Improvements to Symbol Signal-to-Noise Ratio Estimation for the Block V Receiver

Meera Srinivasan*, Andre Tkacenko*, and Frank Cheng†

In this paper, we analyze problems arising from the implementation of the Split-Symbol Moments Estimator (SSME) for symbol signal-to-noise ratio (SNR) estimation currently used for the Block V Receiver (BVR). Through computer simulations, it is shown that the problems arise from a bias in the moments forming the SNR estimate which occur whenever the number of baseband samples per symbol at the receiver is not an even integer. To overcome these problems, the use of a sample-based M_2M_4 estimator is proposed which is shown to yield satisfactory symbol SNR (SSNR) estimates regardless of the number of samples per symbol. Given the results of the simulations and the performance of the M_2M_4 estimator, several recommendations are made for addressing the SSNR estimation problems currently present in the BVR.

I. Introduction

In May 2006, cruise testing with the Mars Reconnaissance Orbiter (MRO) resulted in the identification of certain deficiencies in currently implemented Block V Receiver (BVR) algorithms related to estimation of signal and system noise power levels. These issues were summarized in [1]. In particular, the carrier-to-noise ratio (P_c/N_0) and system noise temperature (SNT) estimates are corrupted when there is no subcarrier and when the data power is strong. Furthermore, future bandwidth efficient modulation formats for high data rate telemetry [2] and transpondring will lead to the presence of non-rectangular modulation pulses. Current signal quality estimation techniques used in the BVR were not designed to operate with these signal types.

Table 1 contains a portion of data from BVR testing of the P_c/N_0 and symbol SNR (SSNR) estimators at various data rates and signal levels. The true P_c/N_0 and SSNR values are obtained from measurements taken using a data subcarrier at 360 kHz, and are

*Communications Architectures and Research Section

†Communications Ground Systems Section

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Symbol rate (sym/s)	P_c/N_0 true (dB-Hz)	SSNR true (dB)	SSNR est error (dB)	P_c/N_0 est. error using est. SSNR (dB-Hz)	P_c/N_0 est. error using true SSNR (dB-Hz)
2399	67.0	38.7	11.5	14.0	2.7
4998	67.0	40.7	5.9	4.4	-1.3
9996	67.2	37.1	1.1	0.5	-0.2
40004	67.2	31.1	-0.1	-0.2	1.0
100025	66.9	26.8	-0.1	1.6	1.6
2399	62.0	43.6	16.0	13.0	-1.7
4998	62.0	40.5	5.4	4.1	-1.1
9996	62.2	39.3	2.7	1.3	-2.5
40004	61.9	31.2	-0.2	0.1	0.4
100025	61.8	27.1	-0.1	0.4	0.4

Table 1. BVR hardware test data

compared with estimates of these quantities made under the direct modulation format (no subcarrier). In [3], it was shown that when direct modulation is used, the P_c/N_0 estimate is corrupted by data crosstalk caused by carrier phase tracking errors. In order to correct for this, a statistically-based correction term for the data contribution is applied to the P_c/N_0 estimate that is described in [4]. This correction term requires an estimate of the data symbol SNR; specifically, the corrected carrier-to-noise ratio estimate $\left(\widehat{\frac{P_c}{N_0}}\right)_c$ is given by,

$$\left(\widehat{\frac{P_c}{N_0}}\right)_c = \left(\widehat{\frac{P_c}{N_0}}\right)_{uc} \left(1 + 2B_l \tan^2(\theta_m) \left(1 + 2\widehat{\frac{E_s}{N_0}}\right) / R_{sym}\right) \tag{1}$$

where $\left(\widehat{\frac{P_c}{N_0}}\right)_{uc}$ is the uncorrected P_c/N_0 estimate, B_l is the carrier loop tracking bandwidth, θ_m is the modulation index, $\widehat{\frac{E_s}{N_0}}$ is the SSNR estimate, and R_{sym} is the symbol rate. The fifth column of Table 1 shows the P_c/N_0 estimation error when the corrected estimate of (1) uses the SSNR estimate of the BVR. We observe that at a P_c/N_0 of approximately 67 dB, the estimation error is 14 dB at the lowest symbol rate and decreases as symbol rate increases. We also note the SSNR estimator itself is in error by 11.5 dB at the lowest symbol rate (fourth column), and that when the true SSNR value is used in (1), the errors are not as large, as shown by the sixth column of Table 1. This leads us to examine the SSNR estimator as a possible source of the P_c/N_0 estimator problems.

In order to emphasize the significance of symbol SNR estimation, we show in Figure 1 the relationship between certain signal properties, receiver components, and estimators. Non-rectangular modulation, varying symbol rates, and the possible presence of a ranging signal all affect the output of the receiver filter, which in turn impact the performance of the symbol SNR estimator, either through the mechanism of intersymbol interference or spectral overlap. After examining the various factors shown in this diagram, it becomes

clear that the symbol SNR estimator is a nexus of error dependency. Figures 2(a) and 2(b) show how errors in SNR estimation translate quantitatively to errors in P_c/N_0 and system noise temperature estimation. We therefore focus our attention on analyzing the SSNR estimator in the remainder of this paper.

In Sec. II, we review the signal model received at the BVR, including the simulation implementation of the analog-to-digital converter (ADC) module, the digital down converter (DDC), and the receiver matched filter. In Sec. III, the formulation of the SSME is reviewed, followed by simulation results for rectangular pulses used in the BVR and square-root raised cosine (SRRC) pulses that may be used in future applications. The second & fourth order moments estimator (M_2M_4) is detailed in Sec. IV, including a sample-based version for rectangular pulses. The SSME, along with the symbol and sample based M_2M_4 estimators are compared in Sec. V, specifically in regards to the symbol SNR estimation for the BVR. Finally, concluding remarks are made in Sec. VI.

A. Notations

All notations used are as in [5]. In particular, continuous-time (analog) and discrete-time (digital) normalized frequencies are denoted as F and f, respectively. Parentheses and square brackets are respectively used for continuous-time and discrete-time function arguments. For example, $x(t)$ would denote a continuous-time function for $t \in \mathbb{R}$, whereas $y[n]$ would denote a discrete-time function for $n \in \mathbb{Z}$.

Regarding random variables, the notation $a \sim \mathcal{N}(\mu, \sigma^2)$ means that the random variable a has a normal or Gaussian distribution [6] with mean μ and variance σ^2.

II. Review of the Block V Receiver Signal Model

Prior to analyzing the performance of the SSME for the BVR, it is insightful to review the telemetry signal model assumed. From [7], the analog intermediate frequency (IF) signal $r_{IF}(t)$ appearing within the BVR is as shown below.

$$r_{IF}(t) = \sqrt{2P}\sin\left[2\pi F_{IF}t + \Delta D(t) + \theta_c\right] + \eta(t) \tag{2}$$

Here, P is the average signal power, F_{IF} is the IF frequency (which is 200 MHz here), Δ is the modulation index, $D(t)$ is the subcarrier modulated data pulse, θ_c is the carrier phase, and $\eta(t)$ is a narrowband noise process. For the Block V, the signal $D(t)$ is given by the following expression.

$$D(t) = d(t)\,\mathtt{Sin}\left(2\pi F_{sc}t + \theta_{sc}\right) \tag{3}$$

where $d(t)$ is the data pulse,

$$d(t) = \sum_{n=-\infty}^{\infty} d[n]\,p_{TX}(t - nT_d) \tag{4}$$

with $d[n]$ denoting the sequence of data symbols (assumed to come from a binary phase-shift keying (BPSK) constellation), $p_{TX}(t)$ representing the transmit pulse shape,

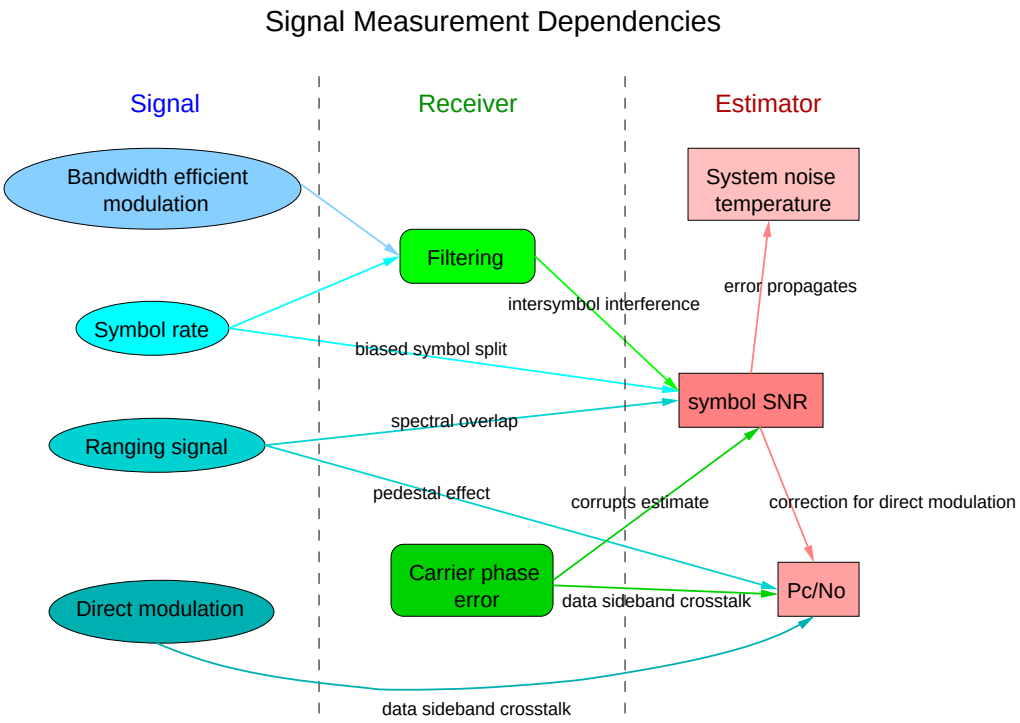


Figure 1. Error dependencies in the Block V Receiver.

and T_d denoting the data symbol time interval. For the Block V, the transmit pulse $p_{TX}(t)$ is a non-return-to-zero (NRZ) rectangular pulse shape with the following form.

$$p_{TX}(t) = \begin{cases} 1\,, & 0 \le t < T_d \\ 0\,, & \text{otherwise} \end{cases} \tag{5}$$

Here, the subcarrier modulation function $\texttt{Sin}(x)$ is defined below as follows.

$$\texttt{Sin}(x) = \begin{cases} \text{sgn}(\sin(x))\,, & \text{for a square wave subcarrier} \\ \sin(x)\,, & \text{for a sine wave subcarrier} \end{cases}$$

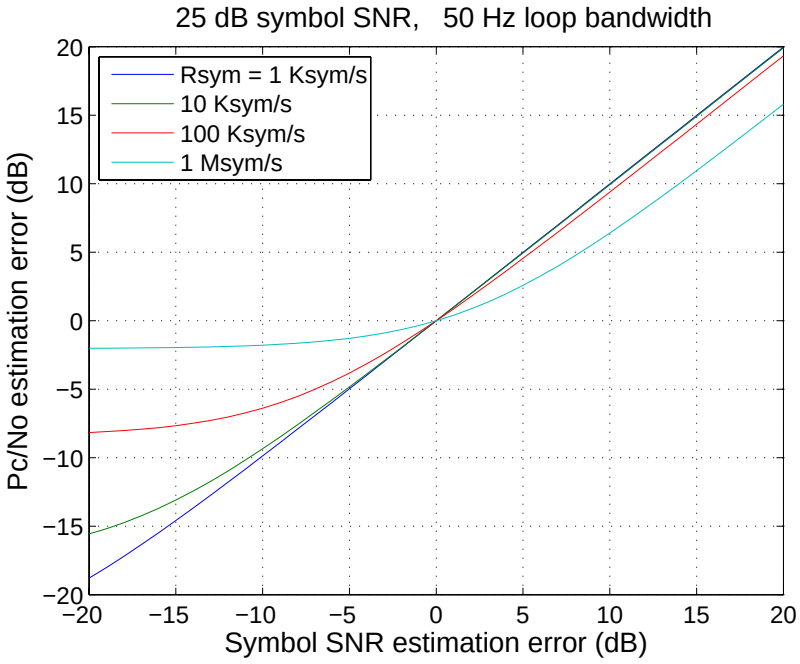
The noise process $\eta(t)$ is assumed to have the following decomposition [7].

$$\eta(t) = \sqrt{2}\eta_c(t)\cos\left(2\pi F_{IF}t + \theta_c\right) - \sqrt{2}\eta_s(t)\sin\left(2\pi F_{IF}t + \theta_c\right) \tag{6}$$

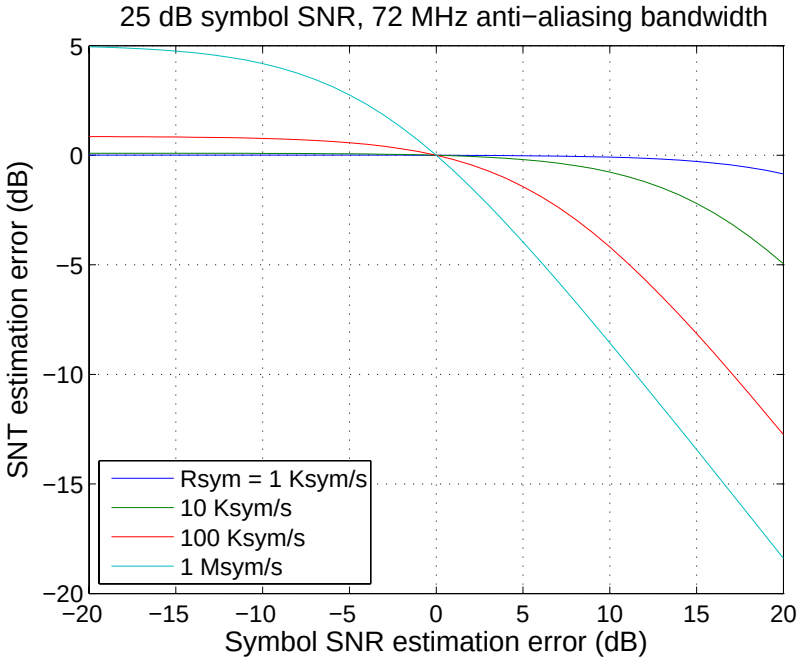
Here, $\eta_c(t)$ and $\eta_s(t)$ are statistically uncorrelated zero-mean bandlimited Gaussian noise processes with power spectral density $N_0/2$ W/Hz and one-sided bandwidth $F_s/2$, where F_s denotes the sampling rate of the complex baseband received signal (which is 80 MHz here). Specifically, this implies that we have [6],

$$S_{\eta_c\eta_c}(j2\pi F) = S_{\eta_s\eta_s}(j2\pi F) = \begin{cases} \frac{N_0}{2}\,, & -\frac{F_s}{2} \le F < \frac{F_s}{2} \\ 0\,, & \text{otherwise} \end{cases} \tag{7}$$

where $S_{\eta_c\eta_c}(j2\pi F)$ and $S_{\eta_s\eta_s}(j2\pi F)$ denote the power spectral densities (PSDs) of $\eta_c(t)$ and $\eta_s(t)$, respectively.



(a)



(b)

Figure 2. (a) P_c/N_0 estimation error vs. SSNR estimation error (b) SNT estimation error vs. SSNR estimation error. (Here, the estimation error in dB is defined to be the difference between the actual value in dB minus the estimate in dB.)

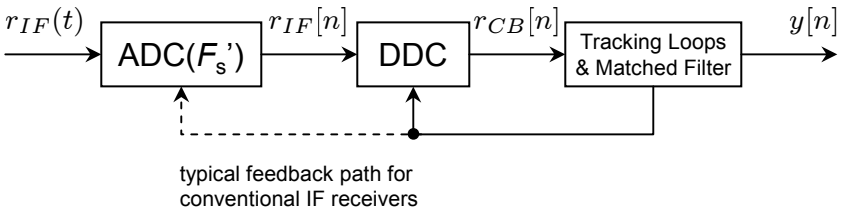


Figure 3. Block diagram of the IF portion of the Block V Receiver.

From (2), (6), and (7), it can be seen that the sample SNR is given by the following expression.

$$\text{SNR} = \frac{P}{N_0 F_s} \tag{8}$$

On the other hand, from (4) and (5), the symbol SNR (SSNR) is given by the following expression.

$$\text{SSNR} = \frac{E_s}{N_0} = \frac{PT_d}{N_0} \tag{9}$$

Here, E_s denotes the symbol energy. Combining (8) and (9), it can be seen that the sample and symbol SNRs are related as follows.

$$\text{SSNR} = (F_s T_d) \cdot \text{SNR} = \frac{F_s}{\mathcal{R}_d} \cdot \text{SNR} \tag{10}$$

where $\mathcal{R}_d \triangleq 1/T_d$ denotes the data or symbol rate. Note that the quantity $F_s/\mathcal{R}_d$ represents the number of samples per symbol at the receiver end.

A rudimentary block diagram of the IF portion of the BVR is shown in Fig. 3. Here, the real IF analog signal $r_{IF}(t)$ is sampled by an analog-to-digital converter (ADC) to produce the real IF digital signal $r_{IF}[n]$. The sampling rate of the ADC is twice that of the complex baseband signal, i.e., $F_s' = 2F_s$ (so that $F_s' = 160$ MHz here). Afterwards, the IF signal $r_{IF}[n]$ is processed by a digital down converter (DDC) to produce the digital complex baseband (CB) signal $r_{CB}[n]$. Finally, the baseband signal $r_{CB}[n]$ is presented to a series of tracking loops and a matched filter to produce the output sequence $y[n]$.

To account for symbol timing and carrier frequency corrections, information from the tracking loops is typically fed back to the ADC and DDC in conventional IF receivers as shown from the feedback paths in Fig. 3. In particular, fractional symbol timing information is often fed back to the ADC, whereas carrier frequency updates are typically fed back to the DDC. However, for the BVR specifically, symbol synchronization is carried out internally using the output of the receiver matched filter, and so the only feedback path relevant to the BVR in Fig. 3 is that going to the DDC in the form of carrier frequency estimates. Regardless, in the following, we will assume that the analog IF signal input to the system of Fig. 3 does not require such corrections (meaning that $r_{IF}(t)$ is of the form given in (2)), and so no feedback paths exist and $y[n]$ is simply the output of $r_{CB}[n]$ from the receiver matched filter.

In addition to this, we will also assume that the data rate $\mathcal{R}_d$ is a rational multiple of the CB sampling rate F_s. Specifically, we will assume from here on that $\mathcal{R}_d$ is related to F_s as

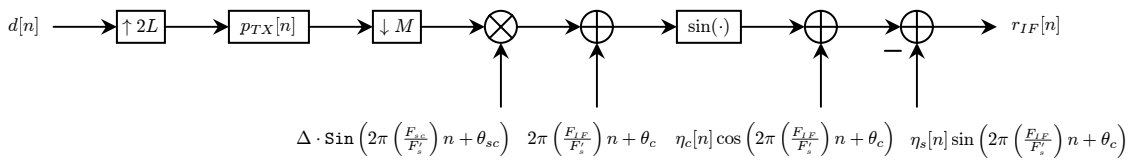


Figure 4. Digital IF signal generation block diagram.

follows.

$$\mathcal{R}_d = \left(\frac{M}{L}\right) F_s \tag{11}$$

where L and M are positive integers with $L \geq M$. Note that from (10), (11) implies that we have the following relation between sample and symbol SNRs.

$$\text{SSNR} = \left(\frac{L}{M}\right) \cdot \text{SNR} \tag{12}$$

Here, the quantity L/M denotes the number of samples per symbol.

A. Simulation Implementation of the ADC Module

From Fig. 3, the real digital IF signal $r_{IF}[n]$ is given by $r_{IF}[n] \triangleq r_{IF}(nT_s') = r_{IF}(n/F_s')$, where $T_s' \triangleq 1/F_s'$ denotes the IF sampling time interval. For the special case in which the data rate $\mathcal{R}_d$ is a rational factor of the underlying CB sampling rate $F_s = F_s'/2$ as in (11), the IF signal $r_{IF}[n]$ can be easily generated in a simulation environment using the block diagram shown in Fig. 4. The data stream $d[n]$ is first processed by a fractional decimation filter system to produce the filtered pulse stream. Here, from (5), the digital transmit pulse shaping filter $p_{TX}[n]$ is given by the following expression.

$$p_{TX}[n] = \begin{cases} 1\,, & 0 \leq n \leq 2L-1 \\ 0\,, & \text{otherwise} \end{cases}$$

A subcarrier is then mixed with the resulting signal, the phase contribution from the IF modulation is added, and then the sine of the result is computed to yield the pure signal component of $r_{IF}[n]$. Note that there is no need to scale the signal component by $\sqrt{2P}$ as in (2), since this factor can be compensated for by cleverly scaling the noise component of the signal. Equivalently, without loss of generality, we can assume that $P = 1/2$ so that the scale factor for the signal component is unity.

After the signal component is computed, the noise component of $r_{IF}[n]$ is added as shown in Fig. 4. Here, the sequences $\eta_c[n]$ and $\eta_s[n]$ are uncorrelated real zero-mean white Gaussian processes. The variances of these processes ($\sigma_{\eta_c}^2$ and $\sigma_{\eta_s}^2$, respectively) are equal and given by the following expression.

$$\sigma_{\eta_c}^2 = \sigma_{\eta_s}^2 = \frac{1}{2}\left[\frac{1}{\text{SNR}}\right] = \frac{1}{2}\left[\frac{1}{\text{SSNR}} \cdot \left(\frac{L}{M}\right)\right] \tag{13}$$

Here, we used (12) in (13). From (13), it follows that $\eta_c[n], \eta_s[n] \sim \mathcal{N}\left(0, \frac{1}{2}\left[\frac{1}{\text{SSNR}} \cdot \left(\frac{L}{M}\right)\right]\right)$.

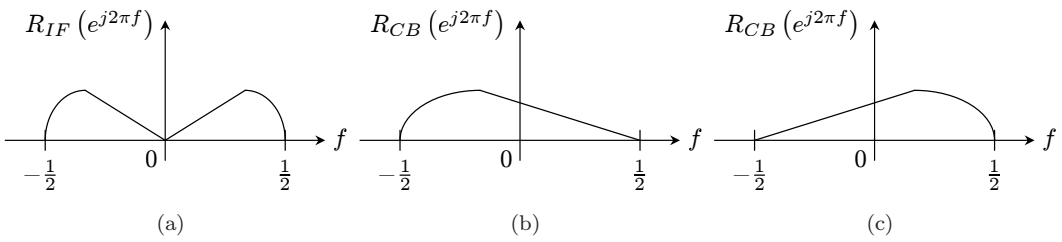


Figure 5. Illustrating the purpose of the DDC: (a) spectrum of IF input signal, (b) spectrum of left-shifted CB output signal, and (c) spectrum of right-shifted CB output signal.

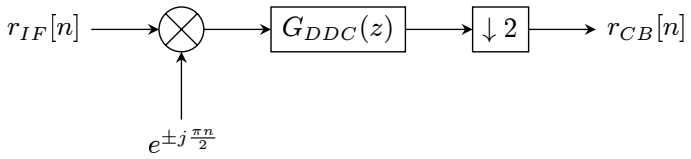


Figure 6. DDC system implementation.

B. Simulation Implementation of the DDC Module

The purpose of the DDC is to convert the real IF digital signal $r_{IF}[n]$ (corresponding to a sampling rate of F_s') to a complex baseband (CB) form $r_{CB}[n]$ (corresponding to a sampling rate of $F_s = F_s'/2$). Pictorially, this is illustrated in Fig. 5 for a fictitious input signal $r_{IF}[n]$ with Fourier transform $R_{IF}\left(e^{j2\pi f}\right)$. In Fig. 5(a), the spectrum of $r_{IF}[n]$ is shown, which illustrates the redundancy of the spectrum on account of the fact that $r_{IF}[n]$ is real. The purpose of the DDC is to preserve only the left or right portions of the spectrum of $r_{IF}[n]$, as illustrated in Fig. 5(b) and (c), respectively.

One way to implement the DDC so as to achieve the desired effect from Fig. 5 is to use the system from Fig. 6. First, the spectrum of the signal is shifted by $f = \pm 1/4$. Shifting by $f = 1/4$ will lead to a left-shifted CB output, whereas shifting by $f = -1/4$ will yield a right-shifted CB output. After shifting the spectrum by the desired amount, the signal is filtered by a lowpass filter $G_{DDC}(z)$ to remove the undesired part of the spectrum. Ideally, the frequency response of $G_{DDC}(z)$ is of the following form.

$$G_{DDC}\left(e^{j2\pi f}\right) = \begin{cases} 2\,, & -\frac{1}{4} \le f < \frac{1}{4} \\ 0\,, & \text{otherwise} \end{cases} \tag{14}$$

Here, the gain factor of the filter is arbitrary and was chosen to be 2 in order to preserve the overall energy of the signal. After filtering by $G_{DDC}(z)$, the resulting signal is decimated by 2 in order to remove the redundant information present to yield the CB output signal $r_{CB}[n]$, which exhibits the desired characteristics shown in Fig. 5(b) or (c).

Though the system of Fig. 6 conceptually downconverts the real IF signal to a CB format as desired, it is not the most efficient implementation of the system. This is due to the fact that the decimation stage occurs after the filtering stage [8]. In order to obtain a more

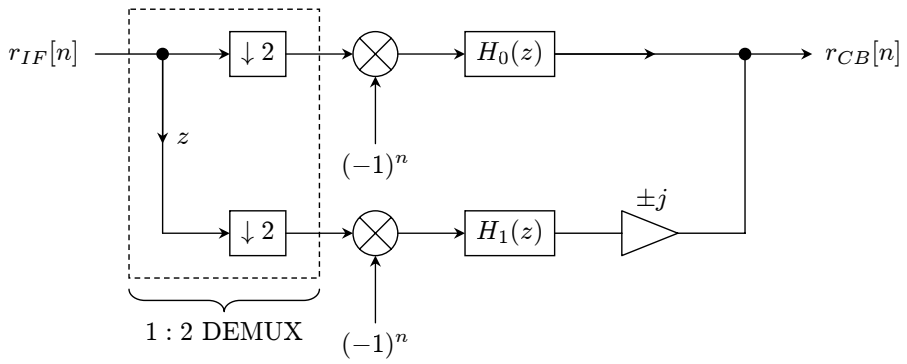


Figure 7. Efficient implementation of the DDC of Fig. 6.

efficient implementation, consider the 2-fold Type 1 polyphase decomposition of the DDC filter $G_{DDC}(z)$, given below as follows [8].

$$G_{DDC}(z) = H_0(z^2) + zH_1(z^2) \tag{15}$$

Through clever manipulation of the *noble identities* [8], the DDC system of Fig. 6 can be efficiently implemented as shown in Fig. 7.

This structure is efficient in many respects. First, the primary stage of the system only consists of a demultiplexing (DEMUX) operation which does not involve any computational resources to implement. Then, each branch of the system is multiplied by $(-1)^n$, which can easily be implemented by alternating the sign bit of each branch. Afterwards, the polyphase filters $H_0(z)$ and $H_1(z)$ must be implemented. As the only requirement of the underlying filter $G_{DDC}(z)$ is that it be lowpass, it follows that both it, and consequently its polyphase components, can be implemented as real coefficient filters. This means that both branches of Fig. 7 remain real up until the very last stage. In fact, in this case, the $\pm j$ multiplier on the bottom branch only serves the purpose of making the bottom branch imaginary. As a result, the real and imaginary components of the desired output $r_{CB}[n]$ can be taken as the outputs of the polyphase filters $H_0(z)$ and $H_1(z)$, respectively (subject to a possible sign change of the imaginary component depending on the sign of the $\pm j$ mutlipler).

Furthermore, if $G_{DDC}(z)$ is designed to be a *half-band* filter [8], meaning that $H_0(z)$ is a constant, then the top branch of Fig. 7 requires no filtering at all. The impetus for designing $G_{DDC}(z)$ as a half-band filter comes from the fact that the ideal lowpass filter response given in (14) is half-band [8] (in fact, $H_0(z) = 1$ here).

For the BVR, the DDC filter $G_{DDC}(z)$ is the linear phase finite impulse response (FIR) [8] half-band filter given below.

$$G_{DDC}(z) = \frac{1}{512}\left(4z^9 - 10z^7 + 21z^5 - 46z^3 + 159z + 256 + 159z^{-1} - 46z^{-3} + 21z^{-5} - 10z^{-7} + 4z^{-9}\right) \tag{16}$$

Technically, $G_{DDC}(z)$ from (16) is implemented causally (i.e., with a delay in order to nullify the positive powers of z) in the Block V. However, for sake of simplicity here, we

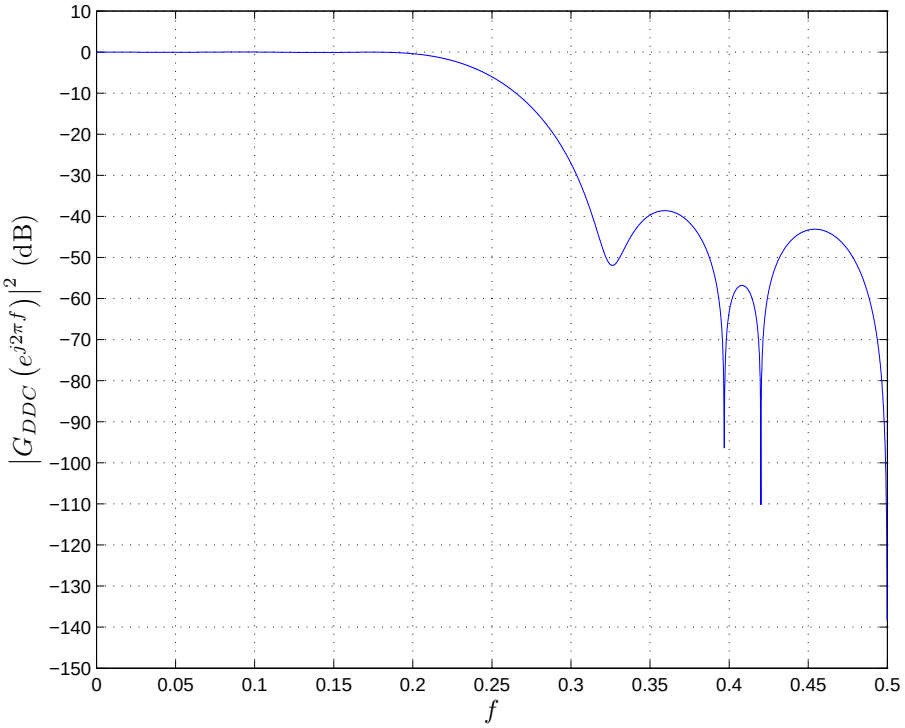


Figure 8. Magnitude response of the DDC filter $G_{DDC}(z)$ used in the Block V Receiver.

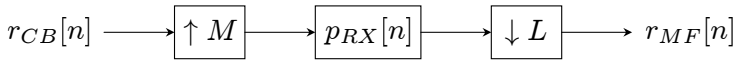


Figure 9. Fractional decimation receiver matched filter system.

chose to present it using the symmetric form as shown in (16). From (16), it follows that the polyphase components $H_0(z)$ and $H_1(z)$ are given below as follows.

$$\begin{aligned} H_0(z) &= \tfrac{1}{2} \\ H_1(z) &= \tfrac{1}{512}\left(4z^4 - 10z^3 + 21z^2 - 46z + 159 + 159z^{-1} - 46z^{-2} + 21z^{-3} - 10z^{-4} + 4z^{-5}\right) \end{aligned}$$

A plot of the magnitude response of the BVR DDC filter $G_{DDC}(z)$ is shown in Fig. 8. As can be seen, it is lowpass, although far from being ideal.

C. Simulation Implementation of the Receiver Matched Filter

Referring to Fig. 3, after the sampling rate of the system has been lowered from F_s' at the output of the ADC to $F_s = F_s'/2$ at the output of the DDC, the resulting CB signal $r_{CB}[n]$ is processed by the receiver matched filter. For the NRZ rectangular analog transmit pulse shape $p_{TX}(t)$ from (5), it follows that the digital matched filter system is a fractional decimation filter system with a rectangular pulse shape as well. In particular, the matched filter system used to obtain the signal $y[n]$ from Fig. 3 is the structure shown in Fig. 9.

Here, the digital receiver pulse shaping filter $p_{RX}[n]$ is given below as follows.

$$p_{RX}[n] = \begin{cases} 1\,, & -(L-1) \le n \le 0 \\ 0\,, & \text{otherwise} \end{cases}$$

As with the DDC filter of (16), $p_{RX}[n]$ as above is technically implemented causally but is presented here in this manner for sake of symmetry and simplicity.

To estimate the SSNR, the SSME operates on the CB signal $r_{CB}[n]$ as opposed to the matched filter output $y[n]$. In contrast to this, the M_2M_4 estimator operates on the matched filter output $y[n]$ and does not require coherence in order to operate properly [9]. For the special case of an NRZ transmit pulse shape, such as that used for the Block V, the M_2M_4 can also operate on the CB signal $r_{CB}[n]$. In this case, from (12), the resulting SSNR estimate $\widehat{\rho}_0$ must be scaled by L/M to obtain the desired SSNR estimate $\widehat{\rho} \triangleq (L/M)\,\widehat{\rho}_0$. Regarding the M_2M_4, the SSNR estimate will be referred to as *symbol-based* when operating on the matched filter output $y[n]$ and *sample-based* when operating on the received CB signal $r_{CB}[n]$. We now proceed to analyze the SSME and M_2M_4 with regards to SSNR estimation for the BVR.

III. Split-Symbol Moments Estimator

A. Estimator Formulation

The SSNR estimator currently in use in the BVR is the Split-Symbol Moments Estimator (SSME). This algorithm is described in [10] and is designed for use with rectangular NRZ pulses. The estimator operates on the in-phase component of the received CB signal $r_{CB}[n]$ from Fig. 3, i.e., on the signal $r_I[n] \triangleq \mathrm{Re}\,\{r_{CB}[n]\}$. Specifically, a block diagram of the SSME within the context of the overall receiver functions is shown in Fig. 10, ignoring certain aspects of the receiver pertaining to presence of a subcarrier. In accordance with the results from Sec. II, the in-phase and quadrature samples have been processed by a DDC half-band filter (HBF) and decimated in order to remove double frequency terms arising from carrier demodulation. Here, the SSNR estimate is formed from only the in-phase symbols as dictated for the SSME, which requires carrier and symbol synchronization. Note that the SSNR estimation is carried out prior to the integrate-and-dump filter (IDF) (i.e., the matched filter for the NRZ pulse).

The SSME SSNR estimate is formed as follows [10].

$$\widehat{\mathrm{SSNR}} = \frac{m_p}{2\left(\frac{1}{4}m_{ss} - m_p\right)}. \tag{17}$$

Here, m_p is the *product* term given by,

$$m_p = \frac{1}{N_{sym}} \sum_{n=0}^{N_{sym}-1} x_1[n]x_2[n] \tag{18}$$

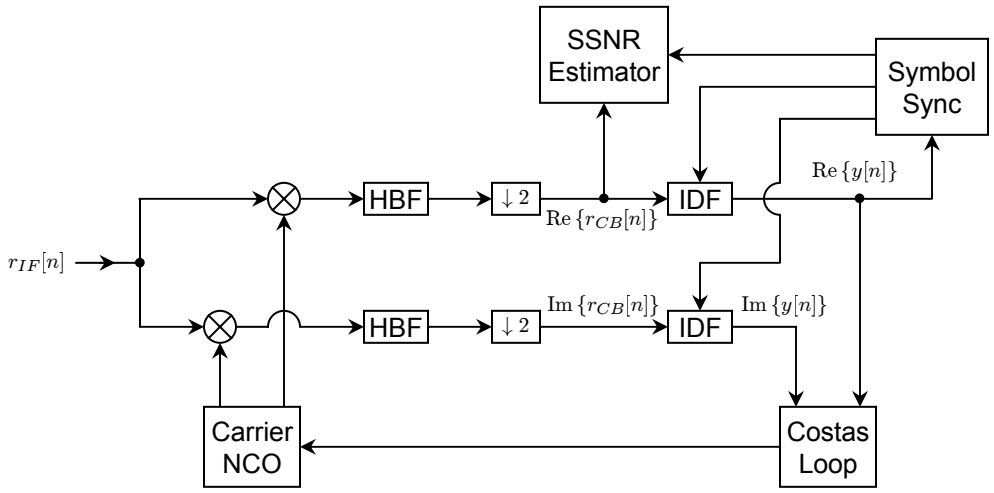


Figure 10. SSME implementation in the Block V Receiver.

and m_{ss} is the *sum-squared* term given by,

$$m_{ss} = \frac{1}{N_{sym}} \sum_{n=0}^{N_{sym}-1} \left(x_1[n] + x_2[n]\right)^2 \tag{19}$$

where N_{sym} denotes the number of symbols used to form the SSNR estimate. A block diagram of the SSME algorithm is shown in Fig. 11. Here, $x_1[n]$ and $x_2[n]$ denote, respectively, the first-half and second-half IDF outputs corresponding to the in-phase component of the received CB signal, namely $r_I[n]$ from above. For the case in which the data rate satisfies $\mathcal{R}_d = (M/L)F_s$ from (11), the half-symbol IDF outputs $x_1[n]$ and $x_2[n]$ are generated using the system shown in Fig. 12. Here, the half-symbol integrator filters $q_1[n]$ and $q_2[n]$ have the following impulse responses.

$$q_1[n] = \begin{cases} 1\,, & -\left(\left\lceil \frac{L}{2} \right\rceil - 1\right) \le n \le 0 \\ 0\,, & \text{otherwise} \end{cases} \tag{20}$$

$$q_2[n] = \begin{cases} 1\,, & -(L-1) \le n \le -\left\lceil \frac{L}{2} \right\rceil \\ 0\,, & \text{otherwise} \end{cases} \tag{21}$$

From (20) and (21), it can be shown that $x_1[n]$ and $x_2[n]$ are given below as follows.

$$x_1[n] = \sum_{k=\left\lceil \frac{Ln}{M} \right\rceil}^{\left\lfloor \frac{Ln+\left\lceil \frac{L}{2} \right\rceil - 1}{M} \right\rfloor} r_I[k]\,,\; x_2[n] = \sum_{k=\left\lceil \frac{Ln+\left\lceil \frac{L}{2} \right\rceil}{M} \right\rceil}^{\left\lfloor \frac{Ln+L-1}{M} \right\rfloor} r_I[k]$$

B. Performance Using Rectangular Pulses in the Block V Receiver

In order to determine the causes of the errors documented in Table 1, software simulations of the SSME were performed. Here, the following parameters were chosen for the

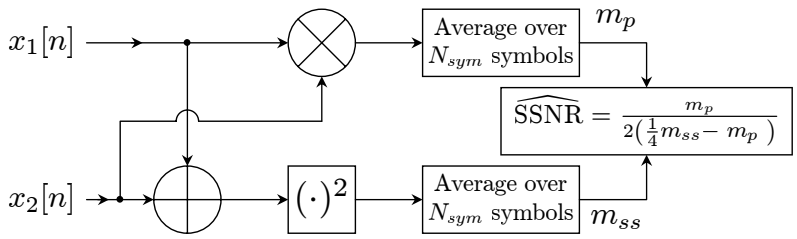


Figure 11. SSME block diagram.

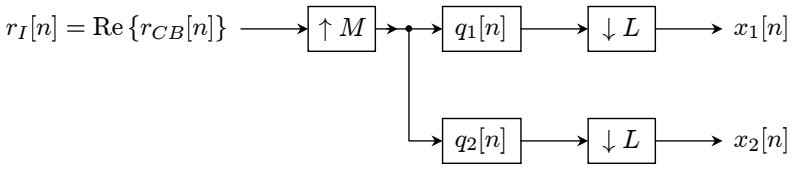


Figure 12. System used to generate the half-symbol IDF outputs required for the SSME.

simulations.

- *Data sequence*: $d[n]$ was an independent, identically distributed (i.i.d.) BPSK data stream
- *Transmit pulse shape*: $p_{TX}(t)$ was an NRZ pulse as in (5)
- *Modulation index*: $\Delta = 84^\circ$
- *Subcarrier parameters*: $F_{sc} = 0$, $\theta_{sc} = 90^\circ$
- *Carrier phase offset*: $\theta_c = 0^\circ$
- *IF carrier frequency*: $F_{IF} = 200$ MHz
- *IF sampling frequency*: $F_s' = 160$ MHz
- *CB sampling frequency*: $F_s = F_s'/2 = 80$ MHz

An intensity plot showing the SSME estimation error as a function of SNR and data rate is shown in Fig. 13, where the number of symbols used in forming the SNR estimate was chosen as $N_{sym} = 2{,}048$. Here, the estimation error is the absolute error between the true and estimated SSNRs, each in dB. More specifically, the absolute dB estimation error ξ_{abs} is defined as follows.

$$\xi_{abs} \triangleq \text{SSNR (dB)} - \widehat{\text{SSNR}} \text{ (dB)}$$

In addition, the data rates were chosen as $\mathcal{R}_d = (M/L)F_s$ according to (11), for appropriate integers L and M. From this plot we observe that the manner in which estimation error varies with SNR depends upon the data rate. More specifically, with a fixed baseband sampling rate of 80 MHz, estimation errors increase with SNR when the data rate is such that the number of samples per baseband symbol is not even. This is

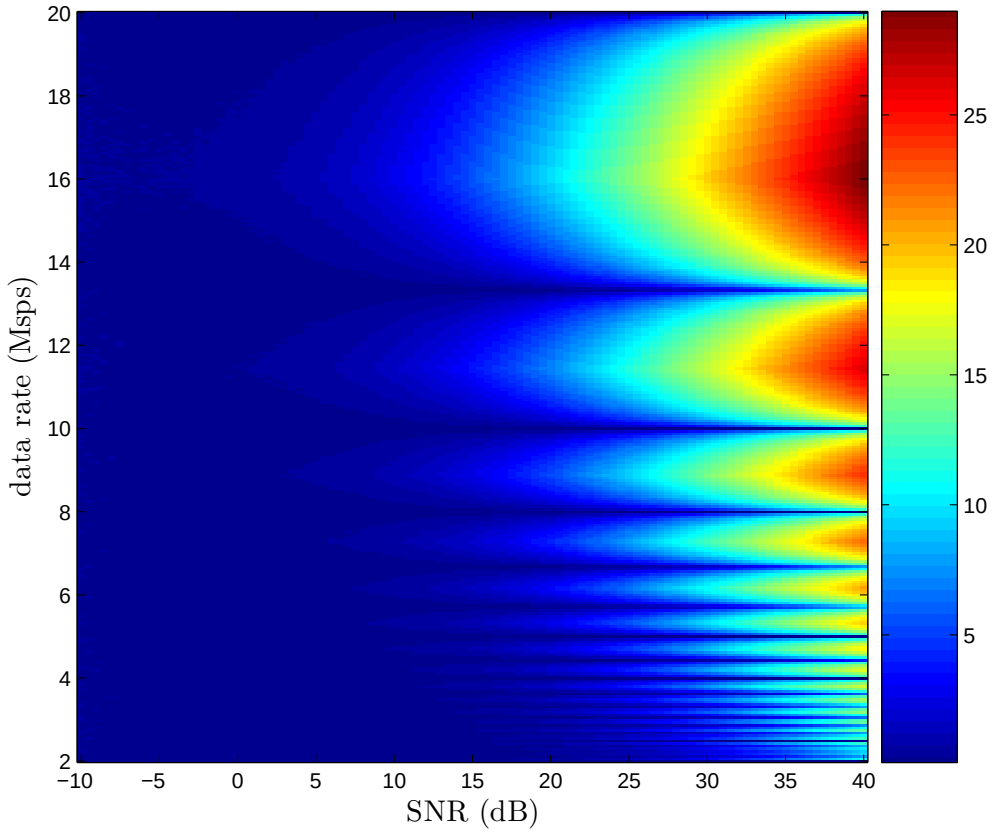


Figure 13. Intensity plot of SSME estimation error as function of symbol SNR and data rate for Block V Receiver rectangular pulses.

demonstrated by the oscillatory effect of the error as a function of data rate at a constant SNR value; the error is at a consistently low level (dark blue) when the data rate divides the baseband sampling rate of 80 MHz evenly, but increases at the other data rates. The dependence upon data rate makes sense because the SSME utilizes the sum and product of half-symbol accumulated statistics, which will be biased when the baseband symbols cannot be split evenly. The increase in estimation error as the SNR increases is due to the fact that at high SNR values, the bias error dominates the thermal noise, resulting in an output SNR saturation effect.

C. Performance Using Square-Root Raised Cosine Pulses

In addition to the problems encountered by the SSME in the BVR implementation, future bandwidth efficient modulations also pose problems to this estimator. Figure 14 shows the performance of the SSME for square-root raised cosine (SRRC) shaped pulses of various rolloff factors α with four samples per baseband symbol, and SRRC transmit and receive filter lengths of 25 taps. The number of symbols N_{sym} is 1,000. The figure shows that the symbol SNR estimate given by the current implementation of the SSME is significantly different from the true value. In [11], a method for correcting the error in the SSME for

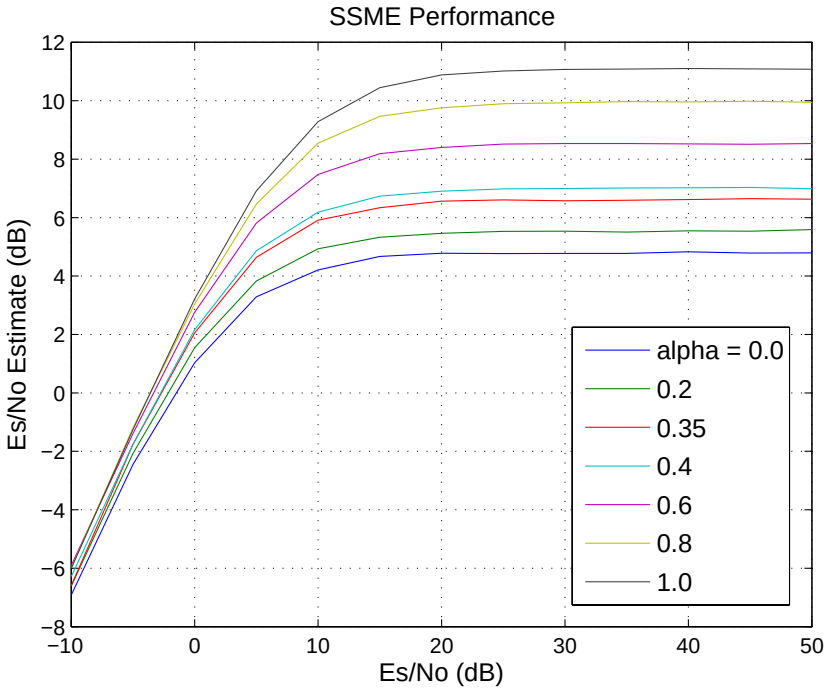


Figure 14. SSME performance for SRRC pulses, filter length 25.

filtered pulses was discussed in which the distortion to the m_{ss} and m_p terms is calculated beforehand and removed from the estimate. While this technique improves performance for filtered pulses, a different modification must be made for each new pulse shape, and the problem of uneven symbol splitting is not alleviated. Therefore we turn our attention to an alternate estimator.

IV. Second & Fourth Order Moments Estimator (M_2M_4)

A. Estimator Formulation

In [9], several estimators are considered for use with SRRC pulses, including the maximum likelihood estimator, the SSME, and other suboptimal schemes. Due to its simplicity, we have selected the second & fourth order moments (M_2M_4) estimator as a promising alternative to the SSME.

The M_2M_4 estimator is based upon solving a system of linear and quadratic equations relating the second and fourth order moments to the signal and noise powers as well as the kurtosis of the signal and noise. For complex noise and M-ary PSK signals,

$$\widehat{\text{SSNR}} = \frac{\sqrt{2M_2^2 - M_4}}{M_2 - \sqrt{2M_2^2 - M_4}} \tag{22}$$

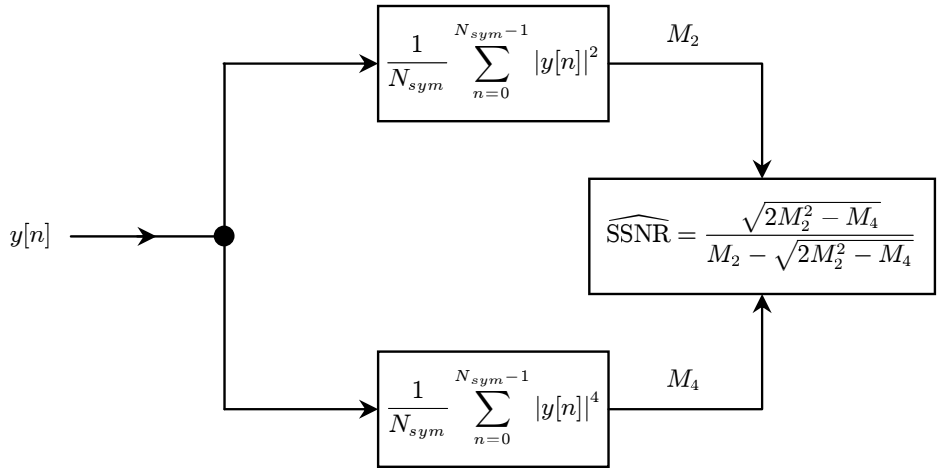


Figure 15. Symbol-level M_2M_4 block diagram.

where the second and fourth order moments (M_2 and M_4, respectively) are calculated as

$$M_2 = \frac{1}{N_{sym}} \sum_{n=0}^{N_{sym}-1} |y[n]|^2 \ , \ M_4 = \frac{1}{N_{sym}} \sum_{n=0}^{N_{sym}-1} |y[n]|^4$$

A block diagram is shown in Fig. 15. Note that here, $y[n]$ is the output of the receiver matched filter taken at the end of the n-th symbol (as in Fig. 9), and that this output can be noncoherent, i.e., although symbol timing is required, phase coherence is not. Thus, in Fig. 10, outputs from both IDF systems would be used to form the complex statistic $y[n]$.

B. Performance Using SRRC Pulses

In Fig. 16 we show the performance of the M_2M_4 estimator for SRRC pulses with four samples per symbol, SRRC filter lengths of 25, and $N_{sym} = 1{,}000$. We observe that at low SNRs, the M_2M_4 estimator shows greater variance than the SSME, which is to be expected as the noise power is raised to the fourth power in the M_2M_4, and is only squared in the SSME. This will necessitate longer averaging times for the M_2M_4 estimator. On the other hand, the M_2M_4 outperforms the SSME in terms of its bias, until it saturates at higher SNR values. As the number of samples per symbol here is even, the cause of saturation here is intersymbol interference due to the finite SRRC filter lengths. Figs. 17 and 18 demonstrate that saturation occurs later when the filter length is longer (33 and 81 taps, respectively).

C. Performance Using Rectangular Pulses in the Block V Receiver

While the M_2M_4 estimator provides a solution to estimating symbol SNR for non-rectangular pulses, we are also interested in comparing its performance with the SSME to alleviate the signal estimation problems in the BVR. Figure 19 shows the intensity plot of the M_2M_4 estimator implemented in place of the SSME in the Block V

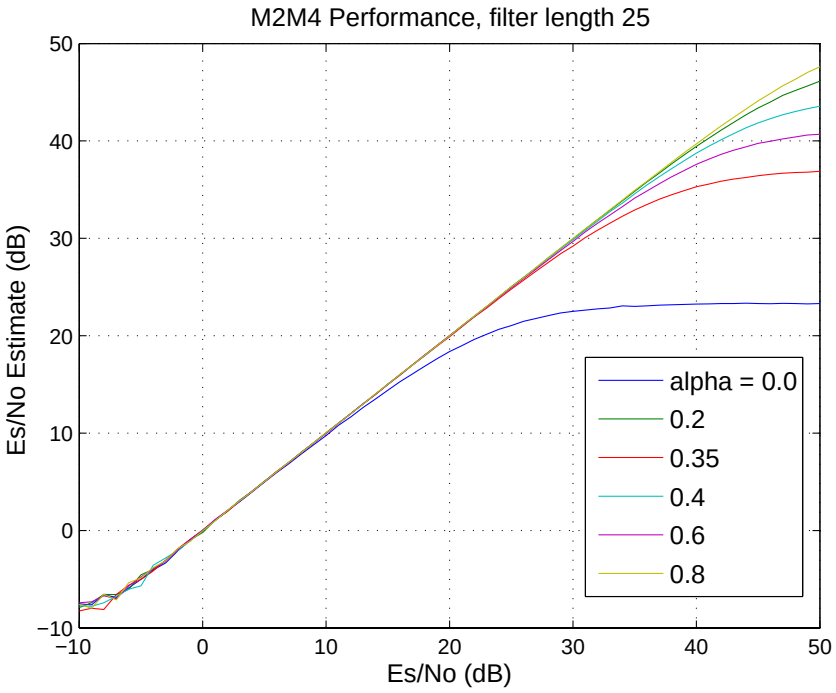


Figure 16. M_2M_4 **estimator performance for SRRC pulses, filter length** 25**.**

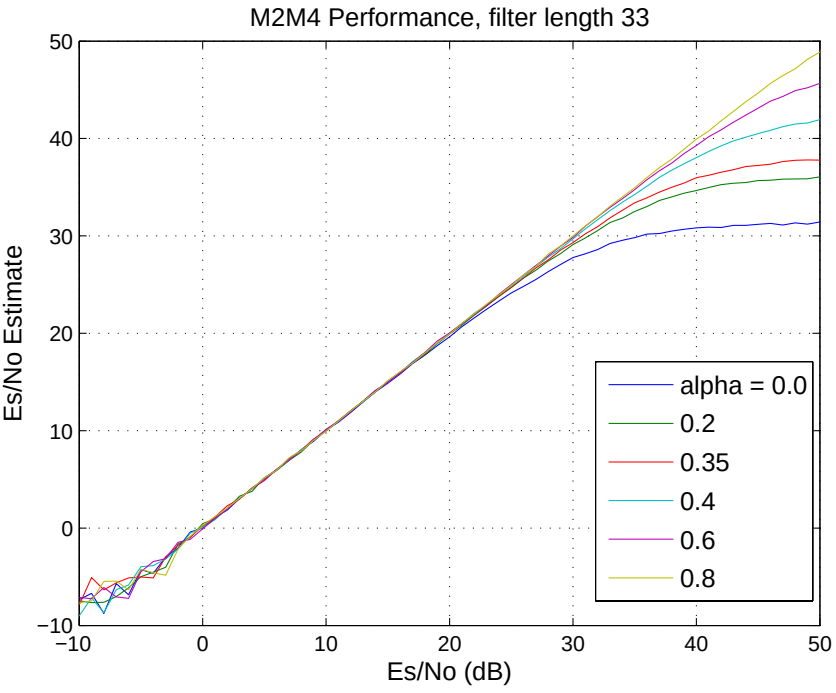


Figure 17. M_2M_4 **estimator performance for SRRC pulses, filter length** 33**.**

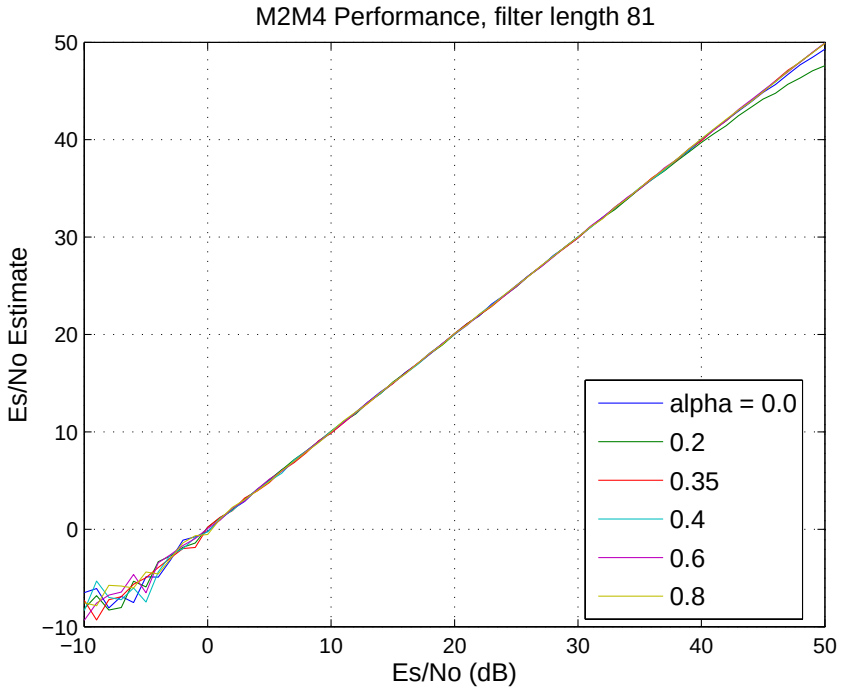


Figure 18. M_2M_4 **estimator performance for SRRC pulses, filter length** 81.

scenario with the half-band filter. Note that the oscillation of the intensity as a function of data rate is still present, but the error is minimized at twice as many data rates as in the case of the SSME. This is because the M_2M_4 operates on the symbol statistics rather than the split-symbol statistics, so an odd number of samples per symbol does not cause estimator bias; however, a non-integer number of samples per symbol still does.

In order to eliminate this bias, we have also examined use of a sample-level M_2M_4 estimator that operates on each sample at the input to the matched filter, rather than on the matched filter symbol output. Note that this technique, illustrated in Fig. 20, can only work with rectangular pulses. The reason for this is that in this case, the samples coming into the receiver matched filter appear themselves to be matched filter outputs with an SNR reduced by a factor equal to the number of samples per symbol, which is L/M here. Note that the number of higher rate symbols N'_{sym} used to form the sample-level M_2M_4 estimate is given in terms of the number of lower rate symbols N_{sym} as follows.

$$N'_{sym} = \left\lfloor \left(\frac{L}{M} \right) N_{sym} \right\rfloor$$

In other words, the number of symbols used to form the sample-level M_2M_4 estimate is roughly equal to the number of symbols used to form the symbol-level estimate, multiplied by the number of samples per symbol. The error intensity plot for the sample-level M_2M_4 is shown in Fig. 21. Note the absence of the oscillatory effect in this plot.

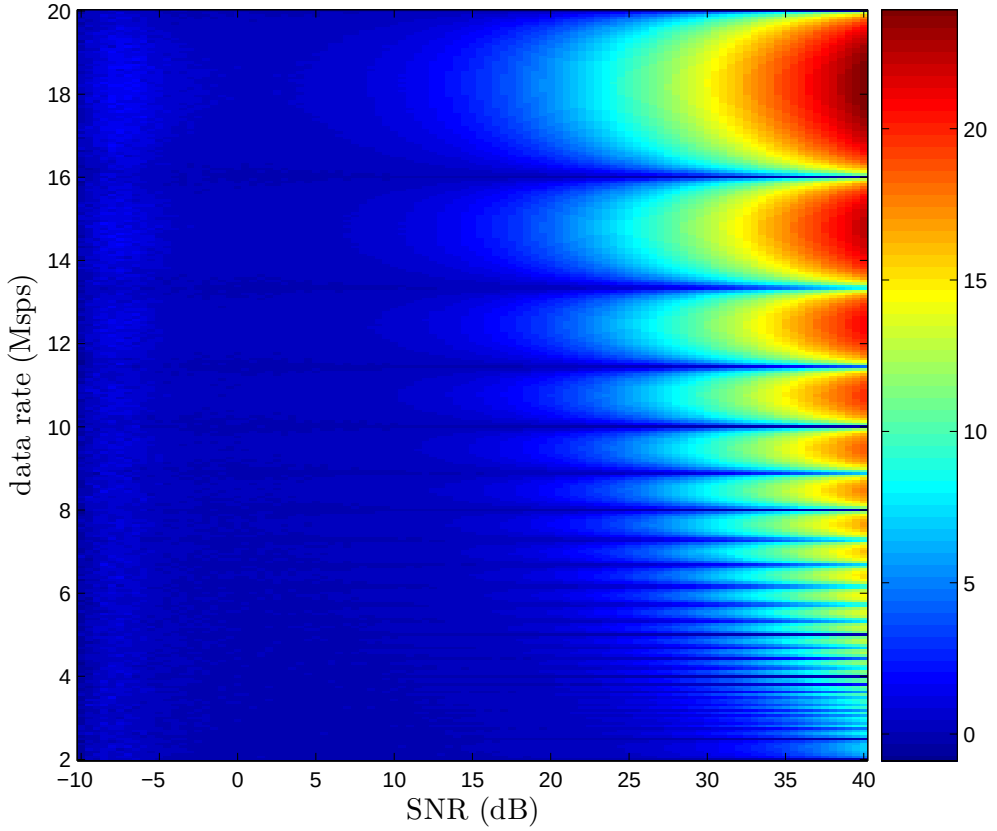


Figure 19. Intensity plot of M_2M_4 estimation error as function of symbol SNR and data rate for Block V Receiver rectangular pulses.

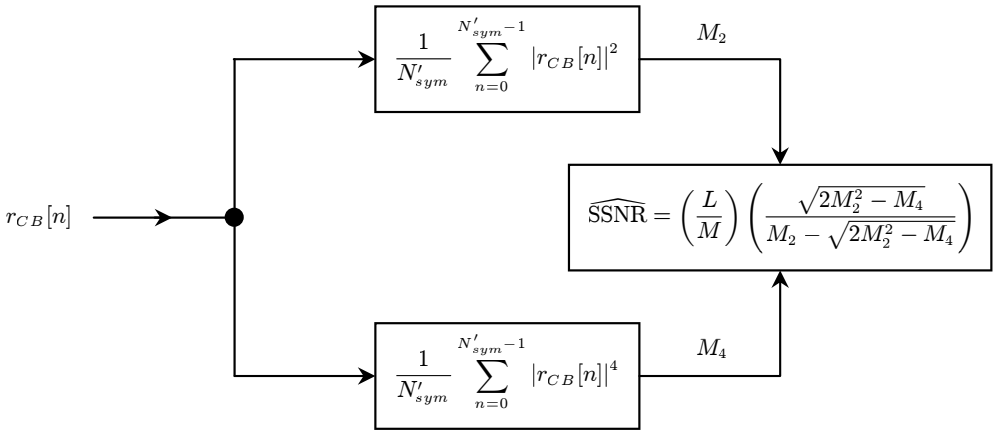


Figure 20. Sample-level M_2M_4 block diagram.

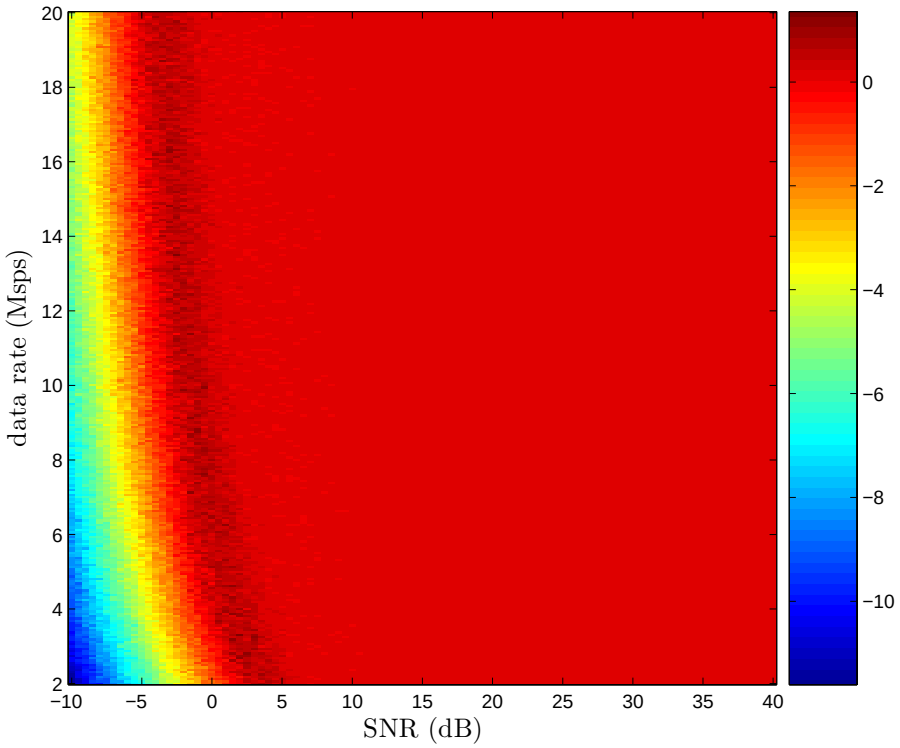


Figure 21. Intensity plot of sample-level M_2M_4 estimation error as function of symbol SNR and data rate for Block V Receiver rectangular pulses.

V. Estimator Comparison Results

In Figs. 22 through 26, we show various slices of the intensity plots and directly compare the SSME with the symbol and sample level M_2M_4 estimators. Fig. 22 shows the estimator error as a function of SNR at the rate of 2 mega-symbols-per-second (Msps), which yields 40 samples per symbol. Here, 2,048 BPSK symbols were used to form all of the estimates. We observe that the sample level M_2M_4 performs poorly below 0 dB SNR due to the very low sample SNR, and that the symbol level M_2M_4 estimator also performs worse than the SSME at the lowest SNR values due to its increased squaring loss. As the SNR increases, however, all of the estimators perform well with little error. However, Fig. 23 shows that once the data rate is increased to 2.0625 Msps, resulting in a fractional number of samples per symbol, both the SSME and the symbol-level M_2M_4 errors increase with increasing SNR, due to the irreducible bias from fractional sampling dominating thermal noise in that region. The sample-level M_2M_4 estimator performs well in the high SNR region. This trend also holds for the higher data rate case of 19.9375 Msps shown in Fig. 24. At this higher rate, there are fewer samples per symbol, resulting in greater impact of the fractional sampling bias at higher SNRs for the SSME and symbol-level M_2M_4 estimator, but reducing the amount of error in the sample-level M_2M_4 estimator due to the higher sample SNR at this data rate.

In Figs. 25 and 26, we fix the symbol SNR and plot estimator error as a function of data

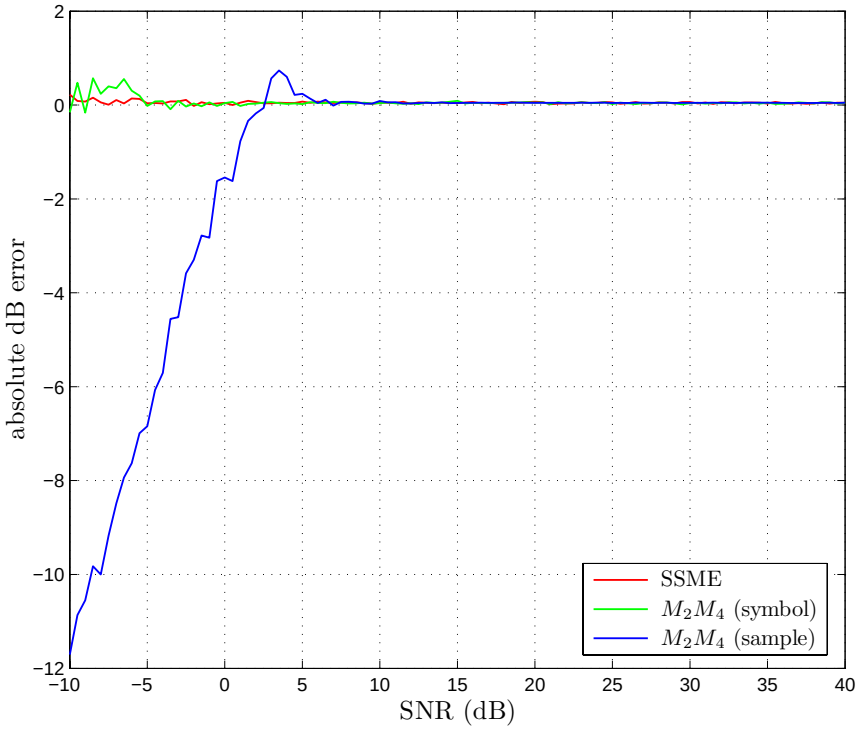


Figure 22. Estimation error as function of symbol SNR at 2 Msps data rate, rectangular pulses.

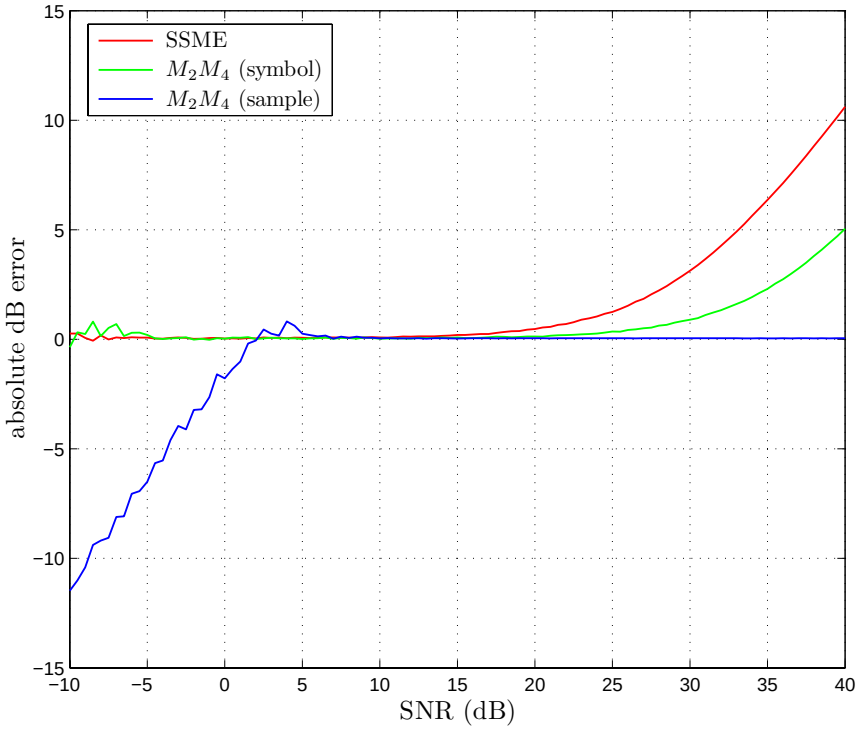


Figure 23. Estimation error as function of symbol SNR at 2.0625 Msps data rate, rectangular pulses.

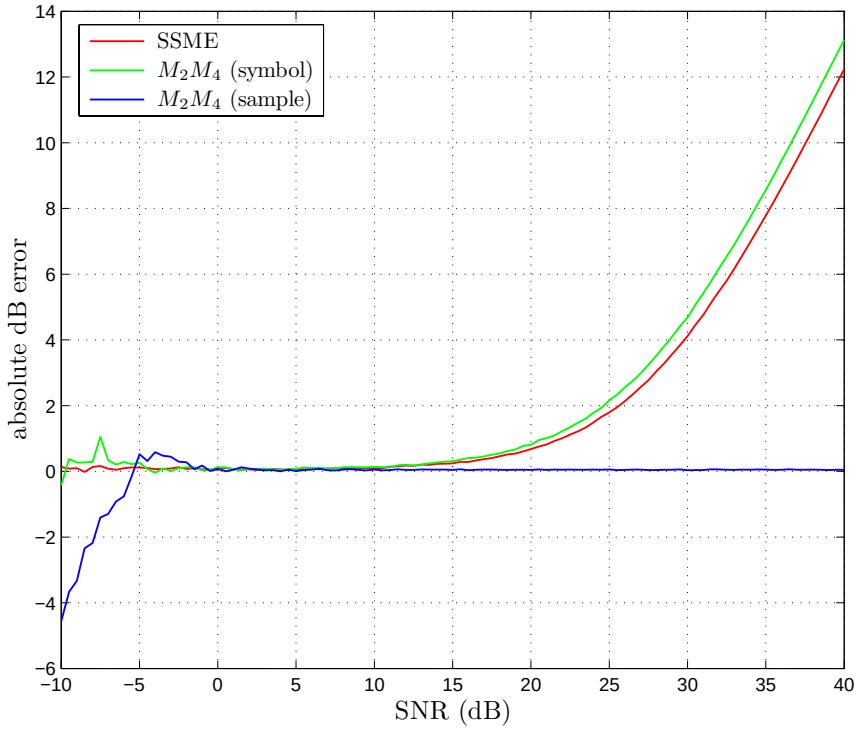


Figure 24. Estimation error as function of symbol SNR at 19.9375 **Msps data rate, rectangular pulses.**

rate. In Fig. 25 the symbol SNR is 0 dB. Here we observe the larger variance in the sample-level M_2M_4 results relative to the other estimators, along with the sample-level M_2M_4 having larger errors at low data rates (low sample SNR) and smaller errors at higher data rates (higher sampler SNR). The SSME and symbol-level M_2M_4 estimator have lower variances, but exhibit oscillations in error as a function of data rate, as explained earlier. In Fig. 26 the symbol SNR is 35 dB; the oscillatory effect is well-defined here, with the symbol-level M_2M_4 generally outperforming the SSME, and having twice as many data rates with minimal errors. Note that the error values at the local minima (even-integer divisors of 80 MHz for SSME, integer divisors of 80 MHz for symbol-level M_2M_4) should all be near zero; they are not shown as such due to the limited resolution of the data rate range in plotting. The sample-level M_2M_4 has virtually no error here.

VI. Concluding Remarks

The parametric analysis of the previous sections may be used to re-analyze the data from Table 1. We observed earlier that this data showed a trend of increasing SNR estimation error with decreasing symbol rate. It should be noted, however, that within each modulation index group, as the symbol rate increases, the symbol SNR operating point decreases, i.e, the symbol SNR is not held fixed for the various symbol rates. In addition, these symbol rates do not divide 80 MHz evenly. Therefore, the larger estimation errors at lower data rates may more likely be related to the sampling bias effect noted earlier, which

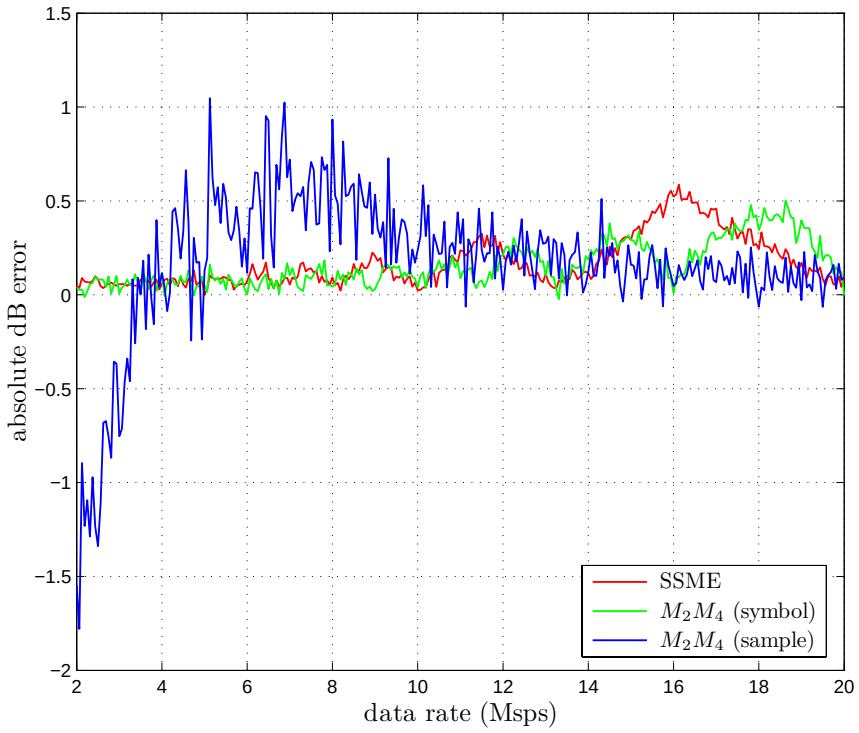


Figure 25. Estimation error as function of data rate at 0 dB symbol SNR, rectangular pulses.

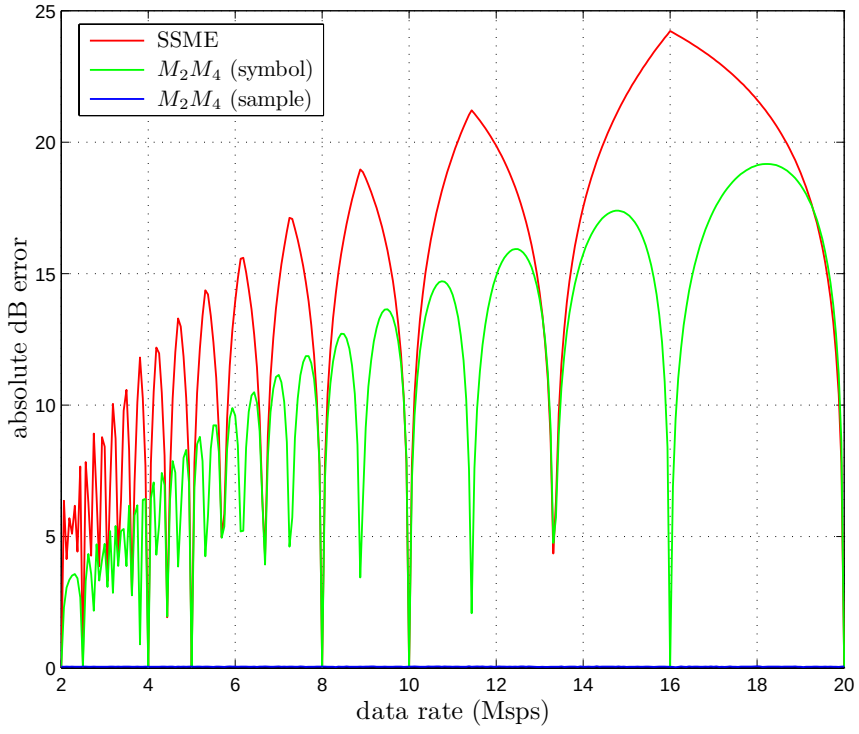


Figure 26. Estimation error as function of data rate at 35 dB symbol SNR, rectangular pulses.

results in increased errors as SNR increases, as shown in Fig. 23. As the data documented here is very limited, further testing should be performed in which the symbol rates and SNR values are each held fixed while the other is varied independently in order to more accurately evaluate the parameter dependencies. In addition, carrier phase error measurements should be recorded in order to determine if there is any significant effect upon the SNR estimate.

Based upon the results discussed here, we make the following conclusions and recommendations on signal quality measurement in the BVR:

- The primary cause of error in SNR estimation for rectangular pulses in the current Block V implementation is sampling bias when the symbol rate does not divide the 80 MHz baseband sample rate evenly.

- The estimation error caused by sampling bias increases with data rate and SNR.

- For rectangular pulses, the sample-level M_2M_4 estimator is recommended for use at high SNRs. The SNR threshold for using the sample-level M_2M_4 instead of the SSME depends upon the data rate and available integration time, as shown in Figs. 22 to 24.

- Intersymbol interference caused by possible future transponder filtering will cause saturation in the symbol SNR estimate. This cannot be fully alleviated by any sample or symbol level symbol estimator, although some degree of correction may be implemented through use of a look-up table approach.

For future upgrades to the ground receiver, including the Wideband Telemetry and Tracking Receiver architecture currently in development [12], bandwidth efficient modulations must be accommodated as well as continuously variable data rates.

- The Wideband Telemetry and Tracking Receiver architecture has a sample rate converter so that there is a fixed (4 to 8) number of samples per baseband symbol; hence there will be no sampling bias.

- The symbol-level M_2M_4 estimator is recommended due to its ability to estimate SNR when using non-rectangular pulses such as SRRC pulses.

- At low SNRs, integration times will have to be increased in order to compensate for greater squaring loss in the M_2M_4 estimator.

- Intersymbol inteference due to finite transmit and receiver filter lengths in partial-response signaling causes saturation in the SNR estimates; this may be alleviated to some limited degree by a look-up table correction approach.

References

[1] M. Srinivasan, "Summary of Signal Estimation Problems", *Interoffice Memorandum*, March 12, 2007.

[2] Consultative Committee for Space Data Systems, *Bandwidth-Efficient Modulations: Summary of Definition, Implementation, and Performance*, CCSDS 413.0-G-1, Green Book, April 2003.

[3] P. Kinman, "P_c/N_0 Estimator for Direct Modulation", personal notes, March 21, 2005.

[4] A. Monk, "Carrier-to-Noise Power Estimation for the Block V Receiver", *JPL TDA Progress Report*, vol. 42-106, pp 353-363, August 15, 1991.

[5] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, Inc., 1999.

[6] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communications Techniques: Signal Design and Detection*. Upper Saddle River, NJ: Prentice Hall PTR, 1994.

[7] S. Hinedi, "A Functional Description of the Advanced Receiver", *JPL TDA Progress Report*, vol. 42-100, pp. 1-18, February 15, 1990.

[8] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice Hall PTR, 1993.

[9] D. R. Pauluzzi and N. C. Beaulieu, "A Comparison of SNR Estimation Techniques for the AWGN Channel," *IEEE Trans. Communications*, vol. 48, no. 10, pp. 1681-1691, October 2000.

[10] M. K. Simon and A. Mileant, "SNR Estimation for the Baseband Assembly", *JPL TDA Progress Report*, vol. 42-85, May 15, 1986.

[11] D. Lee and R. Mukai and M. Simon, "Performance of the Block V Receiver SNR Estimator Using Bandwidth Efficient Modulations", Interoffice Memorandum, November 24, 2003.

[12] K. Andrews et. al., "Real-Time Wideband Telemetry Receiver Architecture and Performance," IPN PR 42-166, pp. 1-23, August 15, 2006.