COMMUNICATION SCIENCES INSTITUTE

Two Problems in Sequence Design

by

Carlos Corrada-Bravo

CSI-02-12-01

USC VITERBI SCHOOL OF ENGINEERING
UNIVERSITY OF SOUTHERN CALIFORNIA
ELECTRICAL ENGINEERING — SYSTEMS
LOS ANGELES, CA 90089-2565

Two Problems in Sequence Design

by

Carlos J. Corrada-Bravo

A Dissertation Presented to the
FACULTY OF THE GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(Electrical Engineering)

June 2002

Contents

List Of Tables

List Of Figures

Chapter 1

PN-Sequences with Good Correlation Properties and Approximately Flat PSD for UWB Applications

1.1 Introduction

There has been tremendous interest in wideband communications systems for personal and cellular communications. Wideband systems offer several advantages over narrowband systems. Some of the attractive features are:

- Wideband systems provide superior ability to operate against several forms of interference, such as multi-path, multi-user interference, and narrowband interference;
- Wideband systems allow considerable flexibility in the number of assigned users for a given channel;
- Wideband systems have the ability to operate in an electromagnetic spectrum that is already occupied by other narrowband users without degrading the performance of these existing users;
- Implementation costs of wideband systems have been substantially reduced by advances in electronic communications technology; etc.

Using these signal features, we can think of various potential applications. One of them is an accurate ranging in a dense multi-path environment. Resolvable multi-path enables precise detection of the direct path, which enables ranging with high accuracy. Penetration capability is expected to allow a beyond line of sight ranging. Another possible application is high data rate indoor wireless LAN. By employing a rake type receiver, more signal energy can be capture from multi-path signals. Also, great targets for this technology are applications where both communication and positioning are required, like what has been called easy living or intelligent spaces. In this application the spaces are expected to react accordingly with the needs of the user based on their position and preprogrammed tasks; for example the desire to print to the closest or fastest printer.

The current emphasis in wideband systems has been on constant-envelope spread spectrum modulations. Unfortunately, this ignores one design with considerable potential, namely time-hopping. The technology for generating and receiving pulses on the order of a nanosecond or less in width, with a shape similar to one cycle of a sine wave, is currently available. These monocycles can be received by correlation

detection virtually at the antenna terminals, making a relatively low cost receiver possible.

Ultra-wideband radio is a communication system whose 3dB signal bandwidth is greater than 25% of its center frequency. UWB radio communicates with sub-nanosecond time modulated narrow pulses without a carrier. A typical signal bandwidth of UWB radio system is around 1 and 2 GHz. This kind of signal has already been used in the radar community, mostly for ground penetration. For this application, transmit power was not a critical issue since the signal was not transmitted into the communication channel. Now that new communication schemes using UWB signal has been introduced, UWB radios involve a risk of interfering with other narrowband systems due to its wide bandwidth. To regulate this new technology the FCC issued a Notice of Proposed Rule Making (NPRM), which means a modification to FCC Part15.

In order to ensure no interference with other narrowband users the power spectrum of the system must be kept under the noise floor of these other users. Our objective here is to achieve as flat a transmitted power spectral density as possible, and hence to put as small an amount of power in other systems' allocated bandwidths as possible. We have a pulse which is approximately one nanosecond in duration, as its derivative is the received pulse waveform (after the antenna processing). Trains of pulses will be constructed analytically by shifting this waveform. In our signal, the selection of the transmission time in each frame is not random. The selected transmission times have to follow a rigorous design that will provide the flatness in the spectrum that we desire as well as to randomize each users signal with respect to all other users.

We will describe four designs, based on coincidence correlation for doubly periodic arrays. As we shall see, there is in fact a mathematical coincidence-correlation design for matrices that is nearly optimal. We expect this approach to yield sets of sequences with very good auto- and cross-correlation properties that will follow a low PSD level.

In Section 1.2 we will describe the UWD system and in Section 1.3 we start with a detailed description of the constructions. Section 1.4 contains the calculation of the PSD of some of the designs and finally Section 1.5 contain the preliminary results and future research.

1.2 UWB and Time Hopping

1.2.1 Introduction to UWB Radio

When the 3 dB bandwidth of a radio signal becomes 25% or more of the signal's center frequency, most agree that this radio should be called *ultra-wideband* (UWB). The combination of a relatively large bandwidth at a relatively low center frequency provides two kinds of dividends. First, of all radios at the same center frequency, an ultra-wideband radio should provide the finest time resolution in a well-designed receiver, and hence have potential advantages in ranging and multi-path mitigation. Second, of all radios with the same bandwidth, ultra-wideband radios operate in the lowest frequency bands and hence have the best chance to propagate through most materials.

One major problem that UWB radios must solve is the satisfactory coexistence of UWB radio signals with the myriad of other narrowband and wideband signals with which they must simultaneously share their frequency bands. This implies that UWB radios should employ spread-spectrum methods to protect them against the interference that they will inevitably encounter from other radio systems. It also implies that UWB radios can only radiate small amounts of power in each of the narrow frequency bands of other radio systems to avoid interfering with them. This latter issue is a matter for the appropriate regulatory entity to oversee.

Regulation of UWB radio currently is being considered in the United States, and may take the form of an upper bound on the radiated power spectral density of the UWB system. Then the efficiency and performance of the UWB system will depend to a great extent on the flatness and frequency extent of its radiated power spectral density.

The technology used to implement UWB radio depends to a great extent on the frequency band in which the radio must operate. Modulations in UWB radios usually are constructed from trains of very short pulses whose width often is in the range of a few nanoseconds to fractions of a nanosecond, giving bandwidths on the order of gigahertz. Since the ability of an antenna to radiate efficiently decreases as frequency approaches zero, the pulse shapes are generally chosen to have little or no energy content as frequency approaches zero. Hence pulses tend to have balanced

positive and negative excursions, e.g., one period of a sinusoid, or the derivative of a Gaussian pulse.

Modulation formats vary, but generally transmit several pulses per data bit and use coherent detection of the pulse train. The radios usually produce carrier-less signals and do not use mixers for the purpose of changing the frequency band of a signal. Some systems resemble baseband direct-sequence spread-spectrum systems and others resemble baseband time-hopped spread-spectrum signals. Digital modulation of these kinds of signals is accomplished by added time shifting or polarity reversal.

1.2.2 Time-Hopping Signal Models

Time hopping for spectral spreading may provide implementation advantages and may be desirable in ranging systems because it may be easier to find the leading edge of an isolated received pulse signal. One possible form of an unmodulated time-hopping ultra-wideband signal generator for the i^{th} time-hopped signal is of the form

$$s^{(i)}(t) = \sum_j p(t - jT_{\text{f}} - c_j^{(i)} T_{\text{c}}) = \sum_n a_n^{(i)} p(t - nT_{\text{c}}) \,, \tag{1.1}$$

where we assume for simplicity that one frame time T_{f} is composed of N_{f} such slots, i.e., $N_{\text{f}} T_{\text{c}} = T_{\text{f}}$. The integer time-hopping code $\{c_j^{(i)}\}$, $0 \le c_j^{(i)} < N_{\text{f}}$, has period N. Then the quantity $a_n^{(i)}$ is defined as

$$a_n^{(i)} = \begin{cases} 1 & \text{if there is an integer } j \text{ such that } n = jN_{\text{f}} + c_j^{(i)} \,, \\ 0 & \text{otherwise.} \end{cases} \tag{1.2}$$

A signal then is described totally by which time slots are occupied and which slots are not.

As it was shown in [42], the normalized periodic correlation between the signals of users i and j is

$$\tilde{R}_{ij}(n_\tau T_{\text{c}}) \triangleq \frac{R_{ij}(n_\tau T_{\text{c}})}{R_{ii}(0)} = \frac{1}{N} \underbrace{\sum_{n=0}^{NN_{\text{f}}-1} a_n^{(i)} a_{n \ominus n_\tau}^{(j)}}_{} \,. \tag{1.3}$$

This equation displays the normalized periodic coincidence correlation between the two transmitted signals when one is shifted by an integer number n_τ of slot widths relative to the other. The quantity N is the period of the time-hopping sequence design measured in frame times, and hence the period of any of the binary sequences $\{a_n^{(i)}\}$ is NN_f, i.e., $a_n^{(i)} = a_{n+NN_\mathrm{f}}^{(i)}$ for all n and i. The notation $\ominus$ in (1.3) denotes subtraction modulo NN_f, leading to the periodic nature of the computation. The peak value 1 occurs when $i = j$ and $n_\tau = 0$.

As structured here, the design requires that one and only one slot in each frame be occupied by a pulse. Hence $a_n^{(i)} = 0$ for all but one value of n in each range $jN_\mathrm{f} \le n < (j+1)N_\mathrm{f}$ for each value of j. Let's display this graphically in an $N_\mathrm{f} \times N$ matrix by mapping the sequence $\{a_n^{(i)}\}$ into the matrix $\mathbf{A}^{(i)}$ as follows.

$$a_0^{(i)}, a_1^{(i)}, a_2^{(i)}, \ldots, a_{NN_\mathrm{f}-1}^{(i)} \iff \begin{bmatrix} a_0^{(i)} & a_{N_\mathrm{f}}^{(i)} & a_{2N_\mathrm{f}}^{(i)} & \cdots & a_{(N-1)N_\mathrm{f}}^{(i)} \\ a_1^{(i)} & a_{N_\mathrm{f}+1}^{(i)} & a_{2N_\mathrm{f}+1}^{(i)} & \cdots & a_{(N-1)N_\mathrm{f}+1}^{(i)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N_\mathrm{f}-1}^{(i)} & a_{2N_\mathrm{f}-1}^{(i)} & a_{3N_\mathrm{f}-1}^{(i)} & \cdots & a_{NN_\mathrm{f}-1}^{(i)} \end{bmatrix} \triangleq \mathbf{A}^{(i)} \tag{1.4}$$

That is, one period of the sequence is entered into the matrix $\mathbf{A}^{(i)}$, filling in order the first column, the second column, etc. The constraint of one pulse per frame time translates to exactly one 1 entry and $N_\mathrm{f} - 1$ entries which are 0 in each column of the matrix $\mathbf{A}^{(i)}$.

Redefining the elements of the matrix by setting $b_{jk}^{(i)} = a_{kN_\mathrm{f}+j}$ where $0 \le N_\mathrm{f} < N_\mathrm{f} - 1$ and $0 \le k < N$,

$$\mathbf{A}^{(i)} \triangleq \begin{bmatrix} b_{00}^{(i)} & b_{01}^{(i)} & b_{02}^{(i)} & \cdots & b_{0,N-1}^{(i)} \\ b_{10}^{(i)} & b_{11}^{(i)} & b_{12}^{(i)} & \cdots & b_{1,N-1}^{(i)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{N_\mathrm{f}-1,0}^{(i)} & b_{N_\mathrm{f}-1,1}^{(i)} & b_{N_\mathrm{f}-1,2}^{(i)} & \cdots & b_{N_\mathrm{f}-1,N-1}^{(i)} \end{bmatrix} \tag{1.5}$$

So for example, the sequence with period 16 and 4 slots per frame, with slots 0, 7, 9, and 14, occupied by pulses is represented by the matrix

$$N_{\mathrm{f}} = N = 4 \text{ and } a_0^{(i)} = a_7^{(i)} = a_9^{(i)} = a_{14}^{(i)} = 1 \iff \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{1.6}$$

the inner product of two of the binary signal matrices, gives the recognizable correlation,

$$\left(\mathbf{A}^{(i)}, \mathbf{A}^{(j)}\right) = \sum_{m=0}^{N_{\mathrm{f}}-1} \sum_{n=0}^{N-1} b_{mn}^{(i)} b_{mn}^{(j)} = \sum_{n=0}^{NN_{\mathrm{f}}-1} a_n^{(i)} a_n^{(j)} = N \cdot \tilde{R}_{ij}(0). \tag{1.7}$$

That is, this matrix inner product is the normalized correlation between signals i and j at zero shift (see (1.3)).

Again, following [42], to accomplish the computation of $\tilde{R}_{ij}(n_\tau T_{\mathrm{c}})$, the elements of $\mathbf{A}^{(j)}$ must be relocated by a "helical" shift to form the matrix

$$\begin{aligned} \mathbb{T}_{n_\tau}\mathbf{A}^{(j)} &= \begin{bmatrix} a_{\ominus n_\tau}^{(j)} & a_{N_{\mathrm{f}}\ominus n_\tau}^{(j)} & a_{2N_{\mathrm{f}}\ominus n_\tau}^{(j)} & \cdots & a_{((N-1)N_{\mathrm{f}})\ominus n_\tau}^{(j)} \\ a_{1\ominus n_\tau}^{(j)} & a_{(N_{\mathrm{f}}+1)\ominus n_\tau}^{(j)} & a_{(2N_{\mathrm{f}}+1)\ominus n_\tau}^{(j)} & \cdots & a_{((N-1)N_{\mathrm{f}}+1)\ominus n_\tau}^{(j)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{(N_{\mathrm{f}}-1)\ominus n_\tau}^{(j)} & a_{(2N_{\mathrm{f}}-1)\ominus n_\tau}^{(j)} & a_{(3N_{\mathrm{f}}-1)\ominus n_\tau}^{(j)} & \cdots & a_{(NN_{\mathrm{f}}-1)\ominus n_\tau}^{(j)} \end{bmatrix} \\ &= \left[\begin{array}{cccccc} b_{N_{\mathrm{f}}-\beta,N-\alpha-1}^{(j)} & \cdots & b_{N_{\mathrm{f}}-\beta,N-1}^{(j)} & b_{N_{\mathrm{f}}-\beta,0}^{(j)} & \cdots & b_{N_{\mathrm{f}}-\beta,N-\alpha-2}^{(j)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{N_{\mathrm{f}}-1,N-\alpha-1}^{(j)} & \cdots & b_{N_{\mathrm{f}}-1,N-1}^{(j)} & b_{N_{\mathrm{f}}-1,0}^{(j)} & \cdots & b_{N_{\mathrm{f}}-1,N-\alpha-2}^{(j)} \\ \hline b_{0,N-\alpha}^{(j)} & \cdots & b_{0,0}^{(j)} & b_{0,1}^{(j)} & \cdots & b_{0,N-\alpha-1}^{(j)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{N_{\mathrm{f}}-\beta-1,N-\alpha}^{(j)} & \cdots & b_{N_{\mathrm{f}}-\beta-1,0}^{(j)} & b_{N_{\mathrm{f}}-\beta-1,1}^{(j)} & \cdots & b_{N_{\mathrm{f}}-\beta-1,N-\alpha-1}^{(j)} \end{array}\right] \end{aligned} \tag{1.8}$$

Here we have used the operator $\mathbb{T}_{n_\tau}$ to represent the effect of moving the entries of the matrix.

Because of the way in which this matrix has been constructed, it follows immediately that

$$\left(\mathbf{A}^{(i)}, \mathbb{T}_{n_\tau}\mathbf{A}^{(j)}\right) = \sum_{n=0}^{NN_{\mathrm{f}}-1} a_n^{(i)} a_{n\ominus n_\tau}^{(j)} = N \cdot \tilde{R}_{ij}(n_\tau T_{\mathrm{c}}). \tag{1.9}$$

Notice that all elements in the same row of $\mathbb{T}_{n_\tau}\mathbf{A}^{(j)}$ have the same row index. On the other hand, $b_{mn}^{(j)}$ elements in the same column of $\mathbb{T}_{n_\tau}\mathbf{A}^{(j)}$ have the same column index only if they are on the same side of the line drawn above the row with index 0 in the last matrix in (1.8). In the special case $\beta = 0$, then all the entries in a column of $\mathbb{T}_{n_\tau}\mathbf{A}^{(j)}$ have the same column index.

1.2.2.1 Characterization of a Doubly Periodic Array Design

Typical mathematical array designs that can be found in the literature are constructed to minimize the doubly periodic correlation properties of the arrays. The doubly periodic correlation computation is slightly different than the calculations of the previous section. Let $\mathbf{A}^{(i)}$ and $\mathbf{A}^{(j)}$ be two matrices as defined in (1.5). The doubly periodic cross-correlation between these two matrices can be defined as

$$P_{\mathbf{A}^{(i)}\mathbf{A}^{(j)}}(n_{\mathrm{r}}, n_{\mathrm{c}}) \triangleq \sum_{m=0}^{N_{\mathrm{f}}-1} \sum_{n=0}^{N-1} b_{m,n}^{(i)} b_{m\ominus n_{\mathrm{r}}, n\ominus n_{\mathrm{c}}}^{(j)} \tag{1.10}$$

for $0 \leq n_{\mathrm{r}} < N_{\mathrm{f}}$ and $0 \leq n_{\mathrm{c}} < N$. Here the computation $\ominus$ is modulo N_{f} in the first subscript and modulo N in the second subscript. To compute this doubly periodic correlation using a matrix inner product, we must transform $\mathbf{A}^{(j)}$ so that its entries are properly positioned to accomplish the computation in (1.10). This can be done using the $N_{\mathrm{f}} \times N_{\mathrm{f}}$ and $N \times N$ permutation matrices. Let's define an $m \times m$ matrix $\mathbf{P}_m$ to be

$$\mathbf{P}_m \triangleq \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} \quad \text{with} \quad \mathbf{P}_m^{-1} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}. \tag{1.11}$$

Multiplication by $\mathbf{P}_m^n$ on the left of a matrix with m rows cyclically permutes the rows of the matrix n positions downward in the matrix. Similarly, multiplication by $\mathbf{P}_m^{-n}$ on the right of a matrix with m columns cyclically permutes the rows of the matrix n positions to the right in the matrix. Using this notation, it follows immediately that

$$\mathbf{P}_{N_\mathrm{f}}^{n_\mathrm{r}}\mathbf{A}^{(j)}\mathbf{P}_N^{-n_\mathrm{c}} = \left[\begin{array}{cccccc} b^{(j)}_{N_\mathrm{f}-n_\mathrm{r},N-n_\mathrm{c}} & \cdots & b^{(j)}_{N_\mathrm{f}-n_\mathrm{r},0} & b^{(j)}_{N_\mathrm{f}-n_\mathrm{r},1} & \cdots & b^{(j)}_{N_\mathrm{f}-n_\mathrm{r},N-n_\mathrm{c}-1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b^{(j)}_{N_\mathrm{f}-1,N-n_\mathrm{c}} & \cdots & b^{(j)}_{N_\mathrm{f}-1,0} & b^{(j)}_{N_\mathrm{f}-1,1} & \cdots & b^{(j)}_{N_\mathrm{f}-1,N-n_\mathrm{c}-1} \\ \hline b^{(j)}_{0,N-n_\mathrm{c}} & \cdots & b^{(j)}_{0,0} & b^{(j)}_{0,1} & \cdots & b^{(j)}_{0,N-n_\mathrm{c}-1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b^{(j)}_{N_\mathrm{f}-n_\mathrm{r}-1,N-n_\mathrm{c}} & \cdots & b^{(j)}_{N_\mathrm{f}-n_\mathrm{r}-1,0} & b^{(j)}_{N_\mathrm{f}-n_\mathrm{r}-1,1} & \cdots & b^{(j)}_{N_\mathrm{f}-n_\mathrm{r}-1,N-n_\mathrm{c}-1} \end{array}\right] \tag{1.12}$$

and it follows that

$$\left(\mathbf{A}^{(i)}, \mathbf{P}_{N_\mathrm{f}}^{n_\mathrm{r}}\mathbf{A}^{(j)}\mathbf{P}_N^{-n_\mathrm{c}}\right) = P_{\mathbf{A}^{(i)}\mathbf{A}^{(j)}}(n_\mathrm{r}, n_\mathrm{c})\,. \tag{1.13}$$

In [47] it was shown that a set of M matrices $\mathbf{A}^{(i)}$, $i = 1, \ldots, M$, with non-matched doubly periodic correlation at most

$$P_{\max}\left(\{\mathbf{A}^{(i)}\}_{i=1}^M\right) \triangleq \max_{\substack{i,\,j,\,n_\mathrm{r},\,n_\mathrm{c}: \\ i \neq j \text{ or } n_\mathrm{r} \neq 0 \text{ or } n_\mathrm{c} \neq 0}} P_{\mathbf{A}^{(i)}\mathbf{A}^{(j)}}(n_\mathrm{r}, n_\mathrm{c})\,, \tag{1.14}$$

can be mapped into a set of M time-hopping sequences (using (1.4)) with the normalized correlation bound

$$\max_{\substack{i,\,j,\,n_\tau: \\ i \neq j \text{ or } n_\tau \neq 0}} \tilde{R}_{ij}(n_\tau T_\mathrm{c}) \leq \frac{2}{N} P_{\max}\left(\{\mathbf{A}^{(i)}\}_{i=1}^M\right)\,. \tag{1.15}$$

When necessary, it is not difficult to adapt this correlation bounding approach to include guard times between pulse frames and other implementation constraints, by inserting mandatory rows of zeros in the $\mathbf{A}^{(i)}$ matrices.

1.3 Sequence Designs

1.3.1 The Johnson Bound

The Johnson bound (see [34] page 327, and [37]) is a bound on the number of constant-weight words that can be achieved in the design of a cyclic code with a prescribed minimum Hamming distance. Viewing our sequences $\{a_n^{(i)}\}_{i=1}^{M}$ and their cyclic shifts as constant weight cyclic code words over the binary field of two elements, we can transform this bound to one on the number of time-hopping sequences that can be designed with a prescribed bound on auto- and cross-correlation.

Clearly, the design objective is to minimize the periodic correlations $N\tilde{R}_{i,j}(n_\tau T_w)$ except when $i = j$ and $n_\tau = 0$ in which case $N\tilde{R}_{i,j}(n_\tau T_w) = N$. Let there be M time-hopping sequences in all. If we impose the condition

$$N\tilde{R}_{i,j}(n_\tau T_w) \;\leq\; \lambda, \text{ when either } i \neq j \text{ or } n_\tau \neq 0, \tag{1.16}$$

by specification of the parameter λ, then the Johnson bound states that

$$M \leq \left\lfloor \frac{1}{N} \left\lfloor \frac{NN_\mathrm{f}-1}{N-1} \cdots \left\lfloor \frac{NN_\mathrm{f}-(\lambda-1)}{N-(\lambda-1)} \left\lfloor \frac{NN_\mathrm{f}-\lambda}{N-\lambda} \right\rfloor \right\rfloor \cdots \right\rfloor \right\rfloor \tag{1.17}$$

where N is the weight of the sequence, NN_f is the sequence period, and $\lfloor a \rfloor$ denotes the largest integer $\leq a$. When both N and N_f are large and λ is small, the bound can be approximated by

$$M \leq \frac{(NN_\mathrm{f})^\lambda}{N^{\lambda+1}} \;=\; \frac{N_\mathrm{f}^\lambda}{N}. \tag{1.18}$$

Note that unless $N_\mathrm{f} >> N$, setting $\lambda = 1$ would result in a small number of time-hopping sequences. Thus the Johnson bound indicates that $\lambda = 2$ is the smallest value for which a multiple-access signal design may be feasible.

1.3.2 Moreno, Zhang, Kumar and Zinoviev Sequences

1.3.2.1 Construction

Our first construction is based on the fact that the number of roots of a polynomial in a finite field is less than or equal to the degree of the polynomial.

Let α and β be primitive in GF(q), with $q = p^m$, p prime. It was shown in [37] that we can construct a family of codes with

$$f(x) = x^2 + x + c$$

where $c \in \mathrm{GF}(q)$. Let define a matrix as:

$$A(i,j) = \begin{cases} 1 & \text{if } \ f(i) = \beta^j \\ 0 & \text{otherwise} \end{cases}$$

then let

$$f(\alpha^i) = \beta^j$$

therefore

$$f(\alpha^i) = \alpha^{2i} + \alpha^i + c$$

or

$$j = \log_\beta(\alpha^{2i} + \alpha^i + c)$$

Then we convert this matrix to a sequence $j_1, \ldots, j_{q-1}$ where j_i is the value of j at the i^{th} position. This construction generates a family of q members with length $q - 1$.

Example. Let $q = 2^3$ with $x^3 = x^2 + 1$ (see Table 1.1) and $\alpha = \beta$ then we can construct the sequences in Table 1.2.

$x = \alpha$
$x^2 = \alpha^2$
$x^3 = \alpha^2 + 1$
$x^4 = \alpha^2 + \alpha + 1$
$x^5 = \alpha + 1$
$x^6 = \alpha^2 + \alpha$
$x^7 = 1$

Table 1.1: Description of the elements of GF(q)

We can see one of the sequences in the example in its matrix form in Figure 1.1.

7	0	0	0	0	0	0	0
6	0	1	1	0	0	0	0
5	1	0	0	0	1	0	0
4	0	0	0	0	0	0	0
3	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
	5	6	6	$*$	5	$*$	3

Figure 1.1: $c = \alpha^2 + 1$

$c \setminus i$	1	2	3	4	5	6	7
0	6	5	5	3	6	3	$*$
1	4	1	1	2	4	2	7
α	2	7	7	4	2	4	1
α^2	1	4	4	7	1	7	2
$\alpha + 1$	3	$*$	$*$	6	3	6	5
$\alpha^2 + 1$	5	6	6	$*$	5	$*$	3
$\alpha^2 + \alpha$	$*$	3	3	5	$*$	5	6
$\alpha^2 + \alpha + 1$	7	2	2	1	7	1	4

Table 1.2: Complete family

As can be seen from this example in Figure 1.1, there are some empty columns (instances of time were no signal is sent), this happens whenever the polynomial $\alpha^{2i} + \alpha^i + c$ has a root. However it is known that close to half of the polynomials of a finite field are irreducible, therefore if it is necessary we can choose only those as the members of our family, obtaining a family of size$\approx (q - 1/2)$ with maximum auto- and cross-correlation 2.

1.3.3 Moreno-Maric sequences

1.3.3.1 Construction

Let F be a finite field with p^m elements, where p is a prime number. Let P be the projective line over F, in other words $P = F \cup \{\infty\}$ where ∞ has the usual properties. If we consider $f(x) = \frac{ax+b}{cx+d}$, where $ad \neq bc$ and $a, b, c, d \in F$, then it is well known that substitution of elements of P in $f(x)$ produces a permutation of the elements of P.

Furthermore if $g(x)$ is another fractional linear transformation, similar to $f(x)$, then there are exactly two values of x in P for which $f(x) = g(x)$. We now invoke the following result of Berlekamp-Moreno [7].

Theorem 1 *Whenever $x^2 + x + \alpha$ is irreducible, and α primitive in F, then the permutation given by $-\frac{\alpha}{x+1}$ gives a cycle of length $p^m + 1$.*

We assume now that the cycle given by $C = -\frac{\alpha}{x+1}$ begins with 0 and ends with ∞, and since $0 \to -\alpha$, it goes $0 \to -\alpha, \ldots, \to \infty$.

Example. With $q = 2^3$ we get the cycle $0 \to \alpha \to \alpha^3 \to \alpha^6 \to \alpha^4 \to \alpha^2 \to \alpha^5 \to 1 \to \infty$.

We can plot the fractional linear transformation $f(x) = \frac{1}{x}$ using the vertical and horizontal ordering of Example 1 to obtain the permutation matrix in Figure 1.2 (Figures 1.2 and 1.3 are the same, only the labels have changed).

Now we can give more familiar labels to our members to obtain something like Figure 1.3.

Note that in general we can plot permutation matrices corresponding to any $f(x)$ using the horizontal and vertical ordering given by $C(x) = -\frac{\alpha}{x+1}$. Also note that

∞	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0
α^5	0	0	0	0	0	1	0	0	0
α^2	0	0	0	0	0	0	1	0	0
α^4	0	0	1	0	0	0	0	0	0
α^6	0	1	0	0	0	0	0	0	0
α^3	0	0	0	0	1	0	0	0	0
α	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1
	0	α	α^3	α^6	α^4	α^2	α^5	1	∞

Figure 1.2: An example of a MM sequence in matrix form

9	1	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	1	0
7	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	1	0	0
5	0	0	1	0	0	0	0	0	0
4	0	1	0	0	0	0	0	0	0
3	0	0	0	0	1	0	0	0	0
2	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	1
	9	4	5	2	3	7	6	8	1

Figure 1.3: Same MM sequence with different labels

$f[C(x)]$ when plotted would give a permutation matrix corresponding to a horizontal periodical right shift of $f(x)$, of one unit with period p^m+1. Similarly $C[f(x)]$ gives us a vertical up, one unit periodical shift. Since composition of fractional linear transformations gives us also fractional linear transformations, by repeating the above process we can obtain periodical shifts of more units.

Now recall that there are exactly two values of x in P for which $f(x) = g(x)$ for any two fractional linear transformations that are different. From this we gather that if we consider the set of all fractional linear transformations obtained from $f(x)$ by the above shifting process (up and right) and if $g(x)$ is not in this set, then the cross-correlation of the arrays given by $f(x)$ and $g(x)$ is two. In fact

Theorem 2 *There are p^m-1 distinct fractional linear transformations giving doubly periodic arrays of length p^m+1, with auto and cross-correlation two.*

1.3.4 Dickson Permutational Polynomial

1.3.4.1 Construction

Let $a \in \mathrm{GF}(q)$ where $q = p^n$ with p a prime, then the Dickson polynomial

$$g_k(x,a) = \sum_{j=0}^{\lfloor k/2 \rfloor} \frac{k}{k-j} \binom{k-j}{j} (-a)^j x^{k-2j}$$

permutes $\mathrm{GF}(q)$ if and only if $\gcd(k, q^2-1) = 1$.

1.3.4.2 Analysis of the Construction

Maximum Auto- and Cross-Correlation.

$$g_k(x,a) = x^k - akx^{k-2} + \sum_{j=2}^{\lfloor k/2 \rfloor} \frac{k}{k-j} \binom{k-j}{j} (-a)^j x^{k-2j}$$

Since the degree of the polynomial is k then the maximum correlation is k.

Number of users. In order to have multiple users it is also needed that $\gcd(k,p) = 1$. Then there are q different users.

Number of users	N	q
Maxima auto- and cross-correlation	λ	k (degree of polynomial)
Length	N_p	q^2
Weight	ω	q
Alphabet	N_h	q

$a\backslash x$	0	1	2	3	4	5	6
0	0	1	4	5	2	3	6
1	0	1	2	4	3	5	6
2	0	4	6	5	2	1	3
3	0	3	2	1	6	5	4
4	0	5	4	6	1	3	2
5	0	3	5	6	1	2	4
6	0	4	5	1	6	2	3

Figure 1.4: Family of Dickson Polynomial with q=7 and k=5

Example.

1.3.5 Constructions with Trace function

In general the sequences used in wideband systems are cyclic, and therefore a casual observer must not be able to see the complete cycle easily, and a long sequence period is desirable. In addition, a sequence that is generated in more complex ways probably will be harder to intercept and predict even when obtaining a large portion of the cycle. Taking all this into account we design a family of sequences that have a very long cycle while maintaining at the same time a small alphabet, and generate them by more complex algebraic methods.

Our design starts from a well known family of sequences generated by solving polynomials for all the elements in a finite field. Then by algebraic manipulation using the trace function, we reduce the symbol alphabet while maintaining the same sequence period. This allows us to obtain a more secure key for each user.

1.3.5.1 Trace function

The trace function is define as follows: Let x be an element in $\mathrm{GF}(q^a)$ the trace of x from $q^a \to q$ is,

$$\mathrm{Tr}_q^{q^a}(x) = \sum_{i=0}^{a-1} x^{q^i}.$$

Some properties of the trace function are:

1. For $x \in \mathrm{GF}(q^a), \mathrm{Tr}_q^{q^a}(x) \in \mathrm{GF}(q)$
2. For $x, y \in \mathrm{GF}(q^a), \mathrm{Tr}_q^{q^a}(x+y) = \mathrm{Tr}_q^{q^a}(x) + \mathrm{Tr}_q^{q^a}(y)$
3. For $x \in \mathrm{GF}(q^a), c \in \mathrm{GF}(q), \mathrm{Tr}_q^{q^a}(cx) = c \cdot \mathrm{Tr}_q^{q^a}(x)$
4. $\mathrm{Tr}_q^{q^a}(x) = 0 \iff x = \beta^q - \beta$ for some $\beta \in \mathrm{GF}(q^a)$

1.3.5.2 Construction.

Let $f(x) = \sum_{i=0}^{d} f_i x^i$, be a polynomial in F_{q^a} where q is a prime. Then, we will generate a pair (i, j) as follows:

$$\mathrm{Tr}\left(f(\alpha^i)\right) = j$$

where α is a primitive element of F_{q^a}, i is an integer from 0 to $q^a - 2$ and j an integer from 0 to $q-1$. This two-dimension design is converted to a one-dimensional sequence by the use of the CRT. Note that we have attached a 2-D sequence in this way to a polynomial $f(x)$.

Example 3. Let $q^a = 2^{2^2}$ and $q = 2^2$ with $x^4 = x^3 + 1$ and $\alpha = \beta$. Then, let d=2 and $\gamma_0 = \alpha^3$ we can generate the following sequence shown in Table 1.3.

1.3.5.3 Analysis of the Construction

The quality measure of a family of sequences is determined by using some well known bounds (like the Johnson Bound) that relate the length, the alphabet and the correlation to the number of users. At this moment we have bounds for the correlation and the number of users, for a given length and alphabet, and with these results our set satisfies these asymptotically to equality.

3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	0	1	1	0	0	1	0	1	0	0	0	0
1	0	0	0	1	0	0	1	1	0	1	0	1	1	1	1
	2	2	2	1	2	2	1	1	2	1	2	1	1	1	1

Table 1.3: A Trace MZKZ sequence in matrix form

Before some of the parameters can be calculated facts about character functions are needed.

First,

Theorem 3 *If χ is a character then*

$$\chi(0) = 1.$$

Second,

Theorem 4 *If χ is a non-trivial character then*

$$\sum_{\theta \in F_q} \chi(\theta) = 0.$$

Theorem 5 (Carlitz-Uchiyama Bound) *If $f(x)$ is a polynomial over F_{q^a} of degree d then*

$$\sum_{x \in F_{q^a}} \chi\left(Tr(f(x))\right) \leq \pm \left[q^{a/2}(d-1)\right]$$

Number of signature sequences. To calculate the number of signature sequences we generate a set of polynomials G such that shifts (vertical or horizontal) of one is not equal to another. Let δ denote a vertical shift and τ a horizontal shift. Then, a polynomial $f(x) = \sum_{i=0}^{d} f_i x^i$ is in G if and only if the following equality does not hold.

$$\mathrm{Tr}\left(\sum_{i=0}^{d} f_i (x\alpha^{\tau})^i\right) + \delta = \mathrm{Tr}\left(\sum_{i=0}^{d} h_i x^i\right) \forall \ \ x \ \in F_{q^a}^*,$$

which implies

$$\mathrm{Tr}\left(\sum_{i=0}^{d} x^i(f_i\alpha^{\tau i} - h_i)\right) = -\delta$$

or

$$\mathrm{Tr}\left(\sum_{i=1}^{d} x^i(f_i\alpha^{\tau i} - h_i)\right) = \mathrm{Tr}\left(h_0 - f_0\right) - \delta. \tag{1.19}$$

When d is not $>> \sqrt{p^a}$, equation (1.19) holds if and only if $f_i\alpha^{\tau i} = h_i$ and $\mathrm{Tr}(h_0 - f_0) = \delta$. Therefore, by setting $f_d = 1$ and $\mathrm{Tr}(f_0) = 0$ for all sequence, we have that

$$|G| = q^{a(d-1)}q^{a-1} = q^{ad-1}.$$

Maximum Auto- and Cross-Correlation. To calculate the maximum auto- and cross-correlation for this construction we use knowledge of characters on finite fields and the Weil-Carlitz-Uchiyama bound (see [31]). We have to find the number of zeros of a polynomial in a finite field, namely $\mathrm{Tr}(g(x)+\delta)$ (note that since $\delta \in \mathrm{F}_q$, $\mathrm{Tr}(g(x) + \delta)$ =$\mathrm{Tr}(g(x)) + \delta$) where $g(x) = f(x\alpha^\tau) - h(x)$ is a polynomial in F_{q^a} of degree $d-1$ (note that $r < d$). For this we start with the following equation:

$$\sum_{x\in \mathrm{F}_{q^a}} \sum_{\delta\in \mathrm{F}_q} \chi\left(\mathrm{Tr}\left(g(x)\right) + \delta\right) = q\left|\{x \in \mathrm{F}_{q^a} : \mathrm{Tr}\left(g\left(x\right) + \delta\right) = 0\}\right| \tag{1.20}$$

where χ is a character sum and $|\cdot|$ denotes the cardinality of a set.

The equality holds since x fixed in the inner sum implies that $\mathrm{Tr}(g(x))$ is fixed. Therefore, if $\mathrm{Tr}(g(x) + \delta) \neq 0$ then,

$$\sum_{\delta\in \mathrm{F}_q} \chi(\mathrm{Tr}(g(x) + \delta)) = 0.$$

On the other hand, when $\mathrm{Tr}(g(x)+\delta) = 0$ we have that $\chi(0) = 1$ and (1.20) follows.

We now interchange the order of summation. Using the Weil-Carlitz-Uchiyama bound we have:

$$\sum_{\delta\in \mathrm{F}_q} \sum_{x\in \mathrm{F}_{q^a}} \chi\left(\mathrm{Tr}\left(g\left(x\right) + \delta\right)\right) \leq q\left[q^{a/2}\left(d-2\right)\right]. \tag{1.21}$$

Combining (1.20) and (1.21) we have:

$$|\{x \in \mathrm{F}_{q^a} : \mathrm{Tr}\,(g\,(x) + \delta) = 0\}| \leq q^{a/2}\,(d-2)\,.$$

As can be seen the degree of the polynomial affects the correlation linearly while it affects the number of sequences exponentially. Table 1.4 collects these results. With this design the difference between the alphabet N_h and the period N_p is from

Number of users (signature sequences)	N	$q^{a/2}\,(d-2)$
Maxima auto- and cross-correlation	λ	q^{ad-1}
Length	N_p	q^{a+1}
Weight	ω	q^a
Alphabet	N_h	q

Table 1.4: Parameters for the trace constructions

q^a to q, therefore if we have a machine with a fixed maximum N_h we can generate a longer sequence and hopefully more secure with this design. However, the price is paid on the correlation.

1.4 Power Spectral Density Computations

The ability to design a UWB signal set with a flat power spectral density (PSD) is one key to successful UWB signal design. In principle, the flatter the power spectral density of the transmission, the larger the amount of power that can be radiated while still satisfying PSD bounds imposed by regulatory agencies. Such tests may be done on the UWB carrier without data modulation.

1.4.1 The Pulse

We have a pulse $p(t)$ defined as follows:

$$p(t) = A\left[1 - 4\pi\left(\frac{t-t_d}{\tau_n}\right)^2\right]\exp\left[-2\pi\left(\frac{t-t_d}{\tau_n}\right)^2\right]$$

where A is the amplitude, τ_n the spread and t_d the location for which $p(t)$ is symmetric (see Figure 1.5. This pulse is approximately one nanosecond in duration, as

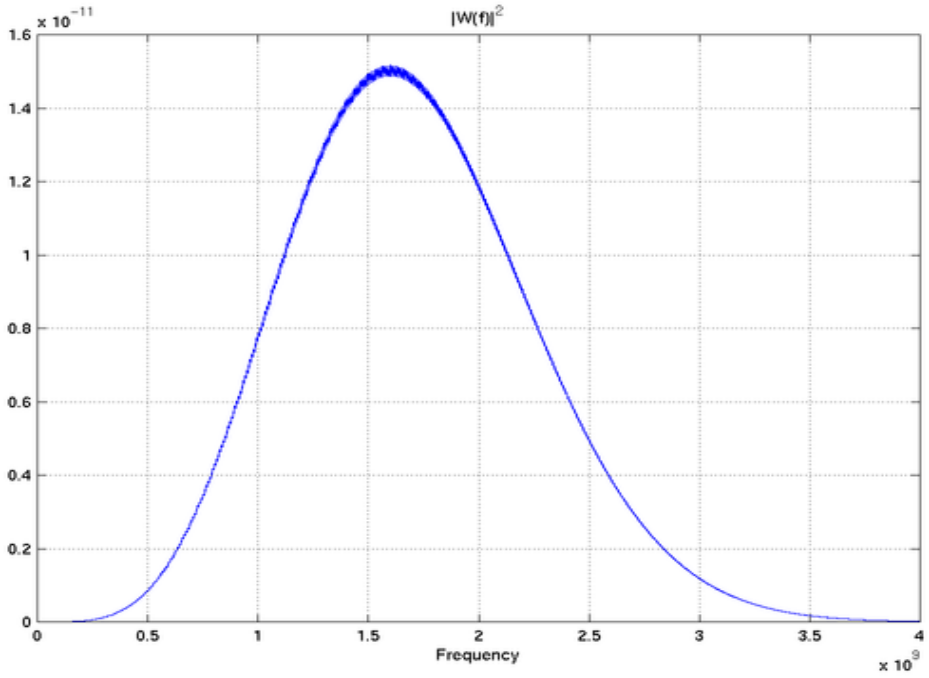

Figure 1.5: PSD of the UWB pulse

its derivative is the received pulse waveform (after the antenna processing). From a dimensional point of view, this means that the system is using at least 10^9 dimensions/second in its transmission format.

In the analytical work to follow, we will assume that the monocycle $p(t)$ is non-zero only in a small interval of the real time line beginning at time $t = 0$. Trains of pulses will be constructed analytically by shifting this waveform, e.g., $p(t - t_0)$ would represent the same waveform shifted in time so that its interval of non-zero value begins at time t_0

1.4.2 UWB signal Model

A wide variety of UWB carriers can be modeled in the simple form

$$s^{(i)}(t) = \sum_n a_n^{(i)} p(t - nT_c)\,, \tag{1.22}$$

where the $\{a_n^{(i)}\}$ are real numbers. This model embraces both time-hopping ($a_n^{(i)} \in \{0, 1\}$) and direct-sequence ($a_n^{(i)} \in \{1, -1\}$) spread-spectrum signals. As before, we

assume that the period of the sequences is N, and use i to represent the user index. One way of mathematically modeling the generation of these signals is shown in Figure 2. The impulse response $h_{\rm op}^{(i)}(t)$ of the one-period sequence generator for the $i^{\rm th}$ waveform is

$$h_{\rm op}^{(i)}(t) = \sum_{n=0}^{N-1} a_n^{(i)} \delta(t - nT_{\rm c}) \tag{1.23}$$

where $\delta(t)$ is the Dirac delta function. The impulse response of the pulse shaping circuit is simply $p(t)$.

Because the output of the code period clock has period $NT_{\rm c}$, it is easily verified that the output has PSD $S_{\rm cpc}(f)$ given by a sum of Dirac delta functions of equal area at multiples of $(NT_{\rm c})^{-1}$,

$$S_{\rm cpc}(f) = \frac{1}{(NT_{\rm c})^2} \sum_k \delta\left(f - \frac{k}{NT_{\rm c}}\right) . \tag{1.24}$$

The system function of the one-step sequence generator is

$$H_{\rm op}^{(i)}(f) = \mathbb{F}\left\{h_{\rm op}^{(i)}(t)\right\} = \sum_{n=0}^{N-1} a_n^{(i)} e^{-j2\pi f n T_{\rm c}} , \tag{1.25}$$

where $\mathbb{F}\{\cdot\}$ denotes the Fourier transform operation. The system function of the pulse shaper is simply the Fourier transform $P(f)$ of the pulse shape $p(t)$. It follows immediately that the PSD of the signal $s^{(i)}(t)$ is

$$\begin{aligned} S_{s^{(i)}}(f) &= \left|P(f)H_{\rm op}^{(i)}(f)\right|^2 S_{\rm cpc}(f) \\ &= |P(f)|^2 \left|\sum_{n=0}^{N-1} a_n^{(i)} e^{-j2\pi f n T_{\rm c}}\right|^2 \frac{1}{(NT_{\rm c})^2} \sum_k \delta\left(f - \frac{k}{NT_{\rm c}}\right) \\ &= \frac{|P(f)|^2}{(NT_{\rm c})^2} \sum_k C_k^{(i)} \delta\left(f - \frac{k}{NT_{\rm c}}\right) , \end{aligned} \tag{1.26}$$

where

$$C_k^{(i)} = \left|\sum_{n=0}^{N-1} a_n^{(i)} e^{-j2\pi k n/N}\right|^2 . \tag{1.27}$$

Hence, the coefficients $\{C_k^{(i)}\}$ represent the effect of code design on the PSD of the UWB signal $s^{(i)}(t)$ without data modulation. The coefficients are periodic, i.e.,

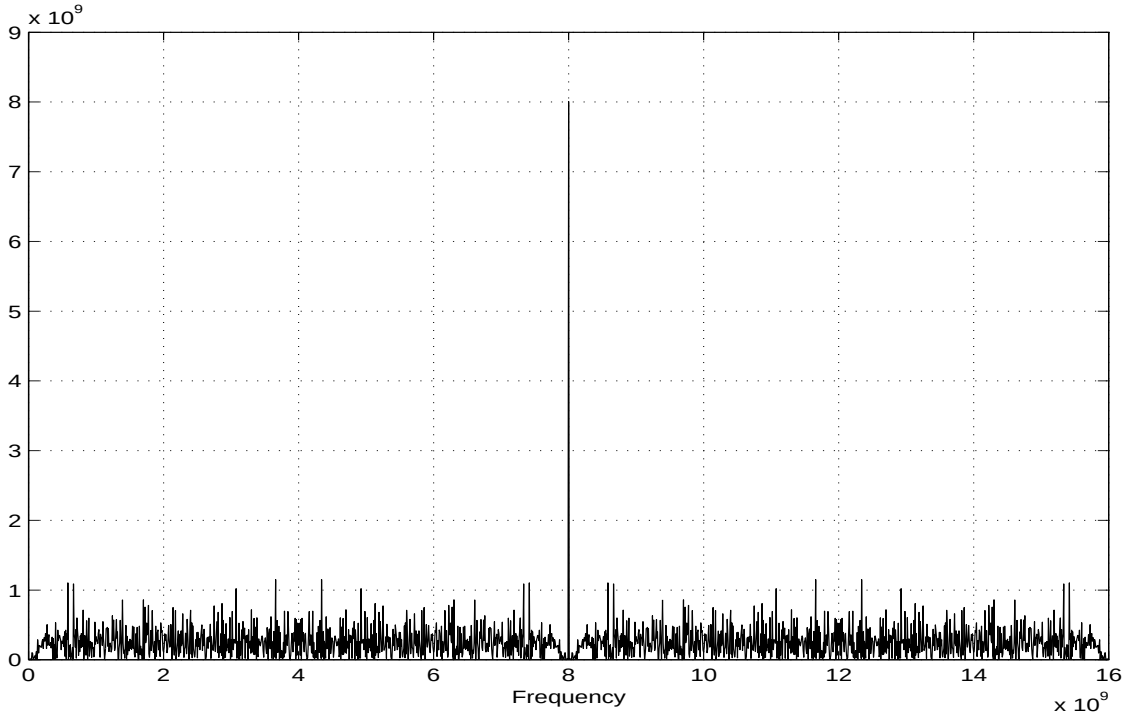

Figure 1.6: C_k^{15} of quadratic construction with p=31

$C_k^{(i)} = C_{k+N}^{(i)}$, and because the $\{a_n^{(i)}\}$ are real, also possess a symmetry, namely $C_k^{(i)} = C_{-k}^{(i)}$ for all k and i.

In Figure 1.6 we show the C_k^{15} of a signal using the quadratic sequence construction with $p = 31$.

1.4.3 Spectral Flatness

The emphasis for many years has been on sequence design for low auto- and cross-correlation. However, repeated structures in the sidelobes of the autocorrelation of a sequence, even if small, can cause lines of uneven height in the spectral density of the sequence.

To make the PSD of (1.26) as flat as possible (see [16]), one could try to design the coefficients of (1.27) so that $C_k^{(i)}$ are inversely proportional to $|P(\frac{k}{NT_c})|^2$. This is a difficult, if not impossible, problem because of the constraints that are caused by the allowable choices for $a_n^{(i)}$ and the symmetries and periodicities of the $\{C_k^{(i)}\}$ sequence. The alternative design objective that will be shown next, which is independent of the choice of pulse shape, is to make the values of $C_k^{(i)}$, $k = 0, 1, \ldots, \lceil N/2 \rceil - 1$,

as uniformly small as possible, the remainder of the values of $C_k^{(i)}$ for other k then being determined by periodicity and symmetry.

Expanding (1.27) gives some insight into designing for spectral flatness in pure time-hopping signals ($a_m^{(i)} \in \{0, 1\}$). Then

$$\begin{aligned} C_k^{(i)} &= \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_m^{(i)} a_n^{(i)} e^{j2\pi k(m-n)/N} \\ &= \sum_{r=0}^{N-1} N_r e^{j2\pi kr/N} \end{aligned} \tag{1.28}$$

where N_r is the number of times that the product $a_m^{(i)} a_n^{(i)} = 1$ when $m - n = r \mod N$ as m and n range from 0 to $N - 1$, i.e.,

$$N_r = \left| \left\{ m : a_m^{(i)} = a_{m+r \mod N}^{(i)} = 1, 0 \le m < N \right\} \right|. \tag{1.29}$$

The number of pulses in one period of the sequence is obviously N_0 and $C_0^{(i)} = N_0^2$, regardless of the location of the pulses within a period of the sequence. The total number of coincidences in the double sum of (1.28) is N^2, and therefore minimizing the maximum value of the remaining N_r can be lower bounded by spreading the remaining coincidences evenly over the $N - 1$ remaining values of N_r.

$$\min_{\{a_n^{(i)}\}} \max_{0<k<N} N_r \ge \frac{N^2 - N_0}{N - 1} \tag{1.30}$$

Achieving this bound with equality is a classic difference set design problem.

1.4.3.1 Difference Sets

Definition. Difference set

Let D be a set of k elements of integers from 0 to $v - 1$. Now, let us build a sequence R of the differences of all the elements of D modulo v

$$R = \{d_i - d_j \mod v : \text{ for all } i, j \text{ from 1 to } k, \text{ with } i \ne j\}$$

where d_i is an element of D. Then, let R_p be the set of pairs

$$R_p = \{(l, n) : l \text{ is the number of times the integer } n \text{ appears in } R\}.$$

Consequently, the set D is a difference set (v, k, λ) if $l = \lambda$ for all pairs in R_p (for a complete treatment of these sets see [3] and [41]). In this sense the difference set sequence will be form by ordering the set in ascending order.

Following our notation in (1.28), $N = v$ and $N_r = \lambda$ for all r from 1 to $v - 1$ and $N_0 = k$. Therefore, we can rewrite $C_k^{(i)}$ as

$$\begin{aligned} C_k^{(i)} &= N_0 + N_r \sum_{r=1}^{N-1} e^{j2\pi kr/N} \\ &= N_0 - N_r + N_r \sum_{r=0}^{N-1} e^{j2\pi kr/N} \\ &= \begin{cases} N_0 - N_r & \text{if } N \nmid k\,, \\ N + N_0 - N_r & \text{otherwise.} \end{cases} \end{aligned} \tag{1.31}$$

The coefficients are flat as suspected, except when $N \mid k$ which can be avoided by aligning it with the null part of $|P(f)|^2$ by setting T_c accordingly.

However, there is the problem of maintaining orthogonality between multiple users. To solve this problem we will multiply the k pulses modulated by the difference set sequence by the elements of a row of a Hadamard matrix (see [44]).

1.4.3.2 Hadamard matrix

Definition. Hadamard matrix

A Hadamard matrix H is a matrix with elements $h_m^{(i)} \in \{-1, 1\}$ where all the rows are orthogonal (the inner product is equal to $k\mathbf{I}$, where k is the size of the matrix and $\mathbf{I}$ is the identity matrix).

With these elements in place we can define $a_n^{(i)}$ for this design as follows:

$$a_n^{(i)} = \begin{cases} h_m^{(i)} & \text{if } n = d_m \in D\ , \\ 0 & \text{otherwise.} \end{cases}$$

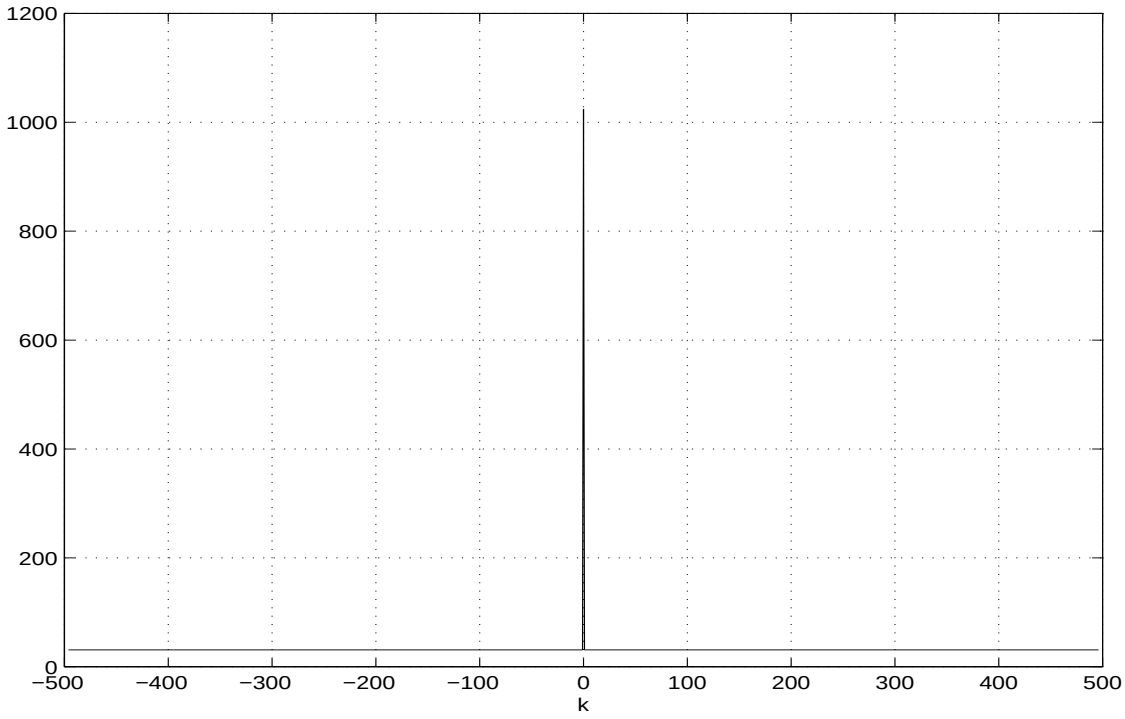

Figure 1.7: $C_k^{(i)}$ of difference set (993,32,1) with the all ones Hadamard vector

where $h_m^{(i)}$ are the elements of the row of the Hadamard matrix assign to the i-th user and d_m is the difference set sequence from set D.

In Figure 1.7 we show the $C_k^{(i)}$ of this design with the all ones Hadamard vector and in Figure 1.8 we have the case when a typical Hadamard vector is used.

1.4.4 PSD of Sequences from Polynomials with Integer coefficients

1.4.4.1 Bounds

Let define $f(x) = k(g(x) + xp^n)$, then $f'(x) = k(g'(x) + p^n)$, where $g(x)$ is the polynomial for the PN-sequence, with degree d, integer coefficients and where the degrees do not have p as a common divisor. Following this, we can define the following upper bound for the PSD of this kind of sequences as:

$$p^{m-n+t} \leq C_k = \left| \sum_{x=1}^{p^m} e^{\frac{2\pi i f(x)}{p^{m+n}}} \right|^2 \leq p^{m-n+t}(d-1)^2$$

where t is the power of p that divides k. In other words, $p^t|k$ (in most cases $t = 0$).

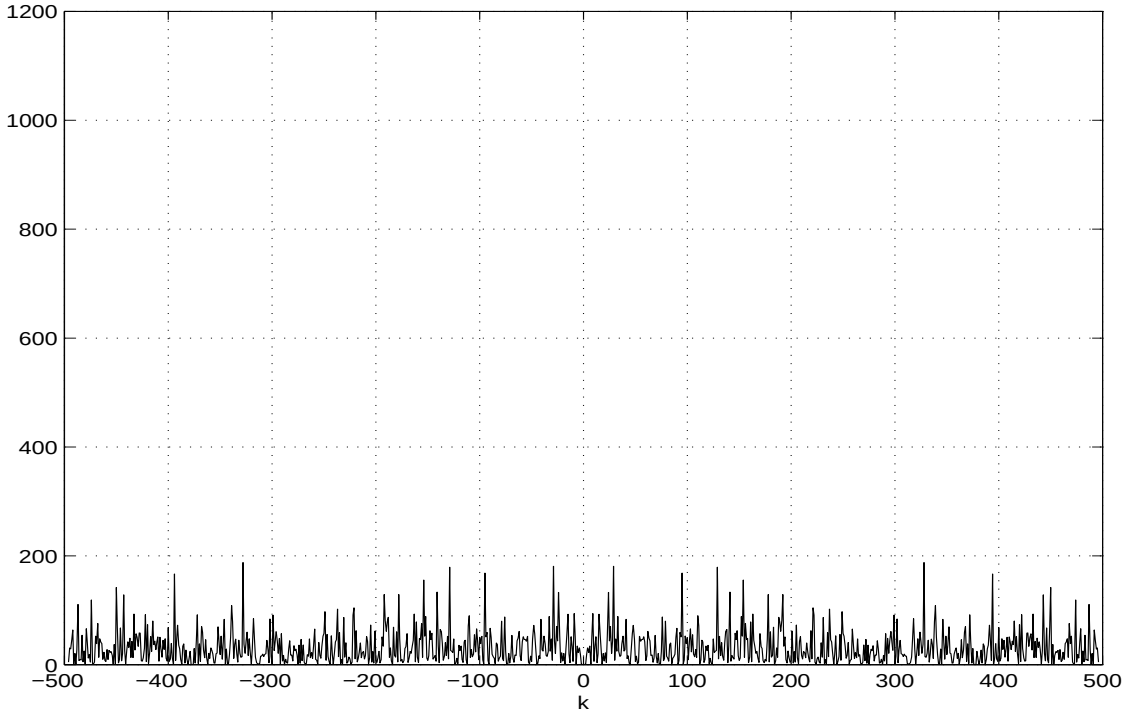

Figure 1.8: $C_k^{(i)}$ of difference set (993,32,1) with a typical Hadamard vector

1.4.5 PSD of Constructions with the Trace Function

Before we go into the particulars we first have to define a mapping that will formalize the notion of the trace function in this context. Recall the definition of C_k;

$$C_k = \left| \sum_{x=1}^{p^m} e^{\frac{2\pi i f(x)}{p^{m+n}}} \right|^2 .$$

Here $f(x) = \mathrm{Tr}_p^{p^{m+n}}(g(x))$, then we have

$$C_k = \left| \sum_{x=1}^{p^m} e^{\frac{2\pi i \mathrm{Tr}_p^{p^{m+n}}(g(x))}{p^{m+n}}} \right|^2 ,$$

but the notion of modulo p^{m+n} is not mathematically sound until we define the following mapping.

Definition. Let $\sum_{i=0}^{p^{m+n}} a_i p^i$ be the p^{m+n}-adic representation of any element of the field. Then, the a mapping $M(x) = x^{p^{m+n-1}} \mathrm{mod} p^{m+n}$ is such that $\mathrm{Tr}(M(x)) = \mathrm{Tr}(x)$.

Proof.

$$\begin{aligned} M(\sum_{i=0}^{p^{m+n}} a_i p^i) &= (\sum_{i=0}^{p^{m+n}} a_i p^i)^{p^{m+n-1}} \\ &= \sum_{i=0}^{p^{m+n}} (a_i p^i)^{p^{m+n-1}}, \end{aligned} \tag{1.32}$$

since all the cross terms contain a multiple of p^{m+n-1}. Now,

$$\begin{aligned} \mathrm{Tr}_p^{p^{m+n-1}}(M(x)) &= \mathrm{Tr}_p^{p^{m+n-1}}(\sum_{i=0}^{p^{m+n}} (a_i p^i)^{p^{m+n-1}}) \\ &= \sum_{j=0}^{m+n-1} (\sum_{i=0}^{p^{m+n-1}} (a_i p^i)^{p^{m+n-1}})^{p^j} \\ &= \sum_{j=0}^{m+n-1} \sum_{i=0}^{p^{m+n-1}} (a_i p^i)^{p^{m+n-1+j}} \\ &= \sum_{j=1}^{m+n-1} \sum_{i=0}^{p^{m+n-1}} (a_i p^i)^{p^{m+n-1+j}} + \sum_{i=0}^{p^{m+n}} (a_i p^i)^{p^{m+n-1}} \\ &= \sum_{k=0}^{m+n-2} \sum_{i=0}^{p^{m+n-1}} (a_i p^i)^{p^{kj}} + \sum_{i=0}^{p^{m+n}} (a_i p^i)^{p^{m+n-1}} \\ &= \sum_{k=0}^{m+n-1} \sum_{i=0}^{p^{m+n-1}} (a_i p^i)^{p^{kj}} \\ &= \mathrm{Tr}_p^{p^{m+n-1}}(\sum_{i=0}^{p^{m+n}} (a_i p^i)^{p^{m+n-1}}) \\ &= \mathrm{Tr}_p^{p^{m+n-1}}(x). \end{aligned} \tag{1.33}$$

Now, we can go into the specific of the constructions.

1.4.5.1 PSD of the Trace of MZKZ Sequences

Bounds.

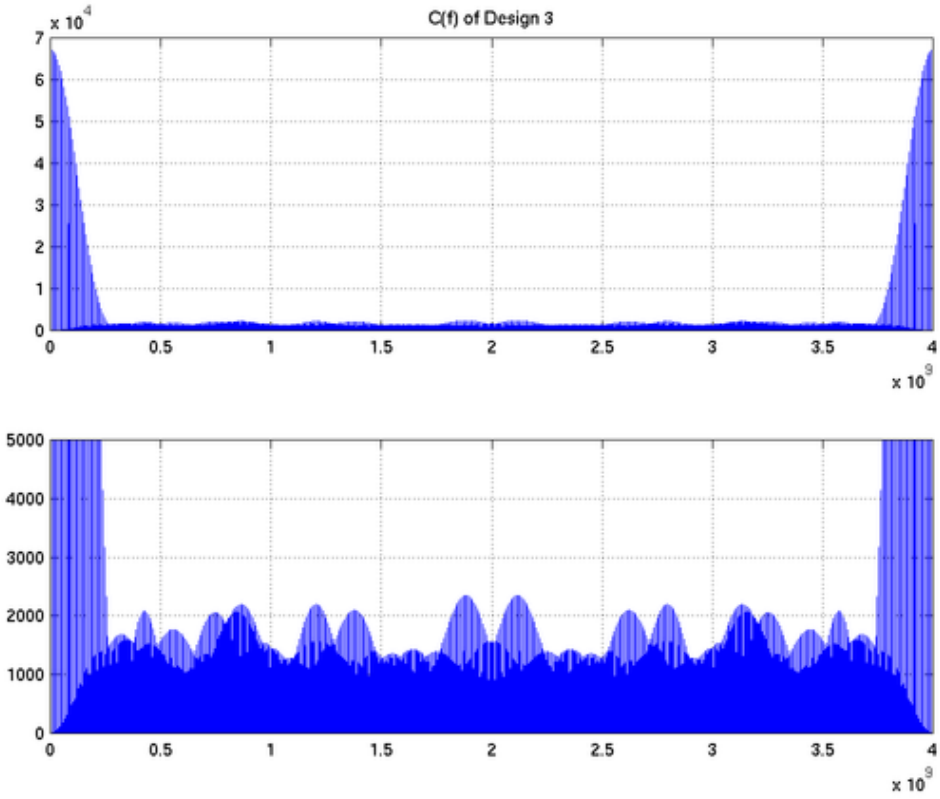

Figure 1.9: $C_k^{(i)}$ of Trace of a MZKZ Sequence with N=1024

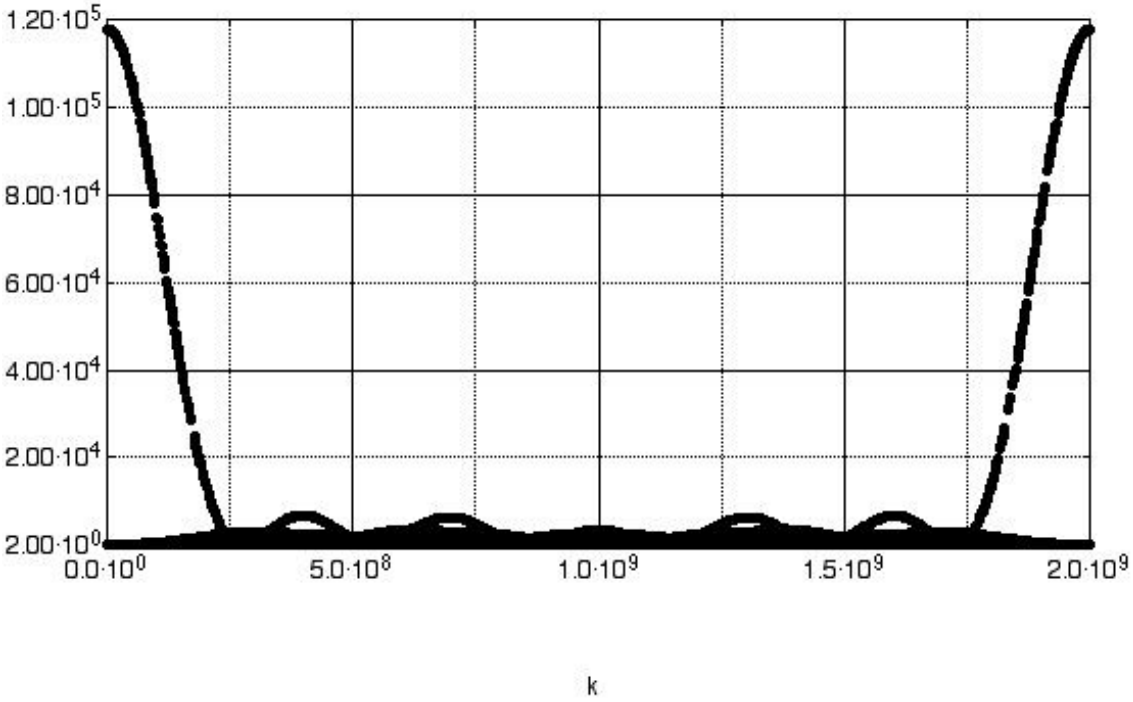

Figure 1.10: C_k of Trace of the Dickson Polynomial with $p = 7, q = 343$ and $d = 11$.

1.4.5.2 PSD of the Trace of Permutations

$$C_k = \left| \sum_{x=1}^{p^m} e^{\frac{2\pi i \mathrm{Tr} f(x)}{p^{m+n}}} \right|^2 \leq p^{m-n+t}(d-1)^2$$

$$\begin{aligned} C_k^{(i)} &= \left| \sum_{x=1}^{p^m} e^{\frac{2\pi i \mathrm{Tr} f(x)}{p^{m+n}}} \right|^2 \\ &= \left| p^{m-1} \sum_{x=0}^{p} e^{\frac{2\pi i x}{p^{m+n}}} \right|^2 \end{aligned} \tag{1.34}$$

1.4.6 Pseudo Difference Sets Sequences

As seen in Section 1.4.3.2, the problem of using difference sets with Hadamard vectors for multiple user environment is the constrains in the parameters that have to coincide. Here two approaches to work in with multiple users are presented.

1.4.6.1 Split DSS

The first solution is to take a (v, k, λ) DSS and split it to make r $(v, k/r, \lambda)$- Split DSS, where the r sequences have perfect cross-correlation. However, λ now becomes an upper bound on the number of times a difference appear. Therefore, as it would be shown in Section 1.4.3.2 the spectral flatness is lost.

1.4.6.2 Difference Triangle Sets

In [38] and [56] constructions of difference triangle sets are given. These are family of sequences where the cross-correlation is perfect and a difference appear λ times or not at all. Here again the spectral flatness is lost.

1.4.6.3 PSD of PDSS: Spectral Flatness is Lost

The strong requirement of the difference set sequences (all the differences appear the same number of times) assure the spectral flatness. However, this requirement can be weakened still maintaining the flatness of the spectrum.

Let go back to equation (1.28),

$$C_k^{(i)} = \sum_{r=0}^{N-1} N_r e^{j2\pi kr/N}.$$

Let $n = pq$, then we can rewrite r as $r = tp + a$ where t runs from 0 to $q-1$ and a from 0 to $p-1$. Then,

$$C_k^{(i)} = \sum_{a=0}^{p-1}\sum_{t=0}^{q-1} N_r e^{\frac{j2\pi k(tp+a)}{N}} = \sum_{a=1}^{p-1}\sum_{t=0}^{q-1} N_r e^{\frac{j2\pi k(tp+a)}{N}} + \sum_{t=1}^{q-1} N_{tp} e^{\frac{j2\pi kt}{q}} + N_0.$$

If we relaxed the requirement that all the differences appear the same number of times to the differences that are equal modulo p appear the same number of times then,

$$C_k^{(i)} = +N_0.$$

$$C_k^{(i)} = + = N_0 - N_{\{0\}}.$$

$$
\begin{aligned}
C_k^{(i)} &= N_0 + \sum_{a=1}^{p-1} N_{\{a\}} e^{\frac{j2\pi ka}{N}} \sum_{t=0}^{q-1} e^{\frac{j2\pi kt}{q}} + N_{\{0\}} \sum_{t=1}^{q-1} e^{\frac{j2\pi kt}{q}} \\
&= N_0 - N_{\{0\}} + N_{\{0\}} \sum_{t=0}^{q-1} e^{\frac{j2\pi kt}{q}} \\
&= \begin{cases} N_0 - N_{\{0\}} & \text{if } N \nmid k\,, \\ N + N_0 - N_{\{0\}} & \text{otherwise.} \end{cases}
\end{aligned} \tag{1.35}
$$

which is the same result as for the Difference sets sequences.

Following this we can see why the two options mentioned above are not solutions for the multiple-users difference sets type problem. However, maybe this last result can give a direction on new constructions for the UWB application.

Chapter 2

Algebraic Construction of Interleavers for Concatenation of Convolutional Codes

2.1 Introduction

In 1993 a class of near channel capacity achieving codes were introduced by Berrou et al[8]. After that, a burst of research have been done to analyze, generalize and in some cases try to minimize their importance. Some areas that had been dormant in the past have been resurrect, like Tanner graphs[51], LDPCs (Gallager codes)[22] and Pearl's believe propagation[35], and it generated a need to review the coding literature in order to give its place among this science (among the discoveries made, Robert McEliece found that work by Leonard Baum and Lloyd Welch in the early 1960's presented what have been called the BCJR algorithm predating the Bahl, Cocke, Jelinek and Raviv by more than a decade). Obviously, turbo codes reinvigorate the coding community not only because of its great performance but also for the amount of research it generated.

Among some of the generalizations that have been made, in [5] turbo codes were presented as a case of a parallel concatenation of convolutional codes among the general concatenation of convolutional codes. This approach will be follow thru this work to simplify the presentation of some of the ideas.

However the way its look at, there is a a crucial component, a so-called interleaver, that plays a fundamental role in the performance of this type of codes. In Figure 2.1 the performance of a turbo code is compare to the performance of a constituent code without the use of an interleaver. In the original implementation a pseudo random interleaver is used and in [5] it was explain why this kind of interleavers worked and some suggestions on what it will take for a good interleaver to be constructed.

In [20] a class called the Semi-random interleaver was introduced. The showed the best performance, even better than the random interleavers but their generation is thru a computer search and in some cases a construction with the desire parameters may not exists. Recently, in [27] a soft version of the semi-random interleavers was presented that ensure existence but by softening the more important parameter. They still performed better that the random interleavers but the construction is still a computerize search. Another problem that all this interleavers (random, s-random and ss-random) have, specially for long block lengths is that they have to be stored in memory and even though people are working to optimize the way they are placed in memory (see [12]) it is well known that memory takes the more amount of area in a chip design. Trying to avoid these problems some people have presented

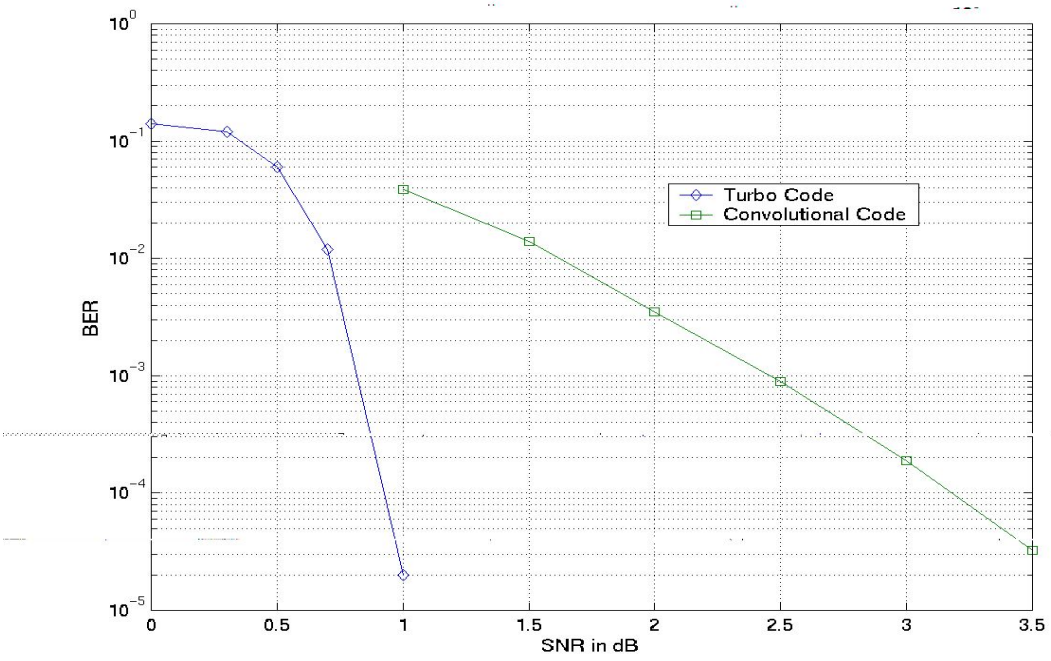

Figure 2.1: Interleaver vs no interleaver

deterministic constructions that performed as well as random interleavers; for small block interleavers (see [2]) and in [50] they work for lengths of powers of 2.

Following this line of though in this work deterministic interleavers are generated using algebraic methods, namely a family of polynomials called permutation polynomials. Permutation polynomials have been used for cryptographic systems and other combinatorial applications. In [17], L. Dickson listed all permutations of degree at most 5 and provided some monomials of very simple form and what have been called the Dickson polynomial that in some cases are also permutation polynomials. More recently, in 1971, T. Kasami introduced a class of polynomials in its work ([28]) with binary Reed-Muller codes that were later called Kasami permutation polynomials and in 1994, P. Muller[39] and S. D. Cohen and R. W. Mathews[15] introduced another class of permutation polynomials that have been called MCM permutation polynomials. Also a study on the composition of polynomials will be of interest to obtain the desired parameters. For this, permutations made of one cycle are useful because their implementation can be made with generic components like a shift-register. Two one-cycle permutations (cycles thru this work) will be study;

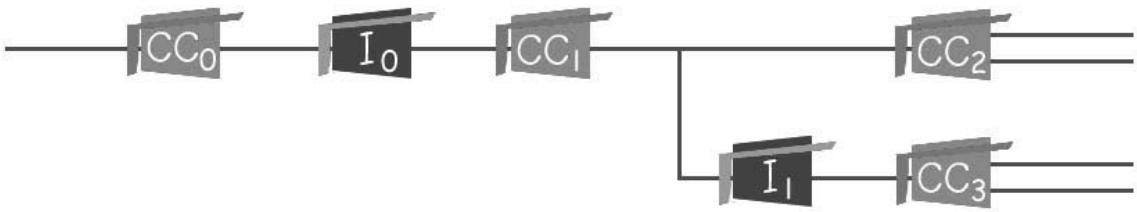

Figure 2.2: Concatenation of Convolutional Codes

a cycle generated from an add and carry operation and a cycle introduced in [7] in their work in extended Goppa codes.

In Section 2.2 the general frame work of concatenation of convolutional codes is introduced. Section 2.3 contains the presentations on interleavers and we start with the description of the permutation polynomials, the cycles and finally in Section 2.4 some preliminary results and routes of research are discussed.

2.2 Concatenation of Convolutional Codes (CCC)

A concatenation of convolutional codes is a system comprise of a number of convolutional codes that are separated by interleavers. The constituent codes can be arrange in any way as long as there is an interleaver between them (see Figure 2.2).

2.2.1 Serial CCC

Serial concatenation of convolutional codes is a case of CCC's. These are codes comprising of a number of convolutional codes separated by interleavers but in this case they are place serially with interleavers between them (see Figure 2.2.1).

Figure 2.3: Serially Concatenated Convolutional Codes

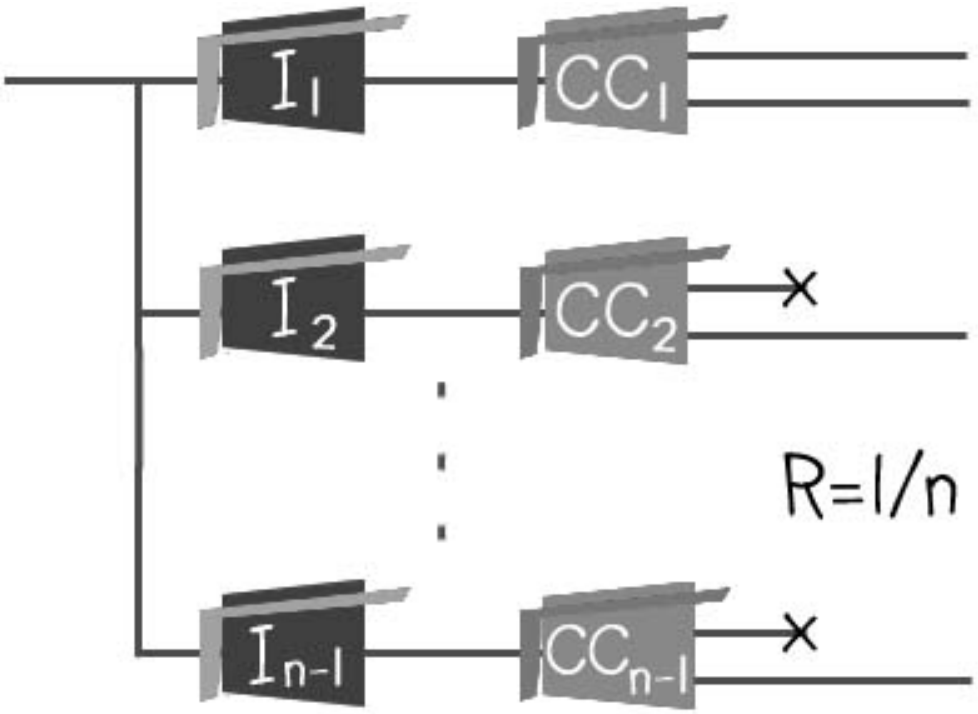

Figure 2.4: Parallel Concatenated Convolutional Codes

2.2.2 Parallel CCC

Parallel concatenation of convolutional codes is a case of CCC's where the codes are placed in parallel (see Figure 2.2.2). As mentioned before the most famous class of PCCCs are turbo codes.

2.2.3 Turbo Codes

As mentioned before, Turbo Codes are a new class of error correction codes that were introduced in 1993, by a group of researchers from France, along with a practical decoding algorithm [8]. The importance of turbo codes is that they enable reliable communications with power efficiencies close to the theoretical limit predicted by Claude Shannon. The original scheme presented in [8], uses a pair of rate 1/2, 16-state, systematic recursive convolutional codes (SRCCs) linked by a pseudo-random interleaver of length 65536 (see Figure 2.2.3).

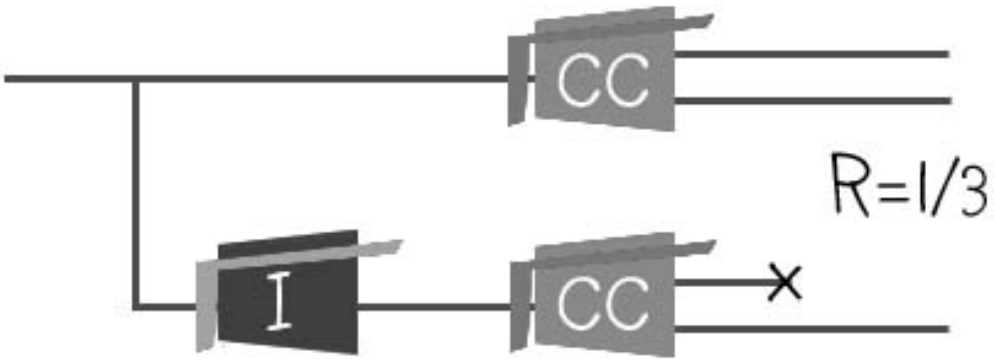

Figure 2.5: Turbo Codes

Figure 2.6: Interleaver Notation

2.3 Interleavers

An interleaver is a single input, single output device that permutes symbols from a fixed alphabet. The classical use of the interleaver is to randomize the locations of errors introduced in transmission. In Figure 2.3 we show the action of the interleaver and illustrate the notation to be used.

2.3.1 Parameters

In [5] it was shown from probabilistic arguments that the weight-2 data sequences (sequences of data whose resulting encoder sequence is of weight 2) are the most important when designing the interleavers. The performance of a turbo code depends on how effectively the data sequences that produce low encoded weights at the output of one encoder are matched with permutations of the same data sequence that yield higher encoded weights at the output of the others.

In order to have a handle on how good an interleaver performs the above describe matching, the spreading factor S was defined.

2.3.1.1 Spreading

Let $\Delta_x = j - i$ and $\Delta_y = \pi(j) - \pi(i)$, where $0 \leq i < j < N$ and $0 \leq \pi(\cdot) < N$. Then, an interleaver I has a spreading factor of S if whenever $\Delta_x < S$ then $|\Delta_y| \geq S$. As it can be seen from the definition the maximum spreading factor an interleaver can have is $\sqrt{N}$.

For the sequence designers the spreading factor can be seen as follows: A sequence have a spreading factor of S if the absolute value of all the differences that appear in the first S rows of the triangle of differences are smaller than S.

However, a permutation that is optimize to have a large spreading factor normally posses so mush regularity that the probabilistic argument that suggested the optimization does not hold. Therefore, in order to have this argument hold a permutation with some randomness is needed. A measure of this randomness is the dispersion parameter d.

2.3.1.2 Dispersion

Lets define the set of pairs D as follows:

$$D = \{(\Delta_x, \Delta_y)\} \tag{2.1}$$

where again $\Delta_x = j - i$, $\Delta_y = \pi(j) - \pi(i)$, with $0 \leq i < j < N$ and $0 \leq \pi(\cdot) < N$. N is the length and $\pi(i)$ is the value of the permutation at position i.

The the dispersion d is defined as

$$d = \frac{|D|}{N(N-1)/2} = \frac{2|D|}{N(N-1)} \tag{2.2}$$

The maximum dispersion an interleaver can have is 1.

For the sequence designer it can be seen as follows: the dispersion is inversely proportional to the number of times the differences appear in a row of the triangle of difference. If there are many repeated differences in the same row the dispersion goes down. From this it can be seen that a Costas sequence have $d = 1$.

2.3.1.3 Girth

The girth is defined as the length of the smallest cycle. Lets define a cycle in this context.

We start by defining the period of the convolutional code.

Example. The period of the convolutional code is 3 and the interleaver is 2 5 0 6 1 3 4 7. As can be seen in Figures 2.7 and 2.8 the girth is 2 (the minimum for this convolutional code and an interleaver of size $N = 8$).

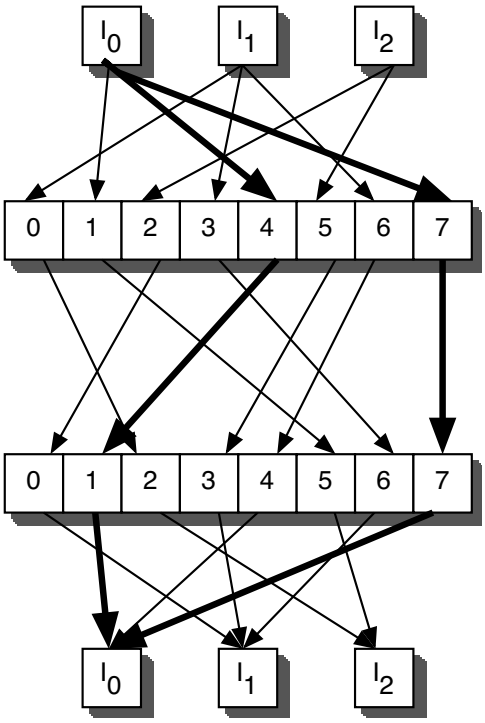

Figure 2.7:

However, if the interleaver is change then, the girth increases to 4.

Example. The period of the convolutional code is 3 and the interleaver is 2 5 0 6 3 1 7 4. As can be seen in Figures 2.9 and 2.10 the girth is 4 (the maximum for this convolutional code and an interleaver of size $N = 8$).

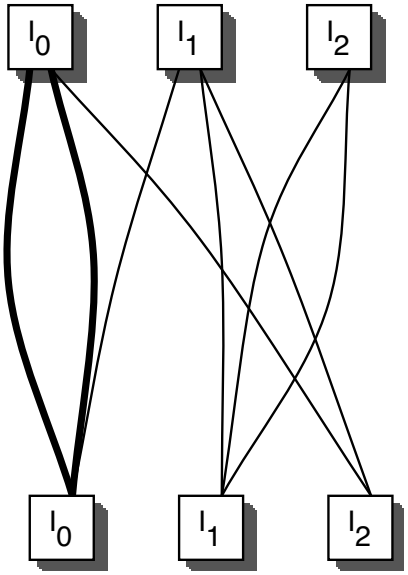

Figure 2.8:

2.3.2 Common Interleavers

2.3.2.1 Block

They can be describe with a matrix of size $N = m \times n$ where the data is written into the rows and read out from the columns. Mathematically they are described as

$$\pi(i) = in + \lfloor i/m \rfloor (\mod N)$$

for $0 \leq iN$

In Figure 2.11 a scatter plot (a dot is placed in the position $(i, \pi(i))$ of a block interleaver of size 1024 with $m = n = 32$ is shown.

2.3.2.2 Random

A random interleaver is generated from a random permutation based on a random noise source. For example, a noise vector of size N is generated and the permutation that puts the vector in order is used as the interleaver.

In Figure 2.12 a scatter plot of a random interleaver of size 1024 is shown.

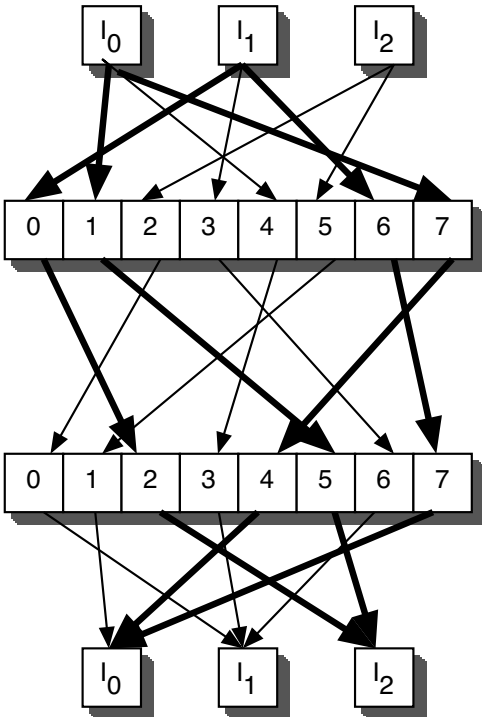

Figure 2.9:

2.3.2.3 Semi-Random

A semi-random interleaver is generated from a random permutation based on a random noise source with a spreading parameter S as a constrain. For example, we get an integer from a noise source and compare it to the previous S selections to see if its $\pm S$ away from them if so it is accepted if not is discarded and another candidate is selected.

In Figure 2.13 a scatter plot of a semi-random interleaver of size 1024 is shown.

2.3.3 Algebraic Interleavers

2.3.3.1 Permutational Polynomials

Some properties

1. If $f(x)$ is a PP of GF(q) then so is $g(x) = af(x+b) + c \; \forall \; a \neq 0, b, c \in \text{GF}(q)$
2. x^k permutes GF(q) if and only if $\gcd(k, q-1) = 1$

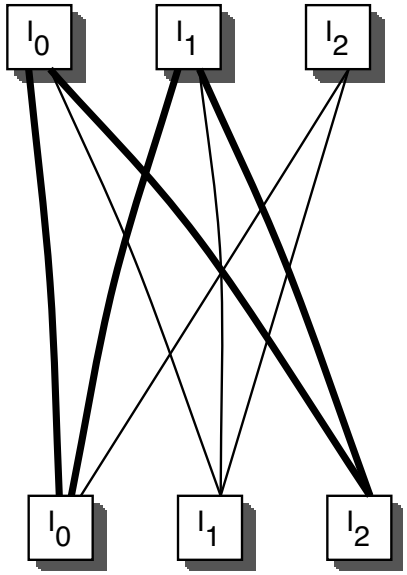

Figure 2.10:

3. If $f(x)$ and $g(x)$ are PP's then so are $g(f(x))$ and $f(g(x))$.

Dickson Polynomial Let $a \in \mathrm{GF}(q)$ where $q = p^n$ with p a prime, then the Dickson polynomial

$$g_k(x,a) = \sum_{j=0}^{\lfloor k/2 \rfloor} \frac{k}{k-j} \binom{k-j}{j} (-a)^j x^{k-2j}$$

permutes $\mathrm{GF}(q)$ if and only if $\gcd(k, q^2-1) = 1$.

Kasami Polynomial Let $q = 2^n$, then if $\gcd(k,n) = 1, k < n$ and $k' = 1/k(\mod n)$

$$q_\alpha(x) = \frac{\sum_{i=1}^{k'} x^{2^{ik}} + \alpha \mathrm{Tr}(x)}{x^{2^k+1}}$$

for $\alpha = 0, 1$ is called a generalized Kasami polynomial and is a permutation polynomial if and only if $k' + \alpha n \equiv 1(\mod 2)$.

In Figure 2.15 a scatter plot of a Kasami interleaver of size 1024, with $\alpha = 0$ and $k = 7$ is shown.

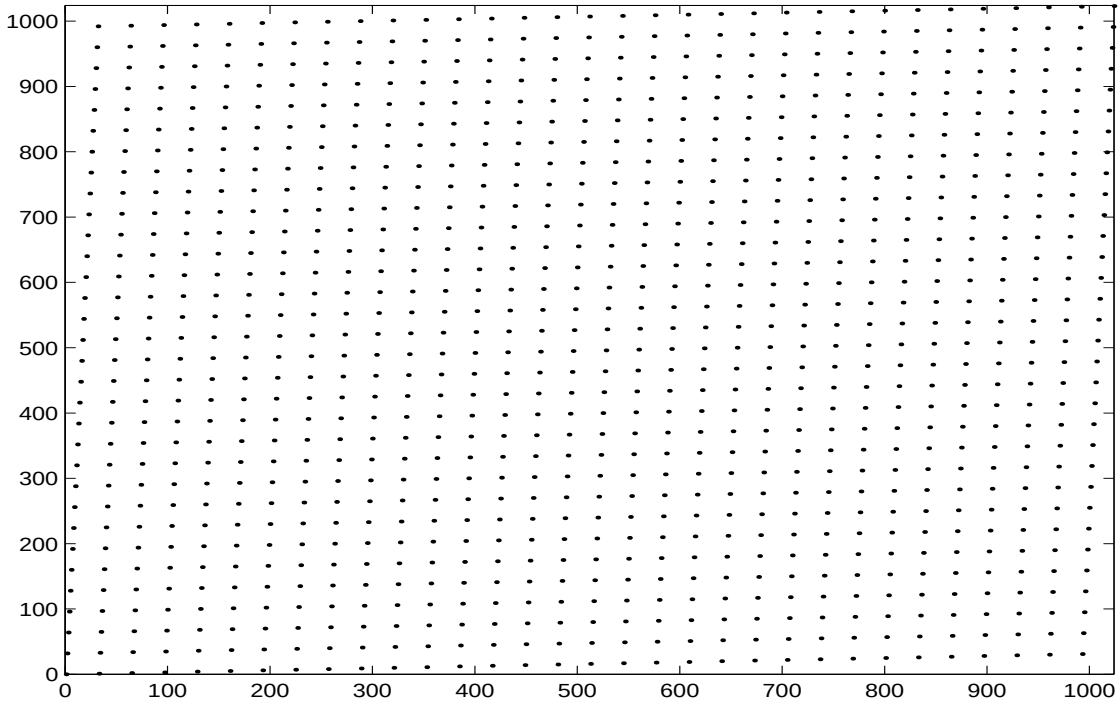

Figure 2.11: Scatter plot of block interleaver

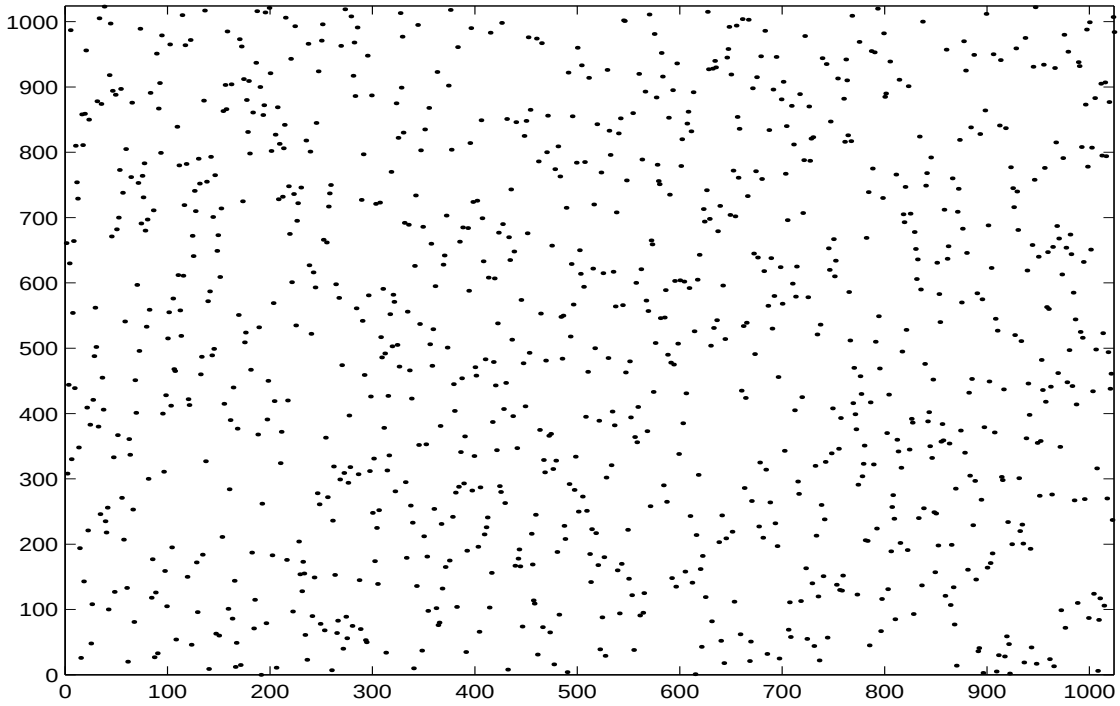

Figure 2.12: Scatter plot of random interleaver

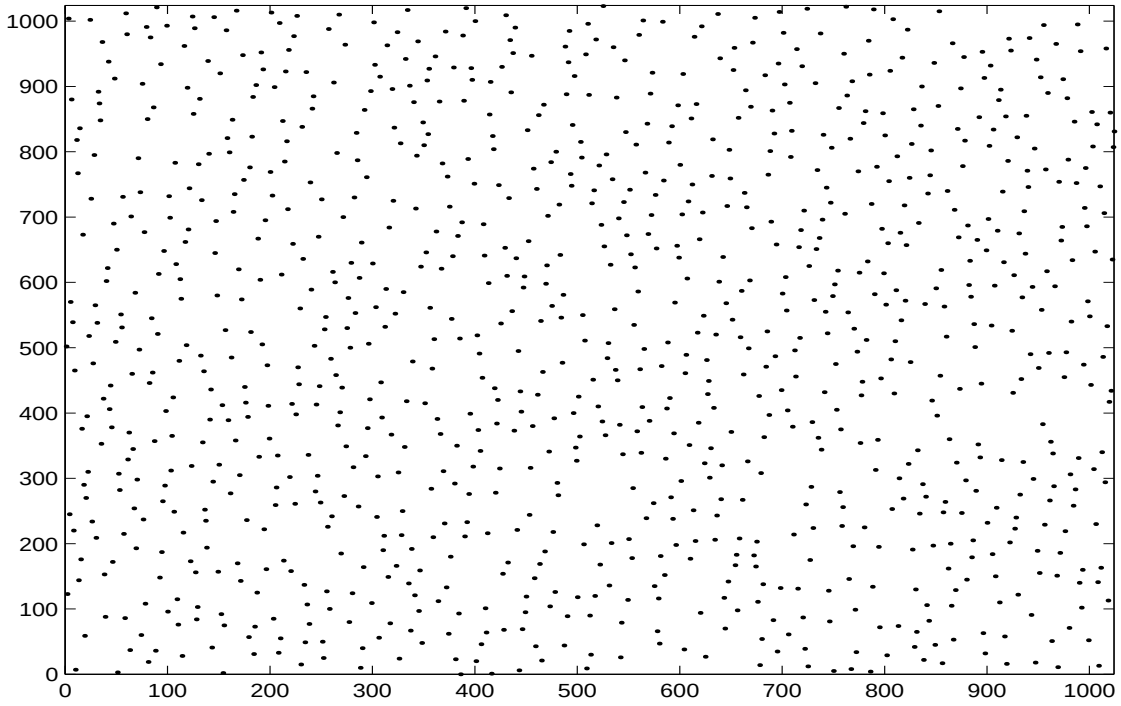

Figure 2.13: Scatter plot of Semi-random interleaver with S=16

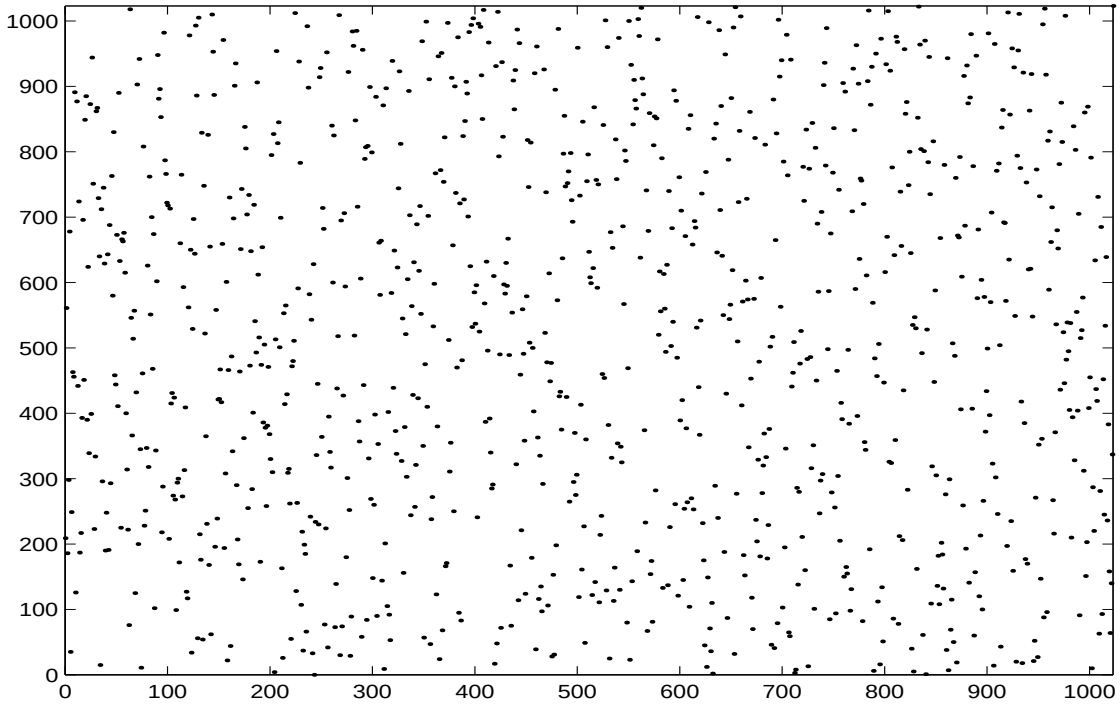

Figure 2.14: Scatter plot of Dickson Polynomial with $q = 1024$, $k = 7$ and $a = 10$

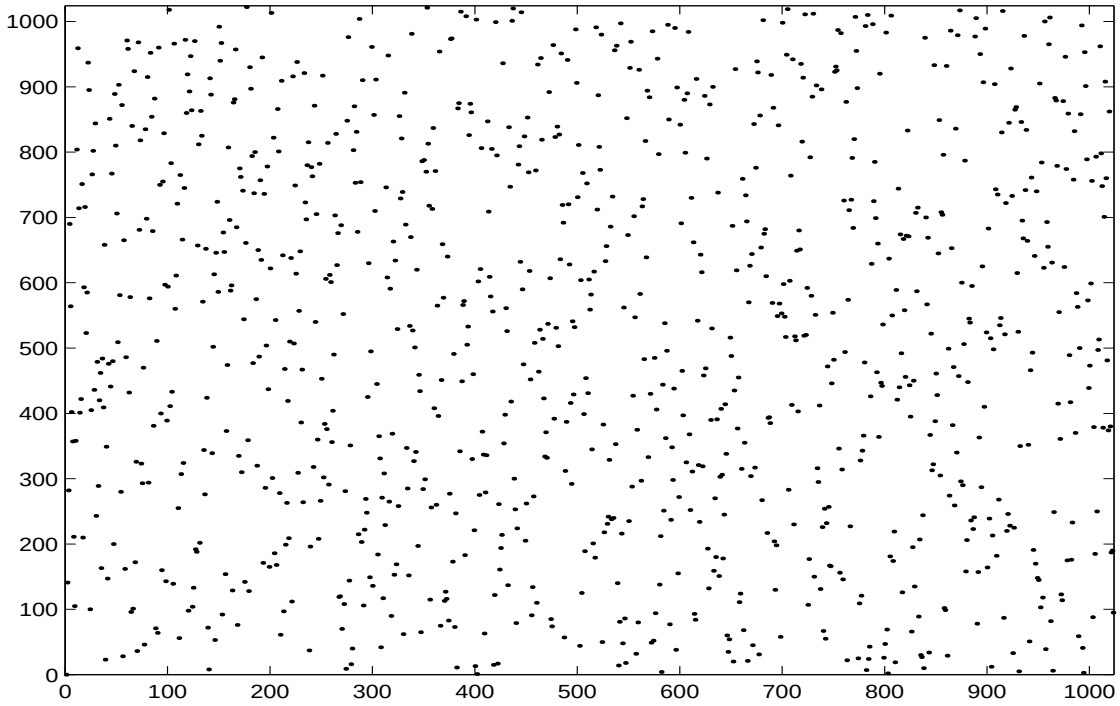

Figure 2.15: Scatter plot of Kasami interleaver with $\alpha = 0$ and $k = 7$

MCM Polynomials Let $q = 2^n$, then if $\gcd(k, n) = 1, k < n$

$$P_\beta(x) = \frac{\left(\sum_{i=0}^{k-1} x^{2^i} + \beta \mathrm{Tr}(x)\right)^{2^k+1}}{x^{2^k}}$$

for $\beta = 0, 1$ is called a generalized MCM polynomial and is a permutation polynomial if and only if $k + \beta n \equiv 1(\mod 2)$.

Linear Congruence This construction introduced for spread spectrum sequences in [53] and [24] is a direct result from the first and second properties of permutation polynomials.

$$f(x) = ax$$

for any a in GF(q).

In Figure 2.16 a scatter plot of a linear congruence interleaver of size 1024, with the p-adic cycle is shown.

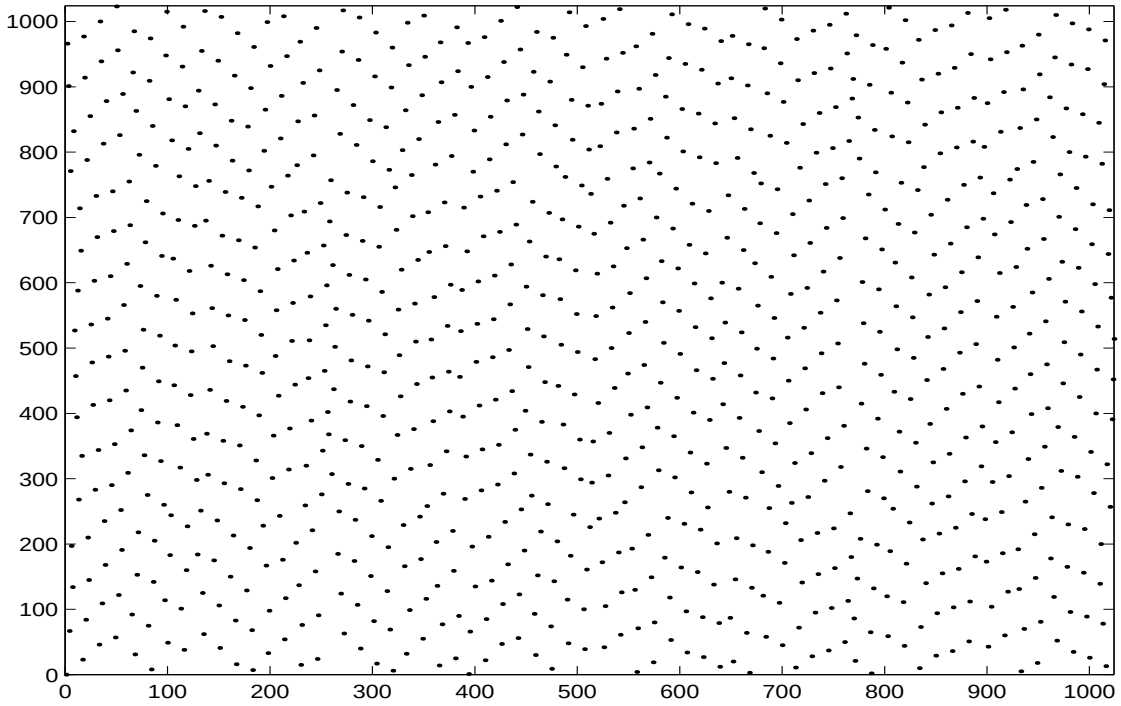

Figure 2.16: Scatter plot of linear interleaver with p-adic cycle

Hyperbolic Congruence This construction was also introduced for spread spectrum sequences in [36] and is also a direct result from the first and second properties of permutation polynomials.

$$f(x) = 1/ax$$

for any a in GF(q). It is obviously a permutation since $\gcd(q-2, q-1) = 1$.

In Figure 2.17 a scatter plot of a hyperbolic congruence interleaver of size 1024, with the p-adic cycle is shown.

2.3.3.2 Cycles

Here the one-cycle permutations are described.

P-adic Cycle. This cycle is generated by the add and carry operation. Starting with the number 0 in p-ary form and count up to $q-1$ taking the power of the primitive element as the value for that position (with the convention that the element 0 goes to $q-1$).

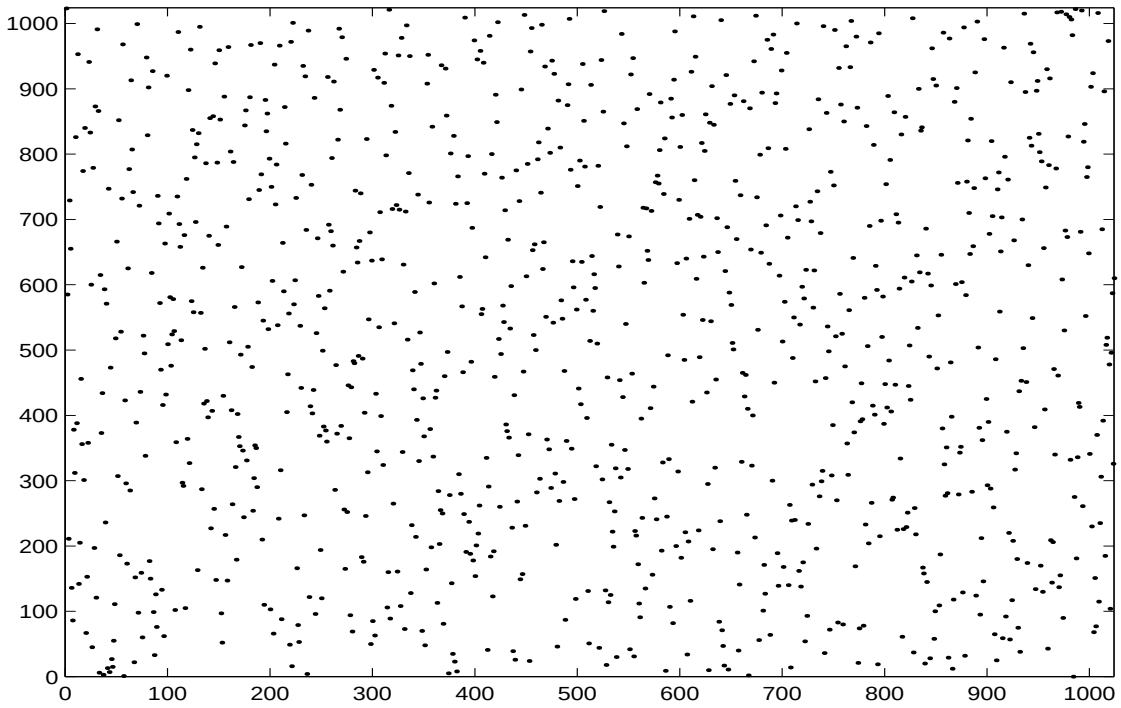

Figure 2.17: Scatter plot of hyperbolic interleaver with p-adic cycle

Example. Let $q = 2^3$ and $\alpha^3 = \alpha + 1$ then the resulting permutation will be $7, 0, 1, 3, 2, 6, 4, 5$.

Moreno-Berlekamp Cycle To describe this cycle, lets present again the result of Berlekamp-Moreno [7].

Theorem 6 *Whenever $x^2 + x + \alpha$ is irreducible, and α primitive in F, then the permutation given by $-\frac{\alpha}{x+1}$ gives a cycle of length $p^m + 1$.*

The powers of the primitive elements are selected as the value for the position (with the convention that the element 0 goes to $q - 1$).

Example With $q = 2^3$ we get the cycle $7, 1, 3, 6, 4, 2, 5, 0$.

2.4 Evaluation of Permutational Polynomial

2.4.1 Dickson Polynomials behave like Random Interleavers

In Table 2.1 the dispersion and spreading of some Dickson polynomial interleavers constructions are presented. It is worth noting that the results appear to be very similar to a random interleaver, with dispersion very similar (around 0.81) and spreading in some cases better. This can also been from the scatter plots in Figures 2.14.

Interleaver type	Dispersion	Spreading
Dickson ( )		
Random	0.812643	1
Random Average of 9	0.813838	1
Random Min	0.812326	1
Random Max	0.814795	1

Table 2.1:

2.4.2 Kasami Polynomials with p-adic cycle behave like Random Interleavers

In Table 2.2 the dispersion and spreading of some Kasami polynomial interleavers constructions are presented. It is worth noting that the results appear to be very similar to a random interleaver, with dispersion very similar (around 0.81) and spreading in some cases better. This can also been from the scatter plots in Figures 2.15.

2.4.3 Hyperbolic Polynomials with p-adic cycle behave like Random Interleavers

In Table 2.3 the dispersion and spreading of some hyperbolic interleavers constructions are presented. It is worth noting that the results appear to be very similar to a random interleaver, with dispersion very similar (around 0.81) and spreading in some cases better. This can also been from the scatter plots in Figures 2.17. However, in Figures 2.18 and 2.19 it can be seen that in simulation the performance of the hyperbolic congruence is not up to par with the random interleaver.

Interleaver type	Dispersion	Spreading
Kasami (k=7, $\alpha = 0$), w/ natural cycle	0.818867	1
Kasami (k=7, $\alpha = 0$), w/ p-adic cycle	0.811471	1
Kasami (k=7, $\alpha = 0$), w/ MB cycle	0.812005	1
Kasami (k=9, $\alpha = 0$), w/ natural cycle	0.623939	1
Kasami (k=9, $\alpha = 0$), w/ p-adic cycle	0.813461	1
Kasami (k=9, $\alpha = 0$), w/ MB cycle	0.811104	1
Kasami (k=9, $\alpha = 1$), w/ natural cycle	0.002929	2
Kasami (k=9, $\alpha = 1$), w/ p-adic cycle	0.829296	1
Kasami (k=9, $\alpha = 1$), w/ MB cycle	0.749754	3
Random	0.812643	1
Random Average of 9	0.813838	1
Random Min	0.812326	1
Random Max	0.814795	1

Table 2.2:

2.4.4 Linear Polynomials with p-adic cycle in average perform better than Random Interleavers

In Table 2.4 the dispersion and spreading of some liner polynomial interleavers constructions are presented. It is worth noting that the results appear to be better in average than a random interleaver, with dispersion smaller (around 0.5) but with a better spreading. This can also been from the scatter plots in Figures 2.16, and in Figures 2.18 and 2.19 it can be seen that in simulation where the better linear polynomial interleaver out performs the random one.

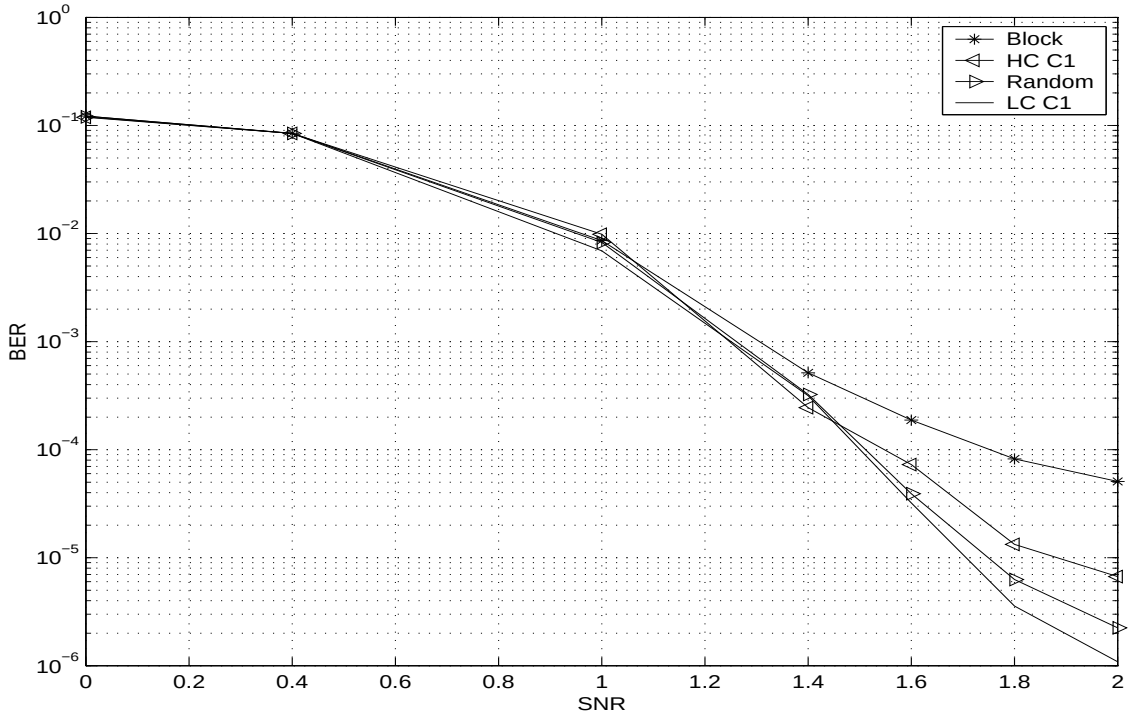

Figure 2.18: Bit error rate of various constructions

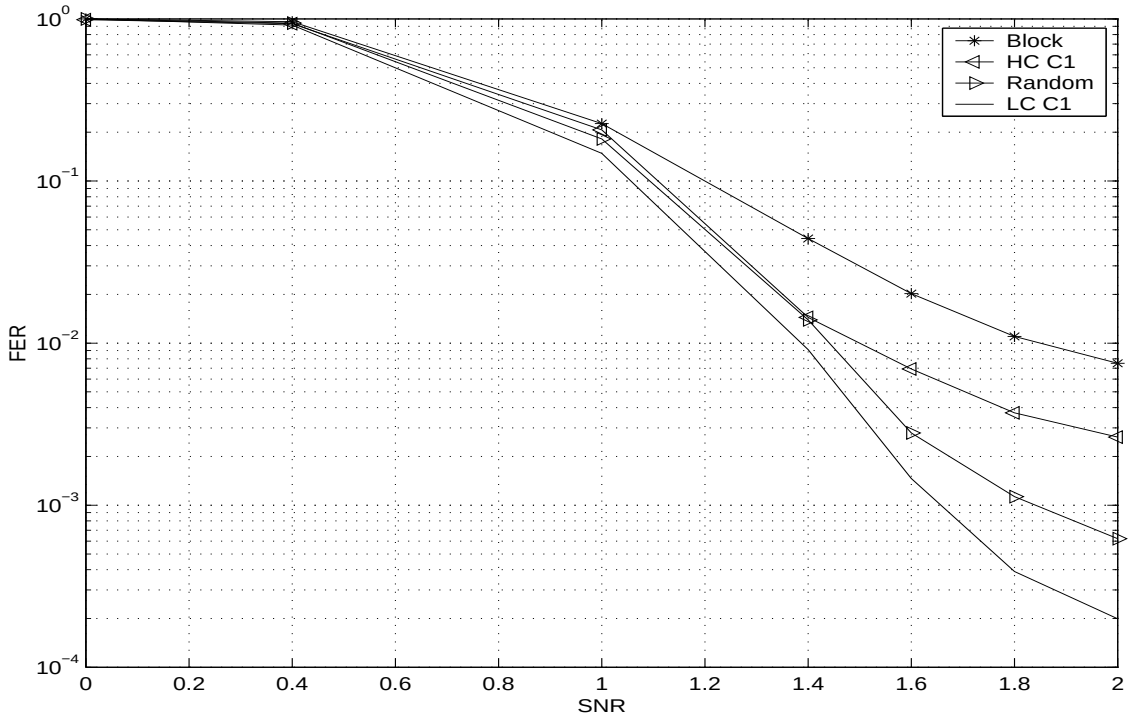

Figure 2.19: Frame error rate of various constructions

Interleaver type	Dispersion	Spreading
Hyperbolic w/ p-adic cycle	0.813920	3
Hyperbolic w/ p-adic cycle Average	0.814398	1.099
Hyperbolic w/ p-adic cycle Min Disp.	0.811369	1
Hyperbolic w/ p-adic cycle Max Disp.	0.816672	1
Random	0.812643	1
Random Average of 9	0.813838	1
Random Min	0.812326	1
Random Max	0.814795	1

Table 2.3:

Interleaver type	Dispersion	Spreading
Linear w/ p-adic cycle	0.560472	11
Linear w/ p-adic cycle Average	0.501552	9
Linear w/ p-adic cycle Min Disp.	0.006564	2
Linear w/ p-adic cycle Max Disp.	0.578068	2
Linear w/ p-adic cycle Min Spre.	0.318940	1
Linear w/ p-adic cycle Max Spre.	0.101435	18
Random	0.812643	1
Random Average of 9	0.813838	1
Random Min	0.812326	1
Random Max	0.814795	1

Table 2.4:

Reference List

[1] L. Y. Astan and A. Kostylev. *Ultra-wideband Radar Measurements: Analysis and Processing*. Inspec/IEE, 1999.

[2] A. S. Barbulescu and S.S. Pietrobon. Interleaver Design for Turbo Codes. *Electronic Letters*, 30:111–113, Jan 1994.

[3] L. Baumert. *Cyclic Difference Sets*, volume 182 of *Lecture Notes in Mathematics*. Springer-Verlag, 1971.

[4] S. Benedetto, D. Divsalar G. Montorsi, and F. Pollara. Serial Concatenation of interleaved codes: Performance Analysis, design and Iterative Decoding. *IEEE Transactions on Information Theory*, 44:909–926, May 1998.

[5] S. Benedetto and G. Montorsi. Design of Parallel Concatenated convolutional codes. *IEEE Transactions on Information Theory*, 44:591–600, May 1996.

[6] S. Benedetto and G. Montorsi. Unveiling Turbo Codes: Some Results on Parallel Concatenated coding schemes. *IEEE Transactions on Information Theory*, 42:409–428, March 1996.

[7] E. R. Berlekamp and O. Moreno. Extended double-error-correcting binary Goppa codes are cyclic. *IEEE Transactions on Information Theory*, 19:817–818, 1973.

[8] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes. In *Proceedings of ICC'93*, pages 1064–1070, Geneva, Switzerland, May 1993.

[9] P. J. Cameron and J. H. Van Lint. *Graph Theory, Coding Theory and Block Designs*. London Mathematical Society Lecture Note Series 19. Cambridge University Press, Cambridge, 1975.

[10] L. Car and et al, editors. *Ultra-wideband Short-Pulse Electromagnetics*, volume 1. Plenum Pub Corp, 1995.

[11] L. Car and et al, editors. *Ultra-wideband Short-Pulse Electromagnetics*, volume 2. Plenum Pub Corp, 1995.

[12] K. M. Chugg, A. Anastasopoulos, and X. Chen. *Iterative Detection*. Kluwer Academic Press, 2001.

[13] H. Chung and P. Kumar. Optical orthogonal codes - new bounds and an optical construction. *IEEE Transactions on Information Theory*, 36(4):866–873, July 1990.

[14] T. Cochrane and Z. Zheng. A Survey on Pure and Mixed Exponential Sums Modulo Prime Powers. To appear in the Proceedings of the Illinois Millennial conference.

[15] S. D. Cohen and R. W. Matthews. A Class of Exceptional Polynomials. *Transactions of the American Mathematical Society*, 345(2):897–909, Oct 1994.

[16] C. J. Corrada-Bravo, R. A. Scholtz, and P. V. Kumar. Generating TH-SSMA sequences with good correlation and low PSD level. In *1999 UWB Conference for Radio and Radar Technology*, Washington DC, September 1999.

[17] L. E. Dickson. *Linear Groups with an Exposition of the Galois Field Theory*. Dover, New York, 1958.

[18] D. Divsalar and R. J. McEliece. Effective Free Distance of Turbo Codes. *Electronic Letters*, 32(5):445–446, Feb 1996.

[19] D. Divsalara, H. J, and R. J. McEliece. Coding Theorems for Turbo-like Codes. In *Proceedings of Allerton Con. Commun., Control, Comp.*, pages 201–210, Allerton House, Illinois, 1998.

[20] S. Dolinar and D. Divsalar. Weight Distributions of Turbo Codes Using Random and Nonrandom Interleavers. Technical Report 42-122, Jet Propulsion Laboratory, Pasadena, CA, Aug 1995.

[21] C. Baum et al, editor. *Ultra-wideband Short-Pulse Electromagnetics*, volume 3. Plenum Pub Corp, 1995.

[22] R. Gallager. Low Density Parity Check Codes. *IEEE Transactions on Information Theory*, 8:21–28, Jan 1962.

[23] F. Q. Gouêa. *p-adic Numbers: An Introduction*. Springer-Verlag, Berlin, Heidelberg, 1993.

[24] T. Healy. Coding and Decoding for Code Division Multiple User Communication. *IEEE Transactions on Communication*, 33:310–316, Apr 1985.

[25] C. Heegard and S. B. Wicker. *Turbo Coding*, volume 476 of *Kluwer International Series Engineering and Computer Science*. Kluwer Academic, Boston, MA, Nov 1998.

[26] E. Heyman, editor. *Ultra-wideband Short-Pulse Electromagnetics*, volume 4. Plenum Pub Corp, 1995.

[27] J. Hokflet, O. Edfors, and T. Maseng. A Turbo Code Interleaver Design Criterion Based on the Performance of Iterative Decoding. *IEEE Communication Letters*, 5:52–54, Feb 2001.

[28] T. Kasami. The Weight Enumerators for Several Classes of Subcodes of the second order binary Reed-Muller Codes. *Information and Control*, 18:369–394, 1971.

[29] S. V. Konyag and I. E. Shparlinski. *Character Sums with Exponential Functions and their Applications*. Cambridge University Press, Cambridge, 1999.

[30] N. M. Korobov. *Exponential Sums and their Applications*. Kluwer Academic Publishers, Dordrecht, 1992.

[31] R. Lidl and H. Niederreiter. *Introduction to Finite Fields and their Applications,*. Cambridge University Press, Cambridge, 1986.

[32] D. J. C. MacKay. Good Error Correcting Codes Based on Very Sparse Matrices. *IEEE Transactions on Information Theory*, 45:399–431, Mar 1999.

[33] D. J. C. MacKay and R. M. Neal. Near Shannon Limit Performance of Low-Density Parity-Check Codes. *Electronic Letters*, 32:1645–1646, Aug 1996.

[34] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error Correcting Codes*. North-Holland, New York, 1983.

[35] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng. Turbo Decoding as an Instance of Pearl's 'Belif Propagation' Algorithm. *IEEE Journal on Selected Areas Communication*, 16:140–152, Feb 1998.

[36] O. Moreno and S. V. Maric. A New Family of Frequency-Hop Codes. *IEEE Transactions on Communications*, 48:1241–1244, Aug 2000.

[37] O. Moreno, Z. Zhang, P. V. Kumar, and V. Zinoviev. New Constructions of Optimal Cyclically Permutable Constant Weight Code. *IEEE Transactions on Information Theory*, 41:448–456, March 1995.

[38] Oscar Moreno. Optical orthogonal codes and sidon sets. 2001.

[39] P. Muller. New Examples of Exceptional Polynomials. In G. L. Mullen and P. J. Shiue, editors, *Finite Fields: Theory, Applicatios and Algorithms*, volume 168 of *Contemp. Math.*, pages 245–249. American Mathematical Society, Providence, RI, 1994.

[40] B. Noel, editor. *Ultrawideband Radar*. CRC Press, 1991.

[41] A. Pott, P. V. Kumar, T. Helleseth, and D. Jungnickel. *Difference Sets, Sequences and Their Correlation Properties*. Nato Science Series, Series C. Kluwer Academic Press, Holland, 1999.

[42] P. V. Kumar R. Scholtz and C. J. Corrada Bravo. Some problems and results in ultra-wideband signal design. In et al. T. Helleseth, editor, *Sequences and Their Applications: Proceedings of Seta '01*. Springer Verlag, 2002. Discrete Mathematics and Theoretical Computer Science Series.

[43] D. Divsalar S. Benedetto, G. Montorsi and F. Pollara. A soft-input soft-output maximum a posteriori (MAP) module to decode parallel and serial concatenated codes. Technical Report 42-127, Jet Propulsion Laboratory, Pasadena, CA, Nov 1996.

[44] R. Scholtz. Optimum CDMA Codes. In *Proceedings of the National Telecommunications Conference*, November 1979.

[45] R. Scholtz. Multiple-Access with Time-Hopping Impulse Modulation. In *MILCOM '93*, Boston, MA, October 11-14 1993.

[46] R. A. Scholtz and et al. Ultra-wideband Radio Deployment Challenges. *PIMRC 2000*, September 18-21 2000. Session 5, Paper 1 (invited).

[47] R. A. Scholtz, P. V. Kumar, and C. J. Corrada-Bravo. Signal design for ultra-wideband radio. In *Proceedings of Sequences and Their Applications*, Bergen, Norway, May 2001.

[48] R. A. Scholtz and M. Z. Win. Impulse Radio. In S. G. Glisic and P. A. Leppanen, editors, *Wireless Communications: TDMA versus CDMA*. Kluwer Academic Publishers, 1997.

[49] M. Simon, J. Omura, R. Scholtz, and B. Levitt. *Spread Spectrum Communications Handbook*. McGraw-Hill, 1994.

[50] O. Y. Takeshita and Jr. D. J. Costello. New Deterministic Interleaver Designs for Turbo Codes. *IEEE Transactions on Information Theory*, 46:1988–2006, Sept 2000.

[51] R. M. Tanner. Minimum-Distance Bounds by Graph Analysis. *IEEE Transactions on Information Theory*, 47:809–821, Feb 2001.

[52] James Taylor, editor. *Introduction to Ultra-wideband Radar Systems*. CRC Press, 1995.

[53] E. L. Titlebaum. Time Frequency Hop signals Part 1: Coding based upon Linear Congruence. *IEEE Trans. Aerosp. Electron. Syst.*, 17:494–501, July 1981.

[54] A. J. Viterbi. Very low rate convolutional codes for maximum theoretical performance of spread-spectrum multiple-access channels. *IEEE Journal on Selected Areas Communications*, 8(4):641–649, May 1990.

[55] B. Vucetic and J. Yuan. *Turbo Codes: Principles and Applications*. Kluwer Academic Publishers, Dordrecht, 2000.

[56] Chen Zhi, Fan Pingzhi, and Jin Fan. Disjoint differfence sets, difference triangle sets, and related codes. *IEEE Transactions on Information Theory*, 38(2):518–522, march 1992.

Appendix A

Clocks and Periods Analysis of Time-Shifted Pulse Modulations Spectrum with Framing Structure

An analysis of the code-dependent factor $C(f)$ in the power spectral density of the pseudonoise-coded signal does reveal a few ideas. Of course, $C(f)$ need only be evaluated at the frequencies of the spectral lines (delta functions) in the density, namely at multiples of $1/T_p$. When $C(f)$ is evaluated at these frequencies, then special results can occur, depending on the particular harmonic and the existence or non-existence of relationships between clock signals. Evaluating $C(f)$ at the k^{th} harmonic generally gives

$$\begin{aligned} C(k/T_p) &= \sum_{n'=0}^{N_p-1} \sum_{n=0}^{N_p-1} \exp\left\{-j2\pi k \left[\frac{(n'-n)T_f}{T_p} + \frac{(c_{n'}-c_n)T_c}{T_p}\right]\right\} \\ &= \sum_{n=0}^{N_p-1} \sum_{i=-n}^{N_p-1-n} \exp\left\{\frac{-j2\pi ki}{N_p}\right\} \exp\left\{\frac{-j2\pi k}{N_p}(c_{n+1}-c_n)\frac{T_c}{T_p}\right\}. \end{aligned}$$

Each of the exponential factors above is periodic in i with period N_p; the first because $\exp(-j2\pi k/N_p)$ is an N_p^{th} root of unity, and the second because $\{c_n\}$ has period N_p. This means that the range of the sum on i can be made independent of n, giving

$$C(k/T_p) = \sum_{i=0}^{N_p-1} \exp\left\{\frac{-j2\pi ki}{N_p}\right\} \sum_{n=0}^{N_p-1} \exp\left\{\frac{-j2\pi k(c_{n+1}-c_n)T_c}{T_p}\right\}. \qquad \text{(A.1)}$$

Notice that there are three basic elements that affect these quantities: (i) the hopping code $\{c_n\}$, (ii) the period of the code N_p and (iii) the slot-to-frame time ratio T_c/T_f. The parameter T_c/T_f in (A.1) is a function of the clock system used to generate the time-hopping modulation. Here are some possibilities.

(a) T_c/T_f is the reciprocal of an integer. For example, this might occur if T_c is the period of the basic clock, and the frame time is an integer number of cycles of this clock.

(b) T_c/T_f is rational. This would occur in situations in which both T_c and T_f are integer counts of the period of a higher frequency master clock. The range of possible numerator and denominator integers in this design is limited by the availability of high frequency master clocks and other design considerations.

(c) T_c/T_f can be any real number within a design range. This might occur in situations in which the slot and frame time generators are independently generated. This is the case in which the frame clock is a stable oscillator and the delay between frame clock and transmission is created by a separate delay device that can address $N_h = 256$ uniformly spaced delays within a time interval $[0, T_{delmax}]$, where $T_{delmax} = N_h T_c$ is adjustable between 7 and 50 nanoseconds.

When the slot-to-frame time ratio is rational as it is in (a) and (b), then there exist positive values of the integer k such that $C(k/T_p)$ takes on its maximum possible value, namely N_p^2. In case (a) this will happen when k/T_p, is any integer multiple of the slot frequency $1/T_c$. In case (b) this will happen when k/T_p is any integer multiple of the master clock frequency. Of course the monocycle energy density also weights the spectral line, and if these problem lines can be placed at a high enough frequency where $|W(k/T_p)|^2$ is small, then spectral smoothing can be accomplished. Case (c) does not have these obvious problem frequencies, but after a code is selected, it would seem prudent to review the exact choice of the slot-to-frame time ratio to make the power spectral density as low and smooth as possible.

Another simplification of the equation (A.1) occurs when k is an integer multiple of N_p, i.e., $k = k'N$, and hence $k/T_p = k'/T_f$. Then

$$\begin{aligned} C(k/T_p) &= \left| \sum_{n=0}^{N_p-1} \exp\left\{ -j2\pi k' c_n \frac{T_c}{T_f} \right\} \right|^2 \\ &= \left| \sum_{m=0}^{N_h-1} C_m \exp\left\{ -j2\pi k' m \frac{T_c}{T_f} \right\} \right|^2 \end{aligned}$$

where C_m, is the number of times that $c_n = m$ as m ranges from 0 to $N_p - 1$. Thus lines at multiples of the frame frequency $1/T_f$ only depend on how many times c_n, takes on particular values, and not on their ordering within the code period. These lines are a result of the framing constraints because the factor $C(f)$, in this case, looks very much like the factor $\left| \sum_{m=0}^{N_h-1} C_m \exp\{-j2\pi f m T_c\} \right|^2$ from our framed random time hopping analysis, evaluated at the line frequencies $f = i/T_f$. From this we can conclude that a good design should have all C_m, at nearly the same value.

Appendix B

From Finite Fields to VHDL