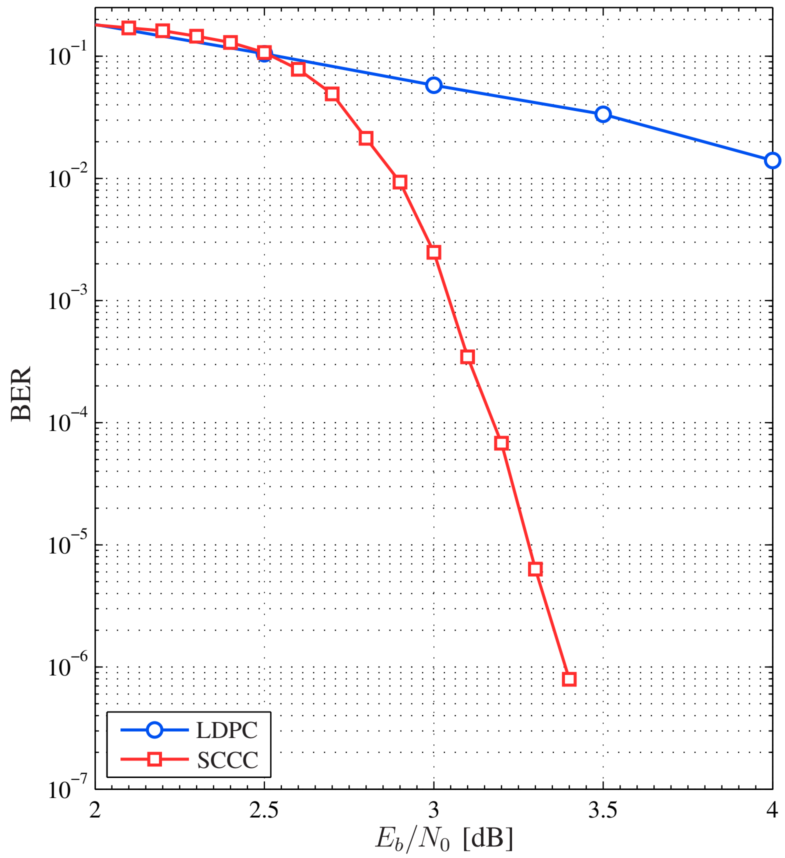

Figure 4: BER results for LDPC and SCCC when paired with a noncoherent demodulator.

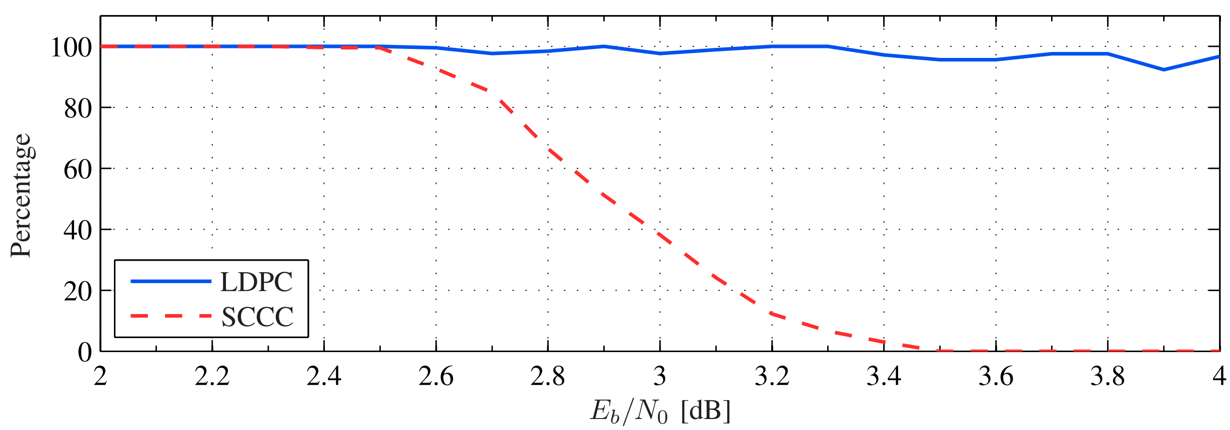

Figure 5: The percentage of phase slips that ultimately lead to decoded bit errors. In the case of LDPC, essentially all phase slips result in decoded bit errors. In the case of SCCC, as E_b/N_0 increases, the SCCC decoder is better able to correct the phase slip errors and at some point the phase slips no longer contribute to the BER.

additional errors induced by the phase shift are always corrected by the decoder and the phase shifts no longer contribute to the BER.

DISCUSSION AND CONCLUSIONS

In the examples above, a value of $a = 7/8$ was used for the forgetting factor. Selecting a value even closer to 1.0 will reduce the rate of phase slips, but at the cost of making the demodulator less able to adapt (i.e. slower to acquire the signal). This is, in fact, the same tradeoff that exists in a PLL-based phase tracking system: a narrower loop bandwidth results in a tighter lock at the expense of a longer acquisition time. Because the noncoherent phase reference requires implementation complexity [in the form of multiplications in (2) and the update in (3)] and because the noncoherent characteristic is minimal due to $a \to 1$, our concluding recommendation is that noncoherent demodulators should not be used in FEC applications, especially those using LDPC codes. Our related paper [3] shows that simple-yet-robust symbol-by-symbol (SxS) demodulators yield strong BER performance at a fraction of the complexity of trellis-based demodulators. Thus, there are other more promising technologies available for FEC applications.

ACKNOWLEDGEMENT

This project is funded by the Test Resource Management Center (TRMC) Test and Evaluation/Science & Technology (T&E/S&T) Program through the U.S. Army Program Executive Office for Simulation, Training and Instrumentation (PEO STRI) under Contract No. W900KK-11-C-0031.

REFERENCES

[1] R. Schober and W. H. Gerstacker, "Metric for noncoherent sequence estimation," *Electron. Lett.*, vol. 35, pp. 2178–2179, Dec. 1999.

[2] A. Syed and E. Perrins, "Comparison of noncoherent detectors for SOQPSK and GMSK in phase noise channels," in *Proc. Int. Telemetering Conf.*, (Las Vegas, NV), Oct. 2007.

[3] E. Perrins, "System design for FEC in aeronautical telemetry," in *Proc. Int. Telemetering Conf.*, (San Diego, CA), Oct. 2012.