6-2000

Planar Approximation for the Least Reliable Bit Log-Likelihood Ratio of 8-PSK Modulation

William H. Thesling
Cleveland State University

Fuqin Xiong
Cleveland State University, f.xiong@csuohio.edu

M. J. Vanderaar

Follow this and additional works at: https://engagedscholarship.csuohio.edu/enece_facpub

Part of the [Systems and Communications Commons](#)

[How does access to this work benefit you? Let us know!](#)

Publisher's Statement

(c) 2000 Institution of Engineering and Technology

Original Citation

Thesling, W.H., Xiong, F., & Vanderaar, M.J.. (2000). Planar approximation for the least reliable bit log-likelihood ratio of 8-PSK modulation, IEE Proceedings- Communications, 147, 3, 144-148, doi: 10.1049/ip-com:20000178

Repository Citation

Thesling, William H.; Xiong, Fuqin; and Vanderaar, M. J., "Planar Approximation for the Least Reliable Bit Log-Likelihood Ratio of 8-PSK Modulation" (2000). *Electrical Engineering & Computer Science Faculty Publications*. 126.
https://engagedscholarship.csuohio.edu/enece_facpub/126

This Article is brought to you for free and open access by the Electrical Engineering & Computer Science Department at EngagedScholarship@CSU. It has been accepted for inclusion in Electrical Engineering & Computer Science Faculty Publications by an authorized administrator of EngagedScholarship@CSU. For more information, please contact library.es@csuohio.edu.

Planar approximation for the least reliable bit log-likelihood ratio of 8-PSK modulation

W.H.Thesling, F.Xiong and M.J.Vanderaar

Abstract: The optimum decoding of component codes in block coded modulation (BCM) schemes requires the use of the log-likelihood ratio (LLR) as the signal metric. An approximation to the LLR for the least reliable bit (LRB) in an 8-PSK modulation based on planar equations with fixed-point arithmetic is developed that is both accurate and easily realisable for practical BCM schemes. Through an error power analysis and an example simulation it is shown that the approximation results in less than 0.06dB in degradation over the exact expression at an E_s/N_0 of 10dB. It is also shown that the approximation can be realised in combinatorial logic using roughly 7300 transistors. This compares favourably to a look-up table approach in typical systems.

1 Introduction

Combined modulation and coding is an efficient method of conveying information through power and bandwidth limited channels. Imai and Hirakawa's multilevel coded modulation schemes (MLCM) [1], also called block-coded modulation (BCM), can achieve trellis-coded modulation (TCM) performance in a block structure. They can be an alternative to TCM in systems where a block format, code flexibility, and decoding speed are important. Though a BCM scheme is generally not maximum likelihood (ML), its structure can offer more coding for less complexity than TCM in some systems, such as in packet switched systems.

The BCM structure applies individual codes for each bit in a modulated symbol. These component codes are denoted $C_0, C_1, \dots, C_{n-1}$ where n is the number of bits in the symbol. Each component code can be a block or convolutional code, and they can be decoded with or without channel information. The error correcting capability of the i th component code is chosen in accordance with the channel bit error probability associated with the i th ($i = 0, 1, \dots, n-1$) bit in the modulated symbol as well as taking into account information provided by the decoder from the $(i-1)$ th level. Usually the overall goal is to 'balance' the system by obtaining approximately the same decoded error probability for each level of decoded bits.

2 8-PSK log-likelihood ratio

In applications such as satellite and mobile communications the digital modulation format 8-PSK is one emerging as a practical choice in bandwidth- and power-limited situations. One example of BCM applied to 8-PSK uses three component codes, one for each bit in an 8-PSK symbol.

The associated encoder and decoder structures are illustrated in Figs. 1 and 2, where the bottom code C_0 is for the least significant bit (LSB) and the top code C_2 is for the most significant bit (MSB). As will be seen shortly, the LSB is also the least reliable bit (LRB). To obtain a benefit from multistage decoding the LSB in the constellation must alternate between binary 0 and 1 as the symbols are defined from 0 to $7\pi/8$ radians [2]. A mapping that fits this criterion is shown in Fig. 3. Each symbol is defined to have a power normalised to 1.

Fig. 1 General 8-PSK multilevel encoder/modulator

Fig. 2 General 8-PSK multilevel/multistage decoder/demodulator

Multistage decoding requires that the bottom code C_0 be decoded first. The signal metric for maximum likelihood decoding (MLD) for this code with the given constellation assignment is the log-likelihood ratio (LLR) [3, 4]. In 8-PSK, the LLR of the rightmost bit or the least reliable bit (LRB) being a binary 0 can be expressed as

$$LLR(I, Q) = \ln \left[\frac{\sum_{i=0, \text{even}}^7 e^{-(E_s/N_0)d_i^2}}{\sum_{i=0, \text{odd}}^7 e^{-(E_s/N_0)d_i^2}} \right] \quad (1)$$

Fig. 3 8-PSK constellation

where E_s is the energy per symbol, N_0 is the single-sided noise power spectral density, and d_i is the distance from the (I, Q) point to the i th symbol in the constellation. The (I, Q) point represents the demodulated I and Q components of the received signal. This expression contains the likelihood of each of four symbols that contain a binary 0 in the LRB in the numerator and the likelihood of each of the four symbols that contain a binary 1 in the denominator. The LLR as a function of the in-phase and quadrature component as a function of the E_s/N_0 equal to 2, 6, and 10dB is plotted in Figs. 4-6, respectively. Note that in each case the LLR has been normalised so that the maximum absolute value is equal to 1 in each of these plots.

Fig. 4 LLR at $E_s/N_0 = 2.0\text{dB}$

Fig. 5 LLR at $E_s/N_0 = 6.0\text{dB}$

An explicit evaluation of the LLR in real-time is very undesirable in most practical systems due to the number of complicated mathematical operations required. For this

reason a look-up table (LUT) approach is used in which the values of the LLR at a particular E_s/N_0 are calculated off-line and stored in dedicated memory. This LUT approach is commonly used for branch metrics in TCM decoders.

Fig. 6 LLR at $E_s/N_0 = 10.0\text{dB}$

Visual inspection of the figure illustrating the LLR at an E_s/N_0 of 10dB suggests that it can be approximated by a series of eight planes. The value of 10dB is of particular relevance because it is near the required E_s/N_0 to obtain a bit error rate of 10^{-6} commonly required in practical coded satellite systems. Note that the LLR for the given 8-PSK constellation is symmetric about the first quadrant. This results in the observation that the LLR is invariant with respect to the absolute value function for both the in-phase (I) and quadrature (Q) channels. Therefore by replacing I and Q with their respective absolute values, the problem is now one of evaluating one of two planar equations as a function of I and Q . The two remaining planes are symmetric about the line $I = Q$. Therefore if $I > Q$ only one planar equation at (I, Q) needs to be evaluated. If $I < Q$ the planar equation is evaluated at (Q, I) . The equation of the LLR planar approximation (LLRPA) can be expressed as

$$LLRPA(I, Q) = \max \left\{ \begin{array}{l} \alpha \times \text{abs}(I) + \beta \times \text{abs}(Q) \\ \alpha \times \text{abs}(Q) + \beta \times \text{abs}(I) \end{array} \right\} \quad (2)$$

where

$$\frac{\alpha}{\beta} = -\tan 22.5^\circ \quad (3)$$

It is important to remember that these values, whether the exact LLR or the LLR planar approximation, are the soft decision metrics to be sent to the decoder. The performance of the decoder does not depend on the absolute size of the metrics. Thus any positive scaling factor that is convenient can be chosen since multiplying all outputs by some constant has no effect on the performance of the decoder. This translates into a freedom of choice for one of the two values for α and β . The other value is determined by the ratio between α and β . If one considers fixed point arithmetic (integers) $\alpha = 29$, and $\beta = -70$ preserves the ratio quite well. Therefore the equation of the plane is given by

$$LLRPA(I, Q) = \max \left\{ \begin{array}{l} 29 \times \text{abs}(I) - 70 \times \text{abs}(Q) \\ 29 \times \text{abs}(Q) - 70 \times \text{abs}(I) \end{array} \right\} \quad (4)$$

The evaluation of the LLRPA as a function of I and Q is plotted in Fig. 7. Unlike the exact values for the LLR, the planar approximation is not dependent on the E_s/N_0 . Visually, the plot looks like an increasing good fit to the LLR as the E_s/N_0 increases.

Fig.7 Log-likelihood ratio planar approximation.

3 Error power analysis

An error power analysis can be used to find the 'effective' SNR degradation due to the use of the LLRPA as compared with the exact LLR. The approach finds the power associated with the LLRPA and considers it as an additional noise term. This noise is considered as an effective increase in the channel noise as depicted in Fig. 8. This analysis is an estimate since both the effect of the nonlinearity associated with LLR device and the fact that the noise term associated with Fig. 8b (the LLRPA noise) is correlated to the channel noise are ignored. The LLRPA noise in Fig. 8b is the error noise of the approximation. Although this noise is i.i.d. and therefore white, it is not Gaussian. However, since a decoder effectively adds and subtracts many outputs, the intermediate values tend toward a Gaussian distribution giving a valid approximate error power analysis.

Fig.8 Channel model with LLRPA and effective model relative to performing true LLR computation

The relative size of the LLRPA noise term associated with Fig. 8c is estimated by the relative size of the noise term associated with Fig. 8b. In other words, the expected

power in the noise term in Fig. 8b is used to compare to the expected power in the output from the exact LLR. The error power is given by the expected value of the squared difference signal. The difference signal is given by

$$DS(I, Q) = LLR(I, Q) - \lambda[LLRPA(I, Q)] \quad (5)$$

$$DS(I, Q) = \ln \left[\frac{\sum_{i=0, \text{even}}^7 e^{-(E_s/N_0)d_i^2}}{\sum_{i=0, \text{odd}}^7 e^{-(E_s/N_0)d_i^2}} \right] - \lambda \max \left\{ \begin{array}{l} 29abs(I) - 70abs(Q) \\ 29abs(Q) - 70abs(I) \end{array} \right\} \quad (6)$$

The coefficient λ is a scaling factor to find the best fit between the LLR and the LLRPA. The best fit is defined when the expected value of the squared value is minimised. As mentioned in Section 2, a scaling factor on the LLRPA does not effect the performance of the decoder. The coefficient λ is therefore omitted in any real system, though it is important in an analysis of error power.

Once the difference signal $DS(I, Q)$ is determined, the expected value of the squared error is found as

$$E[DS^2] = \sum_{i=0}^7 P(S_i) \iint p_i(I, Q) DS^2(I, Q) dIdQ \quad (7)$$

where $P(S_i)$ is the probability that the i th signal was sent, and $p_i(I, Q)$ is the probability density of receiving the point (I, Q) given the i th signal constellation point was transmitted. If the assumption is made that the eight signals are equally likely, owing to the symmetry of the 8-PSK constellation, this simplifies to

$$E[DS^2] = \iint p(I, Q) DS^2(I, Q) dIdQ \quad (8)$$

Here $p(I, Q)$ is the probability density of receiving the point (I, Q) given a particular symbol was transmitted. The expected squared difference signal can then be related to the expected squared signal or signal power (after the LLR operation). This is essentially the expected squared output (no approximation) which is given by

$$E[LLR^2] = \iint p(I, Q) LLR^2(I, Q) dIdQ \quad (9)$$

The ratio

$$\frac{E[DS^2]}{E[LLR^2]} \quad (10)$$

is an estimate of the additional noise-to-signal ratio due to the log likelihood ratio planar approximation. An estimate of the overall signal-to-noise ratio is obtained by

$$SNR_{estimate} = \frac{1}{\frac{1}{SNR_{channel}} + \frac{E[DS^2]}{E[LLR^2]}} \quad (11)$$

In dB, this corresponds to a reduction in SNR given by

$$SNR_{db, reduction} = SNR_{channel, db} - SNR_{estimate, db} \quad (12)$$

4 Example

Consider an E_s/N_0 of 6.0dB as an operating point. We have mentioned that a 10dB operating point of E_s/N_0 is needed

for a coded satellite system to obtain a bit error rate of 10^{-6} . The reason that the 6.0dB example is given here is to demonstrate that the LLRPA even can perform well at an SNR lower than 10dB. Fig. 5 illustrates the LLR for this SNR. The difference signal (DS) is the difference between the normalised LLR and the planar approximation (with the appropriate λ). This is shown in Fig. 9. Fig. 10 is the squared error signal. Fig. 11 is the probability density function of the received signal for a given symbol transmitted at E_b/N_0 of 6.0dB.

Fig. 9 Difference signal as a function of I and Q

Fig. 10 Squared difference signal

Fig. 11 PDF of received signal at $E_b/N_0 = 6.0\text{dB}$

The ratio of the expected squared difference signal and the expected squared true LLR is an estimate of the additional effective noise-to-signal ratio. For the example, the estimated reduction in the signal-to-noise ratio due to the log likelihood ratio planar approximation is calculated numerically to be 0.216dB. This is an estimate of the degradation associated with the LLRPA. The accuracy of the approximate degradation can be assessed through simulation. A realistic simulation example uses the rate 1/4,

16-state convolutional code given in [3] as C_0 , and 8 bits of quantisation on both I and Q . One simulation uses a LLR look-up table, while the other simulation uses the LLRPA equation. Both simulations use the same PN sequences for both the information and the noise. The exact LLR look-up table performs better for all operating points (values of channel SNR), but the difference (as measured in SNR reduction for a given BER or SNR operating point) is quite small. Fig. 12 illustrates the difference between the SNR reductions computed theoretically, and those found by simulation. The E_b/N_0 range in the simulation corresponds to a BER range of 10^{-6} and lower.

Fig. 12 Effective loss in E_b/N_0 due to LLRPA against normal operating point

5 Implementation analysis

Although it is intuitive that a hardware realisation of the LLRPA would be simpler than the exact LLR, in practice the exact LLR is computed via a look-up table (LUT). As such, an implementation analysis is really a comparison between the hardware realisation of the LLRPA and a sufficient size memory based LUT to find the exact LLR. This type of comparison is somewhat system dependent, and the comparison presented here that is based strictly on an approximate transistor count must be taken within the system context.

For example, in a demodulator/decoder that is realised mostly with VLSI technology, coming off the device to an external LUT and then back on the device has disadvantages in both the speed of external routing and the increase of VLSI complexity owing to increased I/O requirements. In this case, the number of transistors required for both techniques in the context of the particular VLSI device is a good comparison. Further, systems implemented with programmable logic such as field programmable gate arrays (FPGAs) tend to be constrained in the amount of memory space available, making the LLRPA implementation attractive. Alternatively, systems that are not fully realised in VLSI circuitry may benefit from the potential simplicity of a single memory device to perform the LLR LUT. The benefits gained from the design maturity of memory technology may outweigh a specific implementation of an algorithm such as the LLRPA.

A block diagram of the required processing for the LLRPA is shown in Fig. 13. The block diagram indicates that 8-bit data from an analogue-to-digital converter or digital filter is first converted to its absolute value. The resulting 7-bit magnitude values of I and Q are compared to find the greatest value. If the magnitude of I is greater than or equal to the magnitude of Q , the I data follow the top leg of processing and the Q the bottom leg. If the mag-

nitude of Q is greater than the magnitude of I this is reversed. The appropriate values are then multiplied by either 29 or 70 and are then subtracted. The result is then divided by 256 to maintain only the six most significant bits. The complexity of the LLRPA implementation can be approximated through a rough estimation of the complexity in terms of gates for each of these functions. These gate counts are then converted to an overall estimate of transistor count. The accuracy of the approximation is subject to the goals of a particular system in terms of speed, power consumption, or real estate. Further, the number representation presented by the upstream hardware and required by the downstream hardware can also be relevant.

Fig. 13 Implementation block diagram of LLRPA

First, in its worst case, the absolute value function requires a magnitude compare, a select, and then an 8-bit addition or subtraction, requiring a rough total of 200 gates. Secondly, the magnitude comparison and select require about 80 gates. Next, the fixed multipliers can be realised by shifts and adds resulting in about 250 gates. The final subtractor requires approximately 200 gates and the divider chooses the six MSBs. Assuming an average of ten transistors per gate, the total approximate transistor count is 7300. For a rough comparison, the LUT table would have a $2^8 \times 2^8 = 65\,536$ memory addresses. If each address contains six bits to maintain good quantisation accuracy this corresponds to a $65\,536 \times 6$ memory. A static random access memory (SRAM) that used five transistors per cell would require 1.97×10^6 transistors. This ignores the transistors required for column decoders, row decoders, and read/write circuitry. These estimates indicate that the LLRPA requires approximately 270 times fewer transistors than the LUT. Also, the LLRPA computation can be implemented in parallel to obtain an operating speed increase. In this case, the number of transistors will increase by the factor of the speed increase plus the gates required to multiplex and demultiplex the I/O.

6 LLR and the C1 code

Once the bottom code C0 is decoded and re-encoded, the re-encoded data is used to determine which of two 4-PSK symbol sets is used for the remaining two bits. That is set $\{S0, S2, S4, S6\}$ or set $\{S1, S3, S5, S7\}$ with respect to Fig. 3. Given one of these two sets, the least reliable bit (which is really the middle bit now) must also alternate between 0 and 1 as the symbols are encountered moving around the circumference of the circle. The data impressed onto this symbol is from the C1 code. For decoding purposes, the optimum signal metric is the log likelihood ratio for this constellation. If we consider the set $\{S0, S2, S4, S6\}$

then the LLR of the right-most bit (middle bit) being a binary 0 against being a 1 can be expressed as

$$LLR_{4PSK}(I, Q) = \ln \left[\frac{\sum_{i=0,4} e^{-(E_s/N_0)d_i^2}}{\sum_{i=2,6} e^{-(E_s/N_0)d_i^2}} \right] \quad (13)$$

which can be approximated by

$$LLRPA_{4PSK} = \text{abs}(I) - \text{abs}(Q) \quad (14)$$

We state without proof that error associated with this approximation is less than that associated with C0. If the set in question is the set $\{S1, S3, S5, S7\}$, a 'rotation' operation will need to be performed. That is, I and Q should exchange positions in eqn. 14.

7 Conclusions

It has been shown that the planar approximation to the log-likelihood ratio in the least reliable bit of an 8PSK modulation format is suitable for practical systems. The approximation results in very little degradation in effective SNR as indicated by an approximate error power analysis and verified through simulation results at relevant operating points. The complexity of the LLRPA discussed as a comparison between the implementation of LLRPA and an equivalent memory based LUT evaluating the exact LLR indicates that the LLRPA is practical for many systems.

Although appropriate for coded 8PSK, the orthogonality of gray-coded QPSK and the single dimension of BPSK make the calculation of the appropriate LLR metric simply equivalent to either the value of I or Q . In these cases an approximation is not necessary. For higher-order PSK systems, a similar approach for a planar approximation can be taken. Although the decision device to determine the multipliers for I and Q may be more complex, the required size LUT for an exact LLR may get undesirably large. It is uncertain whether there exist small integer multipliers that will preserve a good approximation. Finally, due to the complex decision regions it is unclear whether QAM modulation schemes could benefit from a similar approximation technique.

8 Acknowledgments

William Thesling was supported by a NASA space grant/OAI fellowship from the Ohio Space Grant Consortium. Mark Venderaar's work sponsored by NASA contract NAG3-27186.

9 References

- 1 IMAI, H., and HIRAKAWA, S.: 'A new multilevel coding method using error-correcting codes', *IEEE Trans. Inf. Theory*, 1977, **IT-23**, (3), pp. 371-376
- 2 VANDERAAR, M., WAGNER, P., and BUDINGER, J.: 'Least reliable bits coding (LRBC) for high data rate satellite communications'. Proceedings of the AIAA 14th international conference on *Communications satellite systems*, Washington DC, March 1992, pp. 507-514
- 3 LIN, S., and COSTELLO, D.J. Jr.: 'Error control coding: Fundamentals and applications' (Prentice-Hall, Englewood Cliffs, NJ, 1983)
- 4 WOERTZ, H., and HAGENAUER, J.: 'Multistage coding and decoding for a M-PSK system'. Proceedings of the IEEE conference on *Global telecommunications*, San Diego, CA, Dec. 1990, pp. 698-703