$$d^2 = d^2(s_k, s_l) + d^2(s_k, s_l) = 2d^2(s_k, s_l) \qquad (7)$$

where s_k and s_l can be any possible signals. For the (16, 9, 4) array code using the vectorial coefficients of signals given in eqn. 5, d^2_{free} is equal to 4. The minimum Euclidean distance, d_0^2, of the FSK/4PSK signal space is 2. Using these values, eqn. 6 provides 3dB asymptotic coding gain. However, the last row of array codes is generated from the column parity checks, as shown in Fig. 3, which produced one additional symbol in comparison with the uncoded transmission. In the calculation of overall coding gain of array codes in CCM schemes, this factor needs to be taken into account. Therefore the following symbol ratio is defined as

$$R_{array} = \frac{\text{number of uncoded symbols}}{\text{number of array coded symbols}} \qquad (8)$$

For the (16, 9, 4) array code R_{array} is 3/4; then the asymptotic coding gain based on this symbol ratio is calculated to be 2.25dB over the uncoded FSK/4PSK scheme. Theoretically, a maximum coding gain of 3dB can be obtained as the symbol ratio approaches unity ($R_{array} \rightarrow 1$).

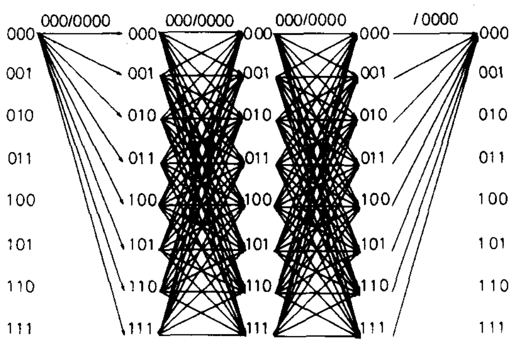

Fig. 2 *Trellis diagram of (16, 9, 4) array code*

Conclusions: A combined coding and modulation technique based on the trellis decoding of array codes is described. It is shown that a trellis coded phase/frequency modulation scheme can provide an asymptotic coding gain up to 3dB in comparison with an uncoded scheme.

29 July 1993
Electronics Letters Online No: 19931185

L. Kaya and B. Honary (*Hull-Lancaster Communication Research Group, Department of Engineering, University of Lancaster, Lancaster LA1 4YR, United Kingdom*)

References

1 UNGERBOECK, G.: 'Channel coding with multilevel/phase signals', *IEEE Trans.*, 1982, **IT-28**, (1) pp. 55–67
2 PADOVANI, R., and WOLF, J.K.: 'Coded phase/frequency modulation', *IEEE Trans.*, 1986, **COM-34**, (5) pp. 446–453
3 FARRELL, P.G.: 'Array codes, algebraic coding theory and applications', G. Longo (Ed.), (Springer-Verlag, 1979, pp. 231–242
4 HONARY, B, KAYA, L., MARKARIAN, G.S., and DARNELL, M.: 'Maximun-likelihood decoding of array codes with trellis structure', *IEE Proc. I*, 1993

Synchronising voiceband modems

R.G.C. Williams

Indexing terms: Voiceband modems, Synchronisation, Quadrature amplitude modulation

Modern voiceband modems often use a symbol rate that is not a factor of their data rate. Combined with the possible use of multidimensional trellis-coded modulation, this means that modems must transmit a fractional number of bits per symbol. To allow for this, bits must be mapped in blocks of symbols. This raises the issue of how to synchronise these blocks. The Letter gives a fast and reliable means to do this.

Introduction: Modern voiceband modems often use a symbol rate that is not a factor of their data rate. Many commercial 19.2kbit/s modems use a symbol rate of (2400 × 8)/7 . 2743 symbol/s (e.g. 1) allowing transmission of 7bit/symbol. This is a good symbol rate for transmitting 19.2kbit/s but most other 2400 bias multiples require a fractional number of bits per symbol to be transmitted. Multidimensional coded modulation [2] is another popular technique among modem designers. This adds a fractional number of redundant bits per symbol. Thus most modern modems have to cope with the transmission of a fractional number of bits per symbol. This means that before transmission, the data to be transmitted must be placed in blocks that will contain an integer number of bits. Several techniques exist for sending a block of symbols that contain a fractional number of bits per symbol [3, 4].

The receiver must accurately maintain the block boundaries to ensure valid data are passed to the user. To do this a synchronisation method is needed. Ideally the method chosen should not degrade the performance of the modem when it is working properly. The method should also be able to detect loss of synchronisation quickly and reliably in all of the operating conditions of the modem. Typically modems will be deemed to be working satisfactorily if they are producing a 1000bit block error rate of 10^{-2} or less. Finally the synchronisation method should be able to recover true synchronisation when a loss of synchronisation has been detected. This Letter presents a method for synchronisation that fulfils all of these requirements.

Synchronising using bit inversion: In a modem using trellis-coded modulation, a redundant bit is generated from the trellis code every code symbol. A code symbol may span several QAM symbols. If the redundant bit is periodically inverted before mapping all the bits to QAM symbols, then a controlled disruption to the trellis path will be seen at the receiver. This disruption can be compensated for by the receiver. As long as the transmitter and receiver stay in synchronisation there is no difference to the performance of the modem. If synchronisation is lost, the receiver will notice a disturbance in the values of the minimum metrics in its decoder. After a period of averaging, the cause of this disturbance can be confirmed as loss of synchronisation rather than the noise on the channel. By monitoring and averaging the minimum metrics in the decoder the receiver can make a reliable decision that synchronisation has been lost.

The best spacing for the inversions depends on the trellis code and the symbol rate used by the modem. For a modem using a 32-state 4-D trellis code [5] at 2400symbol/s it has been found that, for a given reliability, the detection is quickest if the redundant bit is inverted every 2ms. The probability of a false detection of loss of synchronisation should be kept down to the region of 10^{-11}. This keeps the probable occurrence of false detection to less than once a year in continuous use. With these parameters it is possible to detect loss of synchronisation within 0.5s on a channel that would produce a 1000bit block error rate of 10^{-2}.

Many other ways of disrupting the trellis path can be envisaged, but this method has been found to be the cleanest and easiest to implement. It is applicable to all modems that use trellis-coded modulation no matter which decoding algorithm they use. This method has been chosen as the method for synchronisation in the emerging ITU-T V.fast recommendation.

Recovering synchronisation: Once loss of synchronisation has been detected the modem needs to recover true synchronisation. It may be possible to do this by initiating a fast retrain of the modem. In half-duplex applications (such as facsimile), it will probably be quicker for the synchronisation method to recover synchronisation itself. To do this the receiver compensation should be turned off. This is necessary because this compensation may mask the effect of the transmitter inversions. The inversions will (in general) become clearer without the receiver compensation. Similar techniques to the detection of synchronisation loss can now be used to find the inversion position. In simple cases this will be the end of the recovery process and the receiver can start to pass valid data to the user again.

However, it may be the case that a modem is capable of using several symbol rates and in this case it is desirable to synchronise to a common boundary between the blocks of each symbol rate. In the cases where two modem pairs are used either side of a digital demod/remod link this is essential for changing data rate. This means that some method of finding 'superframe' synchronisation must be devised. Superframe synchronisation may also be necessary where the shortest available synchronisation frame is much greater than the preferred inversion frequency. In this case detection of loss of synchronisation and the recovery from this loss will be quicker if a superframe is used and inversions are kept more frequent.

Recovering superframe synchronisation: To enable superframe synchronisation a unique pattern of inversions must be used across the superframe. The simplest method of doing this is to keep the same interval between inversions but occasionally miss an inversion out. The number of inversions that are missed out must be kept low to enable the loss of synchronisation detection algorithm to function properly. The more inversions that are missed out the easier it is to spot the superframe position. It has been found that retaining 75% of the inversions is sufficient to enable fast loss of synchronisation detection. The pattern of these inversions will determine how quickly and reliably superframe synchronisation can be found.

One approach would be to have a pattern of inversions in which every binary b-tuple was unique, e.g. every 6-tuple in the pattern 0 1 1 1 1 1 1 0 1 0 1 1 1 0 1 1 is unique. By minimising b (as it is here for the weight of the pattern) it could be possible to find the superframe position as quickly as possible. This method is not very reliable because in normal operating conditions, the noise on the channel is capable of masking the presence or absence of inversions. This means an averaging process is needed which slows down the recovery of superframe synchronisation.

Another approach is to design the pattern so that its cyclic shifts are as far apart as possible. For short patterns this is the most effective method of choosing the pattern, e.g. all the cyclic shifts of the pattern 0 1 1 1 0 1 1 1 1 1 1 1 1 0 1 0 are at least a Hamming distance of 6 from each other. The reliability of finding superframe synchronisation is increased with the minimum distance between cyclic shifts of the pattern. It is possible to monitor enough inversion positions to cover the length of the pattern. A decoding algorithm may then be employed on the inversions (along with the channel noise on them) to find the closest codeword in the codebook of cyclic shifts of the pattern. The resulting cyclic shift will give the superframe position. The reliability of this decoding obviously increases with the distance between the cyclic shifts of the pattern. For m zeros in a pattern, the largest possible minimum distance between cyclic shifts is $2(m-1)$

The pattern above has the maximum distance possible between its cyclic shifts. The truncated pattern of any length also has the maximum possible distance between its cyclic shifts. This pattern is proposed as the inversion pattern in the forthcoming V.fast recommendation.

Conclusion: A method of maintaining synchronisation in a modem modem by inverting the redundant bit from the trellis code has been presented. This is a reliable method that does not degrade performance when the modem is working properly. The possibility of maintaining superframe synchronisation has also been discussed. It has been argued that to maintain the speed of detection of loss of synchronisation, inversions need to happen quite often. This leads to a high weight pattern of inversions. It has also been argued that this pattern should be as distinct as possible from all of its cyclic shifts to enable reliable superframe recovery. The superframe position is found by effectively decoding a weighty cyclic code. The choice of pattern is a design problem for this cyclic code. A codeword must be found of length n with weight $>w$ that will form a codebook from all its cyclic shifts with the largest possible Hamming distance. The largest Hamming distance such a codebook can have is $2(n-w-1)$.

Acknowledgments: This work has been driven by the need to find an acceptable method of synchronisation to include in the V.fast recommendation. As such, the conclusions reached in this letter are the result of several useful discussions with participating companies. The author would particularly like to thank S. Olafsson, V. Krishnan, D. Forney and K. Jones.

18 August 1993
Electronics Letters Online No: 19931197

R. G. C. Williams (*BT Laboratories, Martlesham Heath, Ipswich, IP5 7RE, United Kingdom*)

References

1 LANG, G.R., and LONGSTAFF, F.M.: 'A Leech lattice modem', *IEEE Sel. Areas Commun.*, 1989, **SAC-7**, pp. 968–973
2 WEI, L.-F.: 'Trellis-cpded modulation with multidimensional constellations', *IEEE Trans.*, 1987, **IT-33**, pp. 483–501
3 FORNEY JR., G.D., GALLAGER, R.G., and QUERSHI, S.U.: 'Efficient modulation for band-limited channels', *IEEE Sel. Areas Commun.*, 1984, **SAC-2**, pp. 632–647
4 WILLIAMS, R.G.C., DELIOT, E., and TAIT, D.J.: 'Practical constellation shaping'. Int. Symp. on Comm. Theory and App., 1993, (Amleside)
5 WILLIAMS, R.G.C.: 'Impulse noise protection with 4-D TCM', *Electron. Lett.*, 1993, **29**, pp. 121–122

Wavelet transform in scattering data interpolation

M.-H. Yaou and W.-T. Chang

Indexing terms: Interpolation, Wavelet transforms

A fast algorithm for scattering data interpolation is presented. Based on the multiresolution wavelet transform, a preconditioning scheme is proposed to expedite the slow computation speed in the interpolation problem. By applying the wavelet transform before and after any conventional iterative solving method, fast data interpolation can be easily achieved.

Introduction: Scattering data interpolation is used to recover a full signal representation when only partial information of the signal is available. This problem plays an important role in many early vision processes such as surface from contours, structure from motion, stereopsis etc. [1]. This is an ill-posed inverse problem and is often described as a regularisation problem. In general, a variation of the functional method which involves a second order maximum smoothness requirement [2, 3] is applied for the regularisation. Various discretisation methods [2, 3] can be used to discretise the problem into an objective function of discrete nodal variables such that an approximated solution can be solved numerically. This discrete formulation then leads to the minimisation of a quadratic energy function

$$\frac{1}{2}\boldsymbol{v}^T\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}^T\boldsymbol{b} + c \qquad (1)$$

where $\boldsymbol{v}$ is a column vector containing the nodal variables to be solved and $\boldsymbol{A}$ is a real symmetric matrix called the stiffness matrix. $\boldsymbol{b}$ and c are the associated column vector and constant. According to the Euler-Lagrange formula, the optimisation of this quadratic function results in a linear equation system $\boldsymbol{Av} - \boldsymbol{b} = 0$. For an $N \times N$ interpolation problem, the size of matrix $\boldsymbol{A}$ will be $N^2 \times N^2$. The resultant equation system is thus usually large and sparse. To solve this problem, iterative methods are usually adopted. How-