# A SYSTEM-LEVEL DESCRIPTION OF A SOQPSK-TG DEMODULATOR FOR FEC APPLICATIONS

Gino Rea
Department of Electrical Engineering & Computer Science
University of Kansas
Lawrence, KS 66045
ginorea@ittc.ku.edu

Faculty Advisor:
Erik Perrins

## ABSTRACT

In this paper we present a system-level description of a demodulator for *shaped offset quadrature phase shift keying*, telemetry group version (SOQPSK-TG) for use in *forward error correction* (FEC) applications. We describe the system in block-diagram form and provide implementation details for data sequence detection, symbol timing synchronization, carrier phase synchronization, and block recovery. This decision-directed demodulator is based on maximum likelihood principles, and is efficiently implemented by the *soft output Viterbi algorithm* (SOVA). We also provide results of the demodulator's performance in the additive white Gaussian noise channel, based on the observed bit error rate at different signal-to-noise ratio levels.

## INTRODUCTION

The telecommunications and information industry has been experiencing a migration to *forward error correction* (FEC) systems in recent years. FEC applications, like *low-density parity-check* (LDPC) and turbo codes, allow the demodulator to detect and correct errors (up to some limit) without the need and, more importantly, the cost of data retransmissions. The introduction of FEC codes is a clear advantage. However, migration to this better technology also represents a challenge because existing demodulators must be enhanced to work with FEC systems.

In this paper, we present a demodulator for *shaped offset quadrature phase shift keying*, telemetry group version (SOQPSK-TG) for use in FEC applications. SOQPSK is categorized as a highly bandwidth efficient *continuous phase modulation* (CPM) scheme, which can be demodulated using a CPM-based detector. The proposed decision-directed detector uses maximum likelihood principles, implemented by the *soft output Viterbi algorithm* (SOVA), to estimate the unknown quantities of information sequence, symbol timing and carrier phase. This detector is attractive for its low complexity and strong performance, and is based on the work presented in [1].

Unlike previously published works, in this paper we describe how a number of separately published technologies can be assembled into a fully synchronized demodulator. The main contributions of this paper are to discuss the implementation details of data sequence detection via the SOVA, symbol timing synchronization, carrier phase synchronization, and block recovery. Also, the demodulator's performance is evaluated based on the measured bit error rate (BER) at different signal-to-noise ratio (SNR) levels. This study should be considered as a simulation of the final model that yet needs to be integrated, developed and tested in hardware.

In the following, the decision-directed maximum likelihood detector implementation is described. The sections of sequence detection, symbol timing synchronization, carrier phase synchronization and block recovery are explained in detail. Finally, the simulation results are summarized and conclusions are presented.

## THE DECISION-DIRECTED MAXIMUM LIKELIHOOD DETECTOR

SOQPSK can be considered as a highly bandwidth efficient form of CPM with unique characteristics that represent considerable advantages at the transmitter side [2]. The transmitted SOQPSK signal, which is described in detail in, e.g. [1], has the following form

$$s(t;\boldsymbol{\alpha}) = \sqrt{\frac{E_s}{T_s}} e^{j\psi(t;\boldsymbol{\alpha})}$$

where $E_s$ is the symbol energy, $T_s$ is the symbol duration and $\psi(t;\boldsymbol{\alpha})$ is the phase of the signal.

We now consider a signaling waveform sent through additive white Gaussian noise, the AWGN channel. The received signal can be expressed as

$$r(t) = \sqrt{\frac{E_s}{T_s}} e^{j\psi(t-\tau;\boldsymbol{\alpha})} e^{j\phi} + w(t) \tag{1}$$

where $w(t)$ is a zero-mean complex-valued AWGN process with one-sided power spectral density $N_0$. This representation shows that the data symbols $\alpha$, symbol timing $\tau$, and carrier phase $\phi$, are unknown to the demodulator and must be recovered. The data symbols $\alpha$ are demodulated by applying maximum likelihood sequence detection (MLSD), which is efficiently implemented via the soft output Viterbi algorithm (SOVA). The soft output provided by the SOVA is required to drive the LDPC decoder. On the other hand, symbol timing synchronization and carrier phase recovery are achieved via a timing error detector (TED) and a phase error detector (PED), respectively, both based on maximum likelihood (ML) principles. The sections of sequence detection and timing recovery presented below are based on the decision-directed maximum likelihood timing error detector (DD-MLTED) described in [1]. In what follows, we refer to the estimated and hypothesized values of a generic quantity $a$ as $\hat{a}$ and $\tilde{a}$ respectively. Also, $\hat{a}$ and $\tilde{a}$ can assume the same value of $a$ itself.

### A. *Maximum Likelihood Sequence Detection*

CPM signals are optimally demodulated by applying MLSD [2, Ch. 7]. SOQPSK is categorized as one form of CPM; therefore, this method can be employed to recover the data symbols $\alpha$ (and consequently, the information sequence $\boldsymbol{u}$).

The ML detector first assumes that the carrier phase $\phi$ and the symbol timing $\tau$ are known [1]. It was shown in [3] that the likelihood function for (1), given a hypothetical information sequence $\tilde{\boldsymbol{u}}$ over the interval $0 \leq t \leq L_0 T$ is

$$\Lambda(\boldsymbol{r}|\tilde{\boldsymbol{u}}) = \exp\left\{ \frac{1}{N_0} \sqrt{\frac{E_s}{T_s}} \sum_{k=0}^{L_0-1} \mathrm{Re}\left\{ e^{-j\phi} \boldsymbol{Z}_k(\tilde{\mathbf{c}}_k, \tilde{\alpha}_k, \tau) e^{-j\tilde{\theta}_k} \right\} \right\} \tag{2}$$

where $\boldsymbol{Z}_k(\cdot)$ are the *matched filter* (MF) outputs. The variables $\tilde{\mathbf{c}}_k$, $\tilde{\alpha}_k$ and $\tilde{\theta}_k$ correspond to hypothetical values obtained from $\tilde{\boldsymbol{u}}$, which are used within the SOVA.

The TG version of SOQPSK is a waveform with partial response $(L > 1)$. An optimal detector for this modulation type requires 512 phase states and MF outputs [1], which is highly complex. A near-optimum approximation called pulse truncation (PT) is applied in, e.g. [4, 5], which allows (2) to be implemented with only the outputs of three MFs (instead of 512) with a loss in performance of $0.2$ dB [3].

The SOVA finds the data symbols sequence $\tilde{\alpha}$ that maximizes (2) and outputs the estimated sequence $\hat{\alpha}$. Implementation details of the SOVA are given in, e.g. [6], [7].

### B. *Maximum Likelihood Timing Synchronization*

Timing synchronization ensures that sampling of the received signal is executed at the correct instance. In general, a clock signal is not transmitted for the purpose of timing synchronization because of spectrum constraints. Therefore, the clock must be recovered from the noisy received waveforms that carry the data [8].

In order to recover the symbol timing $\tau$, the ML detector assumes that the data symbols sequence $\alpha$ and the carrier phase $\phi$ are known. It was shown in [9] that the likelihood function for (1), given a hypothetical timing value $\tilde{\tau}$ over the interval $0 \leq t \leq L_0 T$ is

$$\Lambda(\boldsymbol{r}|\tilde{\tau}) = \exp\left\{ \frac{1}{N_0}\sqrt{\frac{E_s}{T_s}} \sum_{k=0}^{L_0-1} \mathrm{Re}\left\{ e^{-j\phi} \boldsymbol{Z}_k(\mathbf{c}_k, \alpha_k, \tilde{\tau}) e^{-j\theta_k} \right\} \right\}. \tag{3}$$

The ML estimate $\tilde{\tau}$ is the value of $\tau$ that maximizes the logarithm of (3), the log-likelihood function. In order to find $\tilde{\tau}$, we first need to take the partial derivative of the log-likelihood function.

$$\frac{\partial}{\partial \tilde{\tau}} \log(\Lambda(\boldsymbol{r}|\tilde{\tau})) = \sum_{k=0}^{L_0-1} \mathrm{Re}\left\{ e^{-j\phi} \boldsymbol{Y}_k(\mathbf{c}_k, \alpha_k, \tilde{\tau}) e^{-j\theta_k} \right\} \tag{4}$$

where $\boldsymbol{Y}_k$ is the partial derivative of the MF bank outputs $\boldsymbol{Z}_k$ with respect to $\tilde{\tau}$. The ML estimate $\tilde{\tau}$ is the value of $\tau$ that forces (4) to zero.

Previously, it was assumed that $\alpha$ and $\phi$ were known to the demodulator. However, the actual data sequence and carrier phase are yet unknown. Therefore, two close approximations are used to substitute these values. $\alpha$ is replaced by the estimated decisions $\hat{\alpha}$ within the SOVA, and $\phi$, by the most recent phase estimate from the PED. Due to these assumptions, $\tilde{\tau}$ is said to be computed in an iterative and adaptive way.

The output of the TED is a timing error signal, which is a function of the MF outputs and the estimated data symbols $\hat{\alpha}$. The timing error signal is defined as [9]

$$e_\tau[k-D] \triangleq \mathrm{Re}\left\{ e^{-j\hat{\phi}[k-D]} \boldsymbol{Y}_{k-D}(\hat{\mathbf{c}}_{k-D}, \hat{\alpha}_{k-D}, \hat{\tau}[k-D]) e^{-j\hat{\theta}_{k-D}} \right\} \tag{5}$$

where $D$ represents how far we traced back along the SOVA trellis to obtain the tentative decisions for computing the error. The vectors $\hat{\mathbf{c}}_{k-D}$, $\hat{\alpha}_{k-D}$, and $\hat{\theta}_{k-D}$ are taken from the path history of the best survivor in the SOVA. This timing error signal is then fed to a PLL that produces the next estimate of the symbol timing $\tilde{\tau}[k-D]$.

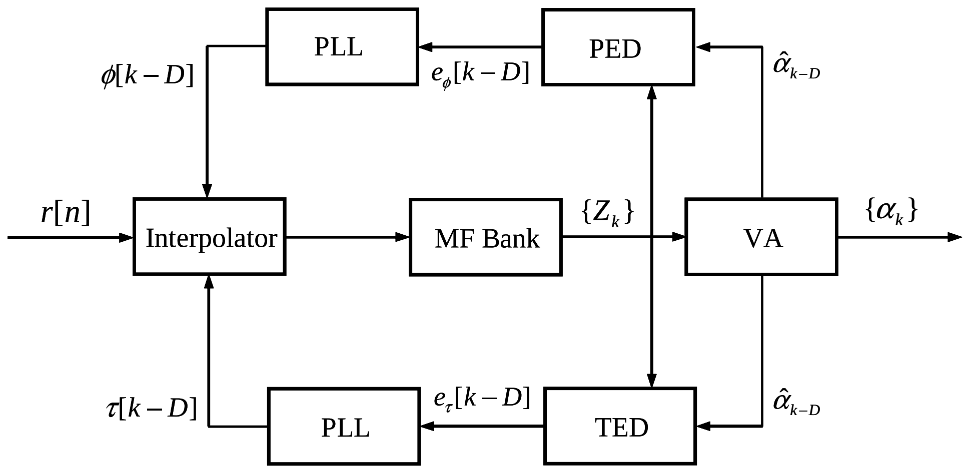


Figure 1: Discrete-time implementation of a decision-directed phase and timing recovery system for SOQPSK.

A discrete-time implementation of the DD-MLTED is shown in block diagram form in the bottom of Figure 1. An interpolator is used to sample the discrete-time received signal $r[n]$ based on the most recent timing estimate $\tilde{\tau}[k-D]$. In addition, a *late* and *early* samples are also taken. The difference between the late and early samples is used to approximate the derivative $\boldsymbol{Y}_k$ as described in, e.g. [9]. All three samples are fed to the MF bank, whose output $\boldsymbol{Z}_k$ is used to calculate and update the branch metrics within the SOVA. Then, the TED provides a timing error signal, $e_\tau[k-D]$, which is used by the PLL to generate the next timing estimate $\tilde{\tau}[k-D]$.

### C. *Maximum Likelihood Phase Synchronization*

Phase synchronization is the process of forcing the local oscillators in the detector to be in both phase and frequency with the carrier oscillators [8]. The implementation of the phase error detector (PED) is similar to that of the TED. First, the detector temporarily assumes that the symbol timing $\tau$ and the data symbols sequence $\alpha$ are known. The likelihood function for (1) given a hypothetical phase value $\tilde{\phi}$ over the interval $0 \le t \le L_0 T$ is

$$\Lambda(\boldsymbol{r}|\tilde{\phi}) = \exp\left\{ \frac{1}{N_0}\sqrt{\frac{E_s}{T_s}} \sum_{k=0}^{L_0-1} \mathrm{Re}\left\{ e^{-j\tilde{\phi}} \boldsymbol{Z}_k(\mathbf{c}_k, \alpha_k, \tau) e^{-j\theta_k} \right\} \right\}. \tag{6}$$

The ML estimate $\tilde{\phi}$ is the value of $\phi$ that maximizes the logarithm of (6), the log-likelihood function. In order to find $\tilde{\phi}$, we first need to take the partial derivative of the log-likelihood function.

$$\frac{\partial}{\partial \tilde{\phi}} \log(\Lambda(\boldsymbol{r}|\tilde{\phi})) = \sum_{k=0}^{L_0-1} \mathrm{Im}\left\{ -je^{-j\phi} \boldsymbol{Z}_k(\mathbf{c}_k, \alpha_k, \tau) e^{-j\theta_k} \right\} \tag{7}$$

where the ML estimate $\tilde{\phi}$ is the value of $\phi$ that forces (7) to zero.

Contrary to timing synchronization, in this case, the imaginary part of the MF outputs is forced to zero. This is because of the multiplication of the $-j$ term, which results from the derivative of $e^{-j\phi}$, with the real and imaginary arguments of $\boldsymbol{Z}_k$.

Previously, the detector assumed that $\alpha$ and $\tau$ were known to the demodulator. However, the actual data sequence and symbol timing are yet unknown. Therefore, two approximations are used one more time. $\alpha$ is replaced by the estimated decisions $\hat{\alpha}$ within the SOVA, and $\tau$, by the most recent symbol timing estimate. Using the supposedly time-synchronized tentative decisions within the SOVA, the PED generates the following phase error signal

$$e_\phi[k-D] \triangleq \mathrm{Im}\left\{-je^{-j\hat{\phi}[k-D]}\boldsymbol{Z}_{k-D}(\hat{\mathbf{c}}_{k-D}, \hat{\alpha}_{k-D}, \hat{\tau}[k-D])e^{-j\hat{\theta}_{k-D}}\right\} \tag{8}$$

which is then fed to a PLL that produces the next estimate of the carrier phase $\tilde{\phi}[k-D]$.

The top of Figure 1 shows a discrete-time implementation of the decision-directed ML phase error detector (DD-MLPED). The interpolator samples the received signal $r[n]$ based on the most recent timing estimate $\tilde{\tau}[k-D]$. The time-synchronized samples are then rotated by the output of the VCO to remove any residual phase error. All three time and phase synchronized samples are fed to the bank of MFs, whose output $\boldsymbol{Z}_k$ is used to calculate and update the branch metrics within the SOVA. Then, the PED outputs a phase error signal, $e_\phi[k-D]$, which is used by the PLL to generate the next carrier phase estimate $\tilde{\phi}[k-D]$.

### D. *Block Synchronization*

Block synchronization is an operation that achieves two purposes. The first one is to determine the beginning of a block in the received data stream. And the second, is to resolve the phase ambiguity that can occur after the carrier phase is locked. Block synchronization is achieved in part by inserting a pattern of known symbols (or "attached synch marker" - ASM) at the front of the data symbols. Considering this factor, the structure of each block or frame consists of some junk bits at the front, then the ASM followed by the codeword, and finally some more junk bits at the end.

Similarly to QPSK modulation, SOQPSK exhibits a $90°$ phase ambiguity. In other words, the PED PLL can lock in four different ways with the carrier: it can lock in phase with the carrier, $90°$ out of phase with the carrier, $180°$ out of phase with the carrier, or $270°$ out of phase with the carrier [8]. After carrier phase lock, the detector searches for the four possible ASM rotations using a correlation operation, and corrects phase ambiguity by inverting the appropriate bits according to the detected ASM rotation.

## SIMULATION

The proposed decision-directed ML detector was simulated and tested in software via the Matlab application. The purpose of this simulation was to evaluate the demodulator's performance based on the observed BER at different SNR levels. We set the SNR test interval from $1.0$ dB to $2.3$ dB, and simulated each data point until a minimum of $1000$ bit errors were obtained. Timing synchronization employs a first order PLL with a user-specified loop bandwidth of $1/1024$. Similarly, phase synchronization relies on a first order PLL with the same loop bandwidth value. After the time and phase-synchronized blocks are extracted from the received data stream, the LDPC decoder is allowed a maximum of 200 iterations to detect and correct any errors in the decoded information sequence. The demodulator's performance results are shown in Figure 2.

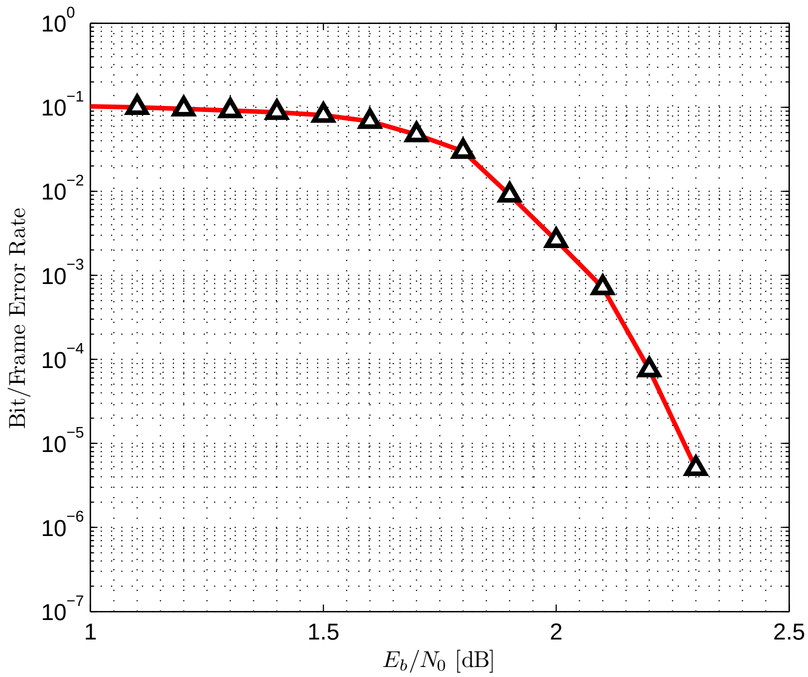


Figure 2: Performance results of SOQPSK-TG demodulator with LDPC decoding scheme

## CONCLUSIONS

In this paper, we have described how a number of separately published technologies can be assembled into a fully synchronized demodulator for SOQPSK compatible with FEC applications. We discussed the implementation details of data sequence detection via the SOVA, symbol timing synchronization and carrier phase synchronization based on ML principles, and block recovery by means of an ASM. Then, we simulated the demodulator in Matlab and provided performance results in the AWGN channel. With a loop bandwidth of $1/1024$, the demodulator's performance is essentially optimal. A larger loop bandwidth value should produce results closer to the perfect timing case. We have verified this with reasonably large values of the loop bandwidth, and it was seen that the detector provides accurate results. Because of scope limitations, these curves are not shown here. As a final remark, the presented timing and phase recovery schemes are compatible with recently introduced optimal CPM-based detectors for SOQPSK, which opens the possibility for potential use in a wide range of applications in the telemetry industry.

## ACKNOWLEDGEMENT

The authors would like to thank the Test Resource Management Center (TRMC) Test and Evaluation/Science and Technology (T&E/S&T) Program for their support. This work was funded in part by the T&E/S&T Program through the U.S. Army Program Executive Office for Simulation, Training and Instrumentation (PEO STRI), contract number W900KK-09- C-0018 for High-Rate High-Speed Forward Error Correction Architectures for Aeronautical Telemetry (HFEC), and in part by a joint grant from the National Aeronautics and Space Administration and the Kansas Technology Enterprise Corporation, grant number NNX08AV84A.

## REFERENCES


[1] P. Chandran and E. Perrins, “Decision-directed symbol timing recovery for SOQPSK,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, pp. 781–789, Apr. 2009.

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

[3] E. Perrins and M. Rice, “Reduced-complexity approach to iterative detection of SOQPSK,” *IEEE Trans. Commun.*, vol. 55, pp. 1354–1362, Jul. 2007.

[4] T. Aulin, C.-E. Sundberg, and A. Svensson, “Viterbi detectors with reduced complexity for partial response continuous phase modulation,” in *Proc. National Telecommun. Conf., NTC’81*, (New Orleans, LA), pp. A7.6.1–A7.6.7, Nov./Dec. 1981.

[5] A. Svensson, C.-E. Sundberg, and T. Aulin, “A class of reduced-complexity Viterbi detectors for partial response continuous phase modulation,” *IEEE Trans. Commun.*, vol. 32, pp. 1079–1087, Oct. 1984.

[6] M. Fossorier, F. Burkert, S. Lin, and J. Hagenauer, “On the equivalence between SOVA and Max-Log-MAP decodings,” *IEEE Trans. Commun.*, vol. 2, pp. 137–139, May. 1998.

[7] D. Alam and E. Perrins, “Coded SOQPSK-TG using the Soft Output Viterbi Algorithm,” in *Proc. Int. Telemetering Conf.*, (Las Vegas, NV), Oct. 2009.

[8] M. Rice, *Digital Communications: A Discrete-Time Approach*. New York: Prentice Hall, 2009.

[9] M. Morelli, U. Mengali, and G. M. Vitetta, “Joint phase and timing recovery with CPM signals,” *IEEE Trans. Commun.*, vol. 45, pp. 867–876, Jul. 1997.