DESIGN AND IMPLEMENTATION OF AN EFFICIENT POINT SLICING ALGORITHM FOR THE V.34 MODEM STANDARD

Oğuz Tanrıkulu

DSP Software Engineering (DSPSE),
Inc., 175 Middlesex Turnpike,
Bedford, MA 01730, USA.
Email: OGUZ@dspse.com

ABSTRACT

The V.34 modem standard uses QAM constellations with 4 to 1664 points depending upon the results of the line probing where the telephone channel is characterized to connect at the maximum possible data rate. In the receiver, the decisions errors due to the channel noise are prevented by using the Viterbi subset decoder which uses a point slicing algorithm that determines the closest valid point on the constellation for a given constellation size. In this paper we present a point slicing algorithm where the complexity is nearly independent of the QAM constellation size.

1. INTRODUCTION

The data rates supported by the V.34 modem standard [1] vary between 2.4 kbps to 33.6kbps. Such variable data rates are made possible by using advanced signal processing techniques such as channel coding using 4D convolutional codes [6], shell mapping [2], precoding [3], nonlinear encoding [4] and pre-emphasis filtering at the transmitter [4].

The 4D convolutional codes yield a coding gain above 4.2dB, and the associated Viterbi subset decoder has reduced complexity due to the lower number of state transitions between the states. Shell mapping is a technique for non-equiprobable signalling that reduces the transmitted signal power and thereby yielding a coding gain of 0.7-0.9dB. The precoder implements decision feedback equalization at the transmitter in a way that creates a valid signal for the Viterbi subset decoder at the receiver. Finally, an intelligent choice of the pre-emphasis filter reduces the transmitted signal power at the low frequencies and hence, the level of nonlinear harmonics in the network echo for a higher level of cancellation.

In the receiver, after demodulation, the baseband signal is subjected to timing recovery, carrier recovery, fractionally-spaced equalization, followed by the viterbi algorithm which decodes the 4D codes of Wei [6]. The Viterbi subset decoder is the most computationally expensive block in the receiver and the complexity depends on the 16, 32, or 64 state codes being used. The Viterbi subset decoder (and the equalizer in the self-learning mode) require the knowledge of the closest hard point on the constellation. When precoding is enabled, there is constellation expansion and the point slicing algorithm, which yields the closest hard point on the constellation assumes that the hard points lie on the infinite lattice [5, 7]. We call the algorithm for this case the Infinite Point Slicing (IPS) algorithm which has a simple implementation. When the constellation expansion is high compared to the constellation size, the distance between the constellation points is effectively reduced in order to maintain the constant transmitted signal power. Therefore, the precoding is disabled in this case, and consequently, the slicing algorithm must yield hard points from the particular finite size constellation used in the transmitter of the remote modem.

In this paper, we present an efficient point slicing algorithm for digital transmission based on QAM constellations. Upon receiving noisy constellation points at the equalizer output, such as shown in Fig. 1 for the V.34 modem, the proposed algorihm obtains the closest valid constellation point in each subset. The computational complexity of the proposed algorithm is nearly independent of the constellation size. The new finite point slicing (FPS) algorithm is based on modifying the IPS algorithm, and conducting a search over a maximum of 9 candidate hard points, when necessary.

2. IPS ALGORITHM

For V.34, the infinite lattice Z^2 is partitioned into four subsets S_i, $i = 0, \ldots, 3$, such as shown in Fig. 2. The

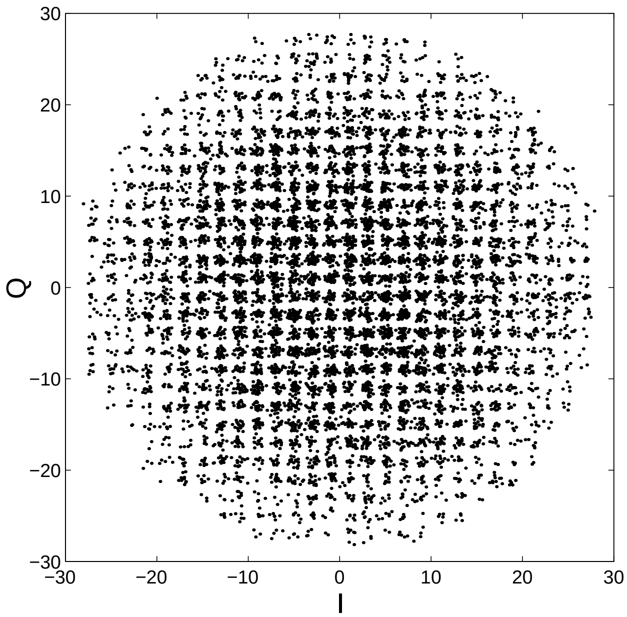

Figure 1: Received signal constellation diagram for 28.8kbps, 3429 symbs/sec V.34 modem at the equalizer output at 31dB SNR.

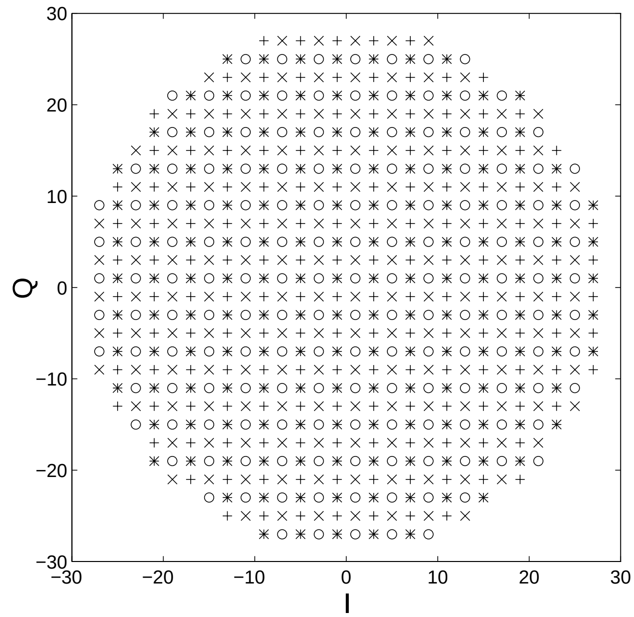

Figure 2: Subsets S_0; 'o', S_1; '+', S_2; 'x' and S_3; '*' the 28,8 kbps V.34 modem.

subsets, S_1, S_2, S_3, are 90^o, 180^o, 270^o clockwise rotations of S_0 as defined by the ITU [6, 1].

For a given the soft (noisy) point (I_s, Q_s), assuming an infinite size constellation, the IPS algorithm determines the closest hard point $(I_h, Q_h) \in S_0$ is as follows;

Step 1. Subtract the offset (1,1) from (I_s, Q_s).

*Step 2.*Round the result of Step 1 to the nearest integer multiple of 4.

*Step 3.*Shift the result of Step 4 back by (1,1) to obtain (I_h, Q_h).

The closest hard points in other subsets can be determined in various ways. For instance, soft points can be rotated counter clockwise onto S_0. This is followed by determining the closest hard point in $(I_h, Q_h) \in S_0$. Finally, this point is rotated back onto the appropriate subset by rotating clockwise. Another method is as follows; once (I_h, Q_h) is determined, the a small search can be conducted over the neighbouring valid hard points that are in other subsets. Hence the key is to determine $(I_h, Q_h) \in S_0$, and we will focus on this problem. Note that, in other modem standards such as in the V.32, the subsets may not be rotationally invariant. However, the proposed algorithm is general enough and can be used in these cases with minor modifications such as keeping a record of the constellation points lying on the subset boundary for each subset. The details of the proposed algorithm are presented below.

3. FPS ALGORITHM

The IPS is a computationally efficient algorithm, however, it generates erroneous hard points when it is necessary to consider the constellation boundary. The necessity for such an algorithm is stated in [6]. Let us denote the finite subsets of interest by $S_i^f \subset S_i$. In the V.34 modem standard [1], the constellation points in S_0 are generated recursively as follows; "The point with the smallest magnitude is labelled as 0. The point with the next larger magnitude is labelled as 1, and so on. When two or more points have the same magnitude, the point with the largest imaginary component is taken first" [1]. It is thus easy to find the set of constellation points with the largest distance, $d_{\max}$, to the origin for any given constellation size. Let the set of these points be

$$\mathcal{D}_{\max} = \{(I,Q) : (I,Q) \in S_0, d(I,Q) = d_{\max}\}$$

then we can write

$$S_0^f = \{(I,Q) : (I,Q) \in S_0, d(I,Q) \leq d_{\max}\}$$

The FPS algorithm determines the closest element in S_0^f as follows. Let (I_h, Q_h) be the hard points obtained from the IPS algorithm;

Step 1. If

$$d(I_s, Q_s) \leq d_{\max} \text{and} (I_h, Q_h) \in S_0^f$$

then, no further operations are necessary.

Step 2. If

$$d(I_s, Q_s) \leq d_{\max} \text{but} d(I_h, Q_h) \geq d_{\max}$$

then (I_h, Q_h) is lying on or immediately outside the constellation boundary. Firstly, find the valid hard points around and including (I_h, Q_h) that are in S_0^f by checking against the elements in $\mathcal{D}_{\max}$. Then, perform a search over these points to find the closest valid hard point.

Step 3. If

$$d(I_s, Q_s) > d_{\max}$$

we find closest hard point in S_0^f as follows; First project (I_s, Q_s) along the line connecting to the origin and get an intermediate soft point (I_s^p, Q_s^p) such that

$$(I_s^p, Q_s^p) = \frac{d_{\max} - \epsilon}{d(I_s, Q_s)} \times (I_s, Q_s)$$

where ϵ is a small positive real number. This projection is such that $d(I_s^p, Q_s^p) \in S_0^f$. Then we use the IPS algorithm to find the hard point (I_h^p, Q_h^p). We may have $d(I_h^p, Q_h^p) <$ or $=$ or $> d_{\max}$. Note also that (I_h^p, Q_h^p) is not necessarily the closest point to (I_s, Q_s) or a valid point in S_0^f. Therefore, we find the closest point we are looking for by using a search over 9 points around, and including, (I_h^p, Q_h^p) and by constraining ourselves to the valid points in S_0^f only. The operations performed for this case are illustrated in Fig. 3.

Examples for various cases above are illustrated in Fig. 4 for S_0^f corresponding to 256 point V.34 constellation. In each case the result of the FPS algorithm and the soft point are connected by a line. The result given by the IPS algorithm is marked by '*'. Note how the IPS algorithm fails to slice the soft points onto valid subset points.

4. COMPLEXITY

The computational complexities of the IPS and FPS algorithms for slicing soft points into S_0^f are compared for the V.34 modem standard for various constellation sizes and at a number of SNR levels. The average number of floating-point operations (FLOPS) performed per second are tabulated in Table 1. The complexity of the FPS algorithm is about twice the complexity of the IPS algorithm and it is nearly independent of the constellation size. Note that the probability of having soft points outside the constellation boundary is smaller for larger constellations, and hence the algorithm works as the IPS algorithm for most soft points. The IPS algorithm has lower complexity but this algorithm makes erroneous decisions on the closest hard point when the soft points are outside the constellation boundary. The computational complexity of using exhaustive search over the constellation is also included. It is clear that this approach is not feasible for large constellations that are necessary for high data rates.

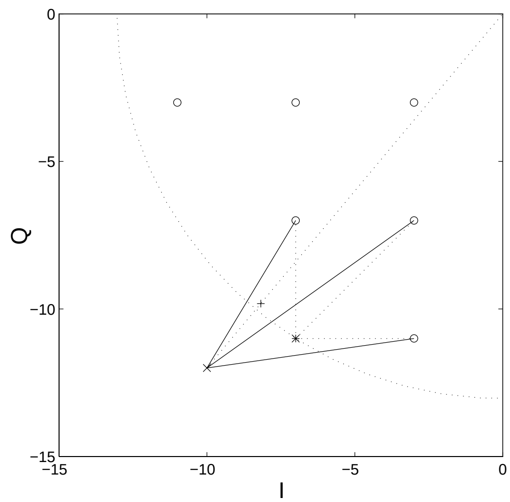

Figure 3: (I_s, Q_s) and (I_s^p, Q_s^p) are respectively denoted by 'x' and '+'. On applying IPS we get (I_h^p, Q_h^p) denoted by '*' which is on the constellation boundary but it is not valid. A local search over three valid points is then conducted.

5. CONCLUSIONS

A computationally efficient algorithm is presented for slicing soft points into finite subsets, which is general enough to be used for all constellation configurations in the V.34 modem standard. The complexity of the proposed algorithm is about twice the complexity of the algorithm that assumes infinite subsets.

6. REFERENCES

[1] ITU Recommendation V.34, "A modem operating at data signalling rates of up to 33 600 bit/s for

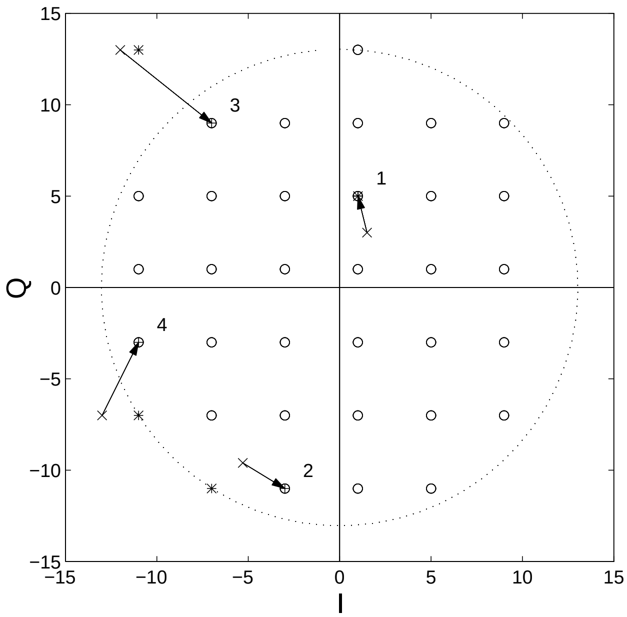

Figure 4: Various examples on the operation of the FPS algorithm.

SNR	20dB	25dB	30dB		
Const. Size	FPS	FPS	FPS	IPS	ES
4	0.17	0.17	0.16	0.04	0.02
32	0.09	0.09	0.09	0.04	0.16
56	0.07	0.07	0.07	0.04	0.29
96	0.08	0.08	0.07	0.04	0.49
144	0.09	0.08	0.08	0.04	0.74
240	0.09	0.08	0.08	0.04	1.23
284	0.09	0.08	0.07	0.04	1.46
416	0.09	0.08	0.07	0.04	2.14
640	0.09	0.08	0.07	0.04	3.29
960	0.09	0.08	0.07	0.04	4.94
1088	0.09	0.08	0.07	0.04	5.60
1664	0.08	0.08	0.07	0.04	8.56

Table 1: Computational complexities of the IPS and FPS algorithms and Exhaustive Search (ES) in MFLOPS for various constellation sizes and Signal-to-Noise Ratios (SNRs).

use on the general switched telephone network and on leased point-to-point 2-wire telephone-type circuits", Geneva, Oct. 9-18, 1996.

[2] R. Laroia, N. Farvardin, S. A. Tretter, "On Optimal Shaping of Multidimensional Constellations", IEEE Transactions on Information Theory, vol. IT-40, no. 4, July 1994.

[3] R. Laroia, "Coding for Intersymbol Interference Channels - Combined Coding and Precoding" IEEE Transactions on Information Theory, vol. IT-4, no. 4, July 1996.

[4] W. L. Betts, E. S. Zuranski, "Method and Apparatus for Adaptively Providing Precoding and Preemphasis conditioning to Signal Data for Transfer over a Communication Channel" US Patent No. 5396519, Mar. 1995.

[5] M. V. Eyuboglu, "Flexible Precoding for V. FAST", Int. Conf. on Data Transmission - Advances in Modem and ISDN Technology and Applications, 1992, Vol.356, Ch.26, No.356, pp.13-18.

[6] L. F. Wei, "Trellis-Coded Modulation with Multidimensional Constellations", IEEE Transactions on Information Theory, vol. IT-33, no. 4, July 1987.

[7] G. D. Forney, Jr., "Advances in Modem Technology since V.32/V.32bis", Int. Conf. on Data Transmission - Advances in Modem and ISDN Technology and Applications, 1992, Vol.356, Ch.26, No.356, pp.1-6.

[8] G. D. Forney, Jr., L. Brown, M. V. Eyuboglu, J. L. Moran, Motorola, Inc., "The V.34 High-Speed Modem Standard", IEEE Communications Magazine, pp. 28-33, Dec.1996.