Trellis Decoding of Permutation Modulations

Emanuele Viterbo[1]
Dip. Elettronica – Politecnico di Torino
C.so Duca degli Abruzzi 24, Torino, Italy
e-mail: viterbo@polito.it

Permutation modulations are particular spherical codes proposed by David Slepian in 1965 [1]. Good permutation modulations may be designed by appropriately selecting the initial vector and may be very efficiently decoded in the AWGN channel and slow fading channel by essentially applying a sorting algorithm to the received signal vector [1]. In this paper we present a new trellis based decoder for ML detection over fast fading channels.

Let $\{\mu_1, \dots, \mu_k\}$ be a set of distinct real numbers with $0 \leq \mu_1 < \mu_2 < \cdots < \mu_k$ and let $\{m_1, \dots, m_k\}$ be a set of positive integers such that $n = \sum_{j=1}^k m_j$. Consider the initial vector with components sorted in ascending order

$$\mathbf{x}_0 = (\underbrace{\mu_1, \dots, \mu_1}_{m_1}, \underbrace{\mu_2, \dots, \mu_2}_{m_2}, \cdots, \underbrace{\mu_k, \dots, \mu_k}_{m_k}) . \quad (1)$$

A Variant I permutation modulation consists of the set $\mathcal{S}$ of vectors obtained by permuting the components of the initial vector $\mathbf{x}_0$. The total number of codewords in such a code is $M_\mathrm{I} = n!/(m_1!\,m_2! \cdots m_k!)$.

Digital transmission over the band-limited fading channel is commonly modeled in the n-dimensional Euclidean space $\mathbf{R}^n$ as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \quad (2)$$

where $\mathbf{y}$ is the received signal vector, $\mathbf{x} = (x_1, \dots, x_n)$ is the transmitted signal vector (or codeword) taken from a finite signal set (or codebook) $\mathcal{S}$, $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \frac{N_0}{2}\mathbf{I})$ is a real Gaussian random vector with i.i.d. components and $\mathbf{H} = \mathrm{diag}(h_1, \dots, h_n)$ is the diagonal channel fading matrix. We distinguish two cases: i) *slow fading*, where the fading matrix is $\mathbf{H} = h\mathbf{I}$ and h changes independently from one codeword to the next; ii) *fast fading*, where the fading coefficients h_i change independently in each component. Finally, we note that when $\mathbf{H} = \mathbf{I}$ is constant, we fall back into the AWGN channel model.

For AWGN and slow fading a simple decoding algorithm is available [1], but in the case of fast fading we need to resort to a new trellis description of permutation codes and perform Viterbi decoding.

The trellis of the permutation code is a multidimensional finite length trellis. It is constructed by considering a finite subset of the k-dimensional integer lattice $\mathbf{Z}^k$, i.e., the set of points inside the parallelepiped of edge lengths $m_1, \dots, m_k$:

$$\mathcal{T} = \{(s_1, \dots, s_k) \in \mathbf{Z}^k \mid 0 \leq s_1 \leq m_1, \cdots, 0 \leq s_k \leq m_k\} \quad (3)$$

A point in this set corresponds to a *state* in the trellis and will be labelled by the integer vector $(s_1, \dots, s_k)$. There are a total of $\prod_{l=1}^k (m_l + 1)$ states throughout the trellis. We define the *initial state* as the one labelled by $(0, \dots, 0)$ and the *final state* as the one labelled by $(m_1, \dots, m_k)$.

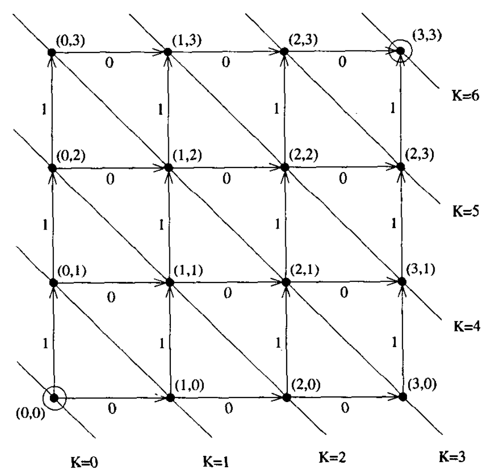
The trellis *branches* are all the segments of length one connecting the states. We can see this trellis as a multidimensional grid. We label with μ_l all trellis branches connecting state $(s_1, \dots, s_l, \dots, s_k)$ to state $(s_1, \dots, s_l + 1, \dots, s_k)$, i.e., all branches pointing in the same direction have the same label taken from the initial vector of the permutation code.

The codewords correspond to paths on this trellis that stem from the initial state and end into the final state, without ever going backwards. The apparently hard problem of counting the number of such paths on the grid becomes trivial, since it corresponds to the total number of codewords $M = n/\prod_{l=1}^k (m_l!)$ of the permutation modulation. This is obvious if we note that the paths have length $\sum_{l=1}^k m_l = n$ and all contain exactly m_l branches labelled with μ_l for $l = 1, \dots, n$ branches in each dimension.

In order to define the trellis sections we define the following state subsets: $\mathcal{T}^{(K)} = \{(s_1, \dots, s_k) \in \mathcal{T} \mid \sum_{l=1}^k s_l = K\}$, $K = 0, \dots, n$. The trellis sections $\mathcal{T}^{(K)}$ contain the states (grid points) laying on the hyperplanes $\sum_{l=1}^k s_l = K$ for $K = 0, \dots, n$. All states in one section $\mathcal{T}^{(K)}$ are connected to the ones in the following section $\mathcal{T}^{(K+1)}$ by a variable number of branches ranging from 1 to k, according to the position within the grid (the points along the borders have less connections).

Example – Let $k = 2$, $m_1 = 3$ and $m_2 = 3$, $\mu_1 = 0$ and $\mu_2 = 1$, then $n = 6$ and $M = 20$. The trellis is given below. It can be seen that the number of states in each section are $|\mathcal{T}^{(0)}| = 1$, $|\mathcal{T}^{(1)}| = 2$, $|\mathcal{T}^{(2)}| = 3$, $|\mathcal{T}^{(3)}| = 4$, $|\mathcal{T}^{(4)}| = 3$, $|\mathcal{T}^{(5)}| = 2$, $|\mathcal{T}^{(6)}| = 1$.

In general we have a multidimensional trellis over which it is possible to perform the soft Viterbi ML decoding algorithm. Given the received word $\mathbf{y}$ as in (2), we compute the branch metrics $\lambda_l = (h_l\mu_l - y_l)^2$ and use them to label to branches pointing along the l-th dimension for $l = 1, \dots, k$.



[1]This work was partly supported by CERCOM.

References

[1] D. Slepian, Permutation modulation, *Proceedings of the IEEE*, March 1965, pp. 228–236.

0-7803-7728-1/03/$17.00 ©2003 IEEE.