

Solution to text problem 4.13 by Ankit Gupta

Statement of the problem

Consider the general setup of (nonlinear) nonantipodal modulation in which user k transmits

$$s_k(t; u)$$

in order to send symbol $u \in [u_1, \dots, u_{n_k}]$

(a) Show that the collection of observables

$$y_{j,u} = \int_{-\infty}^{\infty} v_j(t; u) y(t) dt \quad (1)$$

with

$$v_{k+iK}(t; u) = s_k(t - iT - \tau_k; u) \quad (2)$$

are sufficient statistics

(b) Generalize the metric 4.32 for a dynamic programming solution to maximum likelihood asynchronous multiuser detection.

(c) Find the time complexity per bit of the algorithm in (b) assuming $n_k = m$

(d) Does the time complexity per bit found in (c) decrease if the modulation is linear

Solution

Part a

First we convert the asynchronous problem to a synchronous one by considering each m -ary bit from a different user. The maximum likelihood estimate in this problem is a special case of the hypothesis testing problem with the hypothesis being one of

$$\sum_{j=1-MK}^{MK+K} A_{k(j)} v_j(t; u) + \sigma n(t) \quad (3)$$

where

$$v_{k+iK}(t; u) = s_k(t - iT - \tau_k; u) \quad (4)$$

and

$$A_{k(j)} = A_{(j \bmod K)} \quad (5)$$

and $u \in [u_1, \dots, u_{n_k}]$. The hypothesis are distinguished by different choices of u in the expression 3. The number of hypothesis is therefore $m^{2MK+K-1}$

where $m = n_k$. According to proposition (3.2) in the textbook. We declare the hypothesis that maximizes the following expression.

$$\max \exp \left(-\frac{1}{2\sigma^2} \int_{(1-MK)T}^{(MK+K)T} \left(y(t) - \sum_{j=1-MK}^{MK+K} A_{k(j)} v_j(t; u) \right)^2 \right) \quad (6)$$

Now since the exponential is an increasing function we may as well maximize the argument. The argument can be further subdivided as

$$\begin{aligned} \Omega(b) &= - \int_{(1-MK)T}^{(MK+K)T} (y(t))^2 \\ &+ - \int_{(1-MK)T}^{(MK+K)T} \left(\sum_{j=1-MK}^{MK+K} A_{k(j)} v_j(t; u) \right)^2 \\ &+ 2 \int_{(1-MK)T}^{(MK+K)T} y(t) \sum_{j=1-MK}^{MK+K} A_{k(j)} v_j(t; u) \end{aligned} \quad (7)$$

dropping the first term as being common to all hypothesis we have the parameter to be maximized as the following.

$$\Omega(b) = 2 \sum_{j=1-MK}^{MK+K} A_{k(j)} y_{j,u} - \sum_{j=1-MK}^{MK+K} \sum_{l=1-MK}^{MK+K} A_{k(j)} A_{k(l)} \rho_{j,l}(u_j, u_l) \quad (8)$$

Note that $\Omega(b)$ depends on $y(t)$ only through $y_{j,u}$ and hence the collection of $y_{j,u}$ form a sufficient statistic. Also note that $\rho_{j,l}(u_j, u_l) = \rho_{l,j}(u_l, u_j)$ because in our synchronous problem they are defined as the following

$$\rho_{j,l}(u_j, u_l) = \int_{-\infty}^{\infty} v_j(t, u_j) v_l(t, u_l) \quad (9)$$

Part b

The equation 8 may be further written as.

$$\Omega(b) = \sum_{j=1-MK}^{MK+K} \lambda_j(x_j, u_j) \quad (10)$$

where

$$\lambda_j(x_j, u_j) = 2A_{k(j)} y_{j,u_j} + |A_{k(j)} v_j(t; u)|^2 + 2A_{k(j)} \sum_{i=j-1}^{j-m} A_{k(i)} \rho_{i,j}(u_i, u_j) \quad (11)$$

Where 11 was written using the fact that the symbol in a particular interval only overlaps with 2m-2 symbols. And thus the $\rho_{i,j}(u_i, u_j)$ is non zero for only

those values. Also note that the vector x_j appears on the right hand side implicitly due to the components $i=j-1$ to $j-m$. The dynamic programming algorithm can now be inferred from the form of $\lambda_j(x_j, u_j)$. There are now m^{n-1} nodes at each stage corresponding to each state that the vector x_j can take. The incoming u_j shifts the shift register x_j and depending on the value of y_j there are m computations that are performed. The other values in the equation can be precomputed and stored except for the y_j term and hence there are m computations for each stage of the trellis.

Part c

In each stage of the trellis each computation corresponds to resolving of $\log_2 m$ bits. Therefore the complexity per bit is found to be

$$Complexity = \frac{m^n}{\log_2 m} \quad (12)$$

Part d

For the binary case the complexity per bit is

$$Complexity = 2^K \quad (13)$$

comparing with equation (12) it is obvious that the complexity is less in the binary case. In case of linear antipodal the complexity is lower. However for the case of PAM m -ary linear modulation there are still m^{n-1} nodes at each stage of the trellis and the computation involving y_j has to be performed for each m -ary incoming bit. This implies that linearity does not change the complexity per bit.