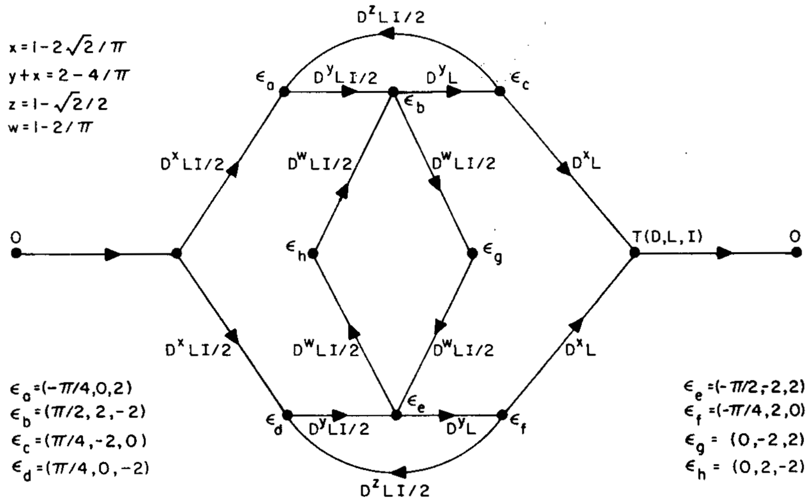

Fig. 10. Reduced error state diagram for MSK modulation and the TFM encoding polynomial.

simplifies the analysis. Only binary transmission with correlative encoding was considered and the tradeoff between power and bandwidth was made explicit through simple expressions for the bit error rate. It was shown that the duobinary encoding rule can give relatively simple receivers with either bandwidth or power efficiency, the latter occurring for a digital FM modulation with modulation index 0.75.

ACKNOWLEDGMENT

I wish to thank G. Woodruff and J. Vaisey of the Department of Electrical Engineering, Queen's University, for their independent derivation of the error graph for duobinary MSK. Also, the modulation state diagram for TFM encoding and MSK modulation was used first in another context by J. Vaisey.

REFERENCES

[1] R. de Buda, "Coherent demodulation of frequency-shift-keying with low deviation ratio," *IEEE Trans. Commun.*, vol. COM-20, pp. 429–435, June 1972.

[2] A. Lender, "The duobinary technique for high-speed data transmission," *IEEE Trans. Commun. Electron.*, vol. 82, pp. 214–218, May 1963.

[3] G. S. Deshpande and P. H. Wittke, "Correlative encoded digital FM," *IEEE Trans. Commun.*, vol. COM-29, pp. 156–162, Feb. 1981.

[4] G. D. Forney, Jr., "Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 363–378, 1972.

[5] ——, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[6] T. Aulin, N. Rydbeck, and C.-E. W. Sundberg, "Continuous phase modulation—Part II: Partial response signaling," *IEEE Trans. Commun.*, vol. COM-29, pp. 210–226, Mar. 1981.

[7] J. B. Anderson and D. P. Taylor, "A bandwidth efficient class of signal space codes," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 703–712, Nov. 1978.

[8] L. F. Lind and A. A. Albuquerque, "Spectral calculation of partial response multi-*h* phase codes," *Electron. Lett.*, vol. 17, pp. 711–713, Oct. 17, 1981.

[9] F. de Jager and C. B. Dekker, "Tamed frequency modulation: A novel method to achieve spectrum economy in digital transmission," *IEEE Trans. Commun.*, vol. COM-26, pp. 523–528, May 1978.

[10] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*. New York: McGraw-Hill, 1979.

[11] J. K. Omura and D. Jackson, "Cutoff rates for channels using bandwidth efficient modulations," in *Proc. Nat. Telecommun. Conf.*, Houston, TX, Dec. 1980, pp. 14.1.1–14.1.11.

[12] D. E. Jackson, "Bandwidth efficient modulation and coding," Ph.D. dissertation, Univ. California, Los Angeles, 1980.

[13] J. K. Omura, "Performance bounds for Viterbi algorithms," in *Proc. Int. Conf. Commun.*, Denver, CO, June 1981.

[14] T. Aulin, "Symbol error probability bounds for coherently detected continuous phase modulated signals," *IEEE Trans. Commun.*, vol. COM-29, pp. 1707–1715, Nov. 1981.

[15] R. F. Pawula and R. Golden, "Simulations of convolutional coding/Viterbi decoding with noncoherent CPFSK," *IEEE Trans. Commun.*, vol. COM-29, pp. 1522–1527, Oct. 1981.

[16] C. Mead and L. Conway, *Introduction to VLSI Systems*. Reading, MA: Addison-Wesley, 1980.

[17] H. E. Rowe and V. K. Prabhu, "Power spectrum of a digital frequency-modulation signal," *Bell Syst. Tech. J.*, vol. 54, pp. 1095–1125, Aug. 1975.

[18] P. J. McLane, "The Viterbi receiver for correlative encoded digital FM," Dep. Elec. Eng., Queen's Univ., Kingston, Ont., Canada, Rep. 82-1, Mar. 1981.

[19] D. Divsalar and J. K. Omura, "Performance of intersymbol interference channels under mismatched conditions," in *Proc. Int. Conf. Commun.*, Toronto, Ont., Canada, June 1978, pp. 30.3.1–30.3.4.

[20] C. E. Sundberg, T. Aulin, and A. Svensson, "Viterbi detectors with reduced complexity for partial response continuous phase modulation," in *Proc. Nat. Telecommun. Conf.*, New Orleans, LA, Nov. 1981.

A Modification of Fletcher's Checksum

BETTY SALZBERG

Abstract—**We modify J. G. Fletcher's checksum algorithm (easily implemented by software) to place two check octets in an arbitrary place in a data unit, rather than at the end. This is useful if one wants the checksum information in the header of a data unit.**

Paper approved by the Editor for Data Communication Systems of the IEEE Communications Society for publication without oral presentation. Manuscript received May 10, 1982.

The author is with the College of Computer Science, Northeastern University, Boston, MA 02115.

J. G. Fletcher has invented a checksum algorithm which can be more easily implemented by software than cyclic redundancy codes. Fletcher shows, in his article referenced below, that his method, while not quite as good as the CRC in detecting errors, is very good. For example, when the arithmetic is done modulo 255 (wrap-around carry) with two check octets (the case we consider below), two undetected bit errors must be 2040 bits apart and the fraction of undetected 16-bit burst errors is 0.000019 percent.

Fletcher's check octets are placed at the end of the data unit. We modify his algorithm to place two check octets in the nth and $(n + 1)$st octet for arbitrary n. The sender must know the value of n but the receiver need not know the value of n. Thus, when the data unit arrives, it can be processed without having to figure out which octets contain the check information. This situation occurs when the check octets are in a variable location in the header of a data unit. In this case, the location of the check octets is indicated by part of the header which can be in error itself.

In addition, by using this method rather than a simple permutation of the data unit octets, the minimum separation of undetected double bit errors (2040) is preserved.

Basically, Fletcher calculates R checksum octets using modulo M arithmetic. For us, R will be 2, the checksum octets are denoted $C(0)$ and $C(1)$, following Fletcher's notation, and M will be 2**8 − 1, i.e., arithmetic will be modulo 255. Using this notation, the algorithm becomes *Fletcher's algorithm for two checksum octets.*

1) [initialize] $B \leftarrow$ first octet in data unit. $C(0) \leftarrow 0$. $C(1) \leftarrow 0$.
2) [add modulo M] First, $C(0) \leftarrow C(0) + B \pmod{M}$. Then, $C(1) \leftarrow C(1) + C(0) \pmod{M}$.
3) [more octets?] If there are more octets in the data unit, then set B to next octet and go to step 2. If not (B is last octet), then terminate.

If we denote the octets $B(1)$, $B(2)$, $B(3)$, $\cdots$, $B(\text{length})$ where length is the number of octets in the data unit, then after this algorithm terminates,

$$C(0) = \sum_{i=1}^{\text{length}} B(i)$$

and

$$C(1) = \sum_{i=1}^{\text{length}} (\text{length} - i + 1) B(i).$$

These formulas can be proven using mathematical induction.

Now we show how to modify this algorithm to place the check octets in an arbitrary place in the data unit.

Assume we wish to use the nth and $(n + 1)$st octets for our check octets. First perform the Fletcher algorithm as above with zero in the nth and $(n + 1)$st octet. We obtain

$$C(0) = \sum_{i=1}^{n-1} B(i) + \sum_{i=n+2}^{\text{length}} B(i)$$

$$C(1) = \sum_{i=1}^{n-1} (\text{length} - i + 1) B(i) + \sum_{i=n+2}^{\text{length}} (\text{length} - i + 1) B(i).$$

Now, let X stand for the value we are going to put in the nth octet and let Y stand for the value to go in the $(n + 1)$st octet. Then, in order for the algorithm to produce zeros at the other end, we want

$$X + Y = -C(0) \pmod{M}$$

and

$$(\text{length} - n + 1)X + (\text{length} - n)Y = -C(1) \pmod{M}.$$

Solving these two equations simultaneously, we obtain

$$X = (\text{length} - n) * C(0) - C(1) \pmod{M}$$

$$\{\text{check octet in } n\text{th place}\}$$

$$Y = (\text{length} - n + 1) * (-C(0)) + C(1) \pmod{M}$$

$$\{\text{check octet in } (n + 1)\text{st place}\}.$$

To summarize,

1) [clear check octets] Put zero in the nth and $(n + 1)$st octets of the data unit.
2) [use algorithm] Perform Fletcher's algorithm as above.
3) [nth check octet] The check octet to go into the nth place is

   $$(\text{length} - n) * C(0) - C(1) \pmod{M}.$$

4) [$(n + 1)$st octet] The check octet to go in the $(n + 1)$st place is

   $$(\text{length} - n + 1) * (-C(0)) + C(1) \pmod{M}.$$

5) Transmit the message.
6) Perform Fletcher's algorithm at the other end. The checksums should be zero, if there have been no errors.

REFERENCES

[1] J. G. Fletcher, "An arithmetic checksum for serial transmissions," *IEEE Trans. Commun.*, vol. COM-30, pp. 247–252, Jan. 1982.

Performance of an IJF-OQPSK Modem in Cascaded Nonlinear and Regenerative Satellite Systems

THO LE-NGOC AND KAMILO FEHER

Abstract—**The probability of error (P_e) performance of an intersymbol-interference and jitter-free-offset QPSK (IJF-OQPSK) modem in conventional and regenerative satellite channels is studied and compared to that of a QPSK modem using computer simulation**

Paper approved by the Editor for Space and Satellite Communication of the IEEE Communications Society for publication after presentation at the International Conference on Communications, Denver, CO, June 1981. Manuscript received April 15, 1981; revised May 15, 1982.

T. Le-Ngoc is with SR Telecom, Dorval, P.Q., Canada H9P 1H7, and the University of Ottawa, Ottawa, Ont., Canada K1N 9B4.

K. Feher is with the Department of Electrical Engineering, University of Ottawa, Ottawa, Ont., Canada K1N 9B4, and SPAR Aerospace Limited, Ste. Anne de Bellevue, P.Q., Canada.

0090-6778/83/0200-0296$01.00 © 1983 IEEE