Assignment of Spreading Codes in Code Division Multiple Access Radio Systems

Samuel Oluwasesan Sanusi

Division of Mathematics and Statistics
School of Technology
University of Glamorgan

A submission presented in partial
fulfilment of the requirements of the
University of Glamorgan/Prifysgol Morgannwg
for the degree of Doctor of Philosophy

June 2006.

Acknowledgements

I would like to express my sincere gratitude to my supervisors Prof. Derek Smith and Dr. Stephanie Perkins for being extremely generous with their time, knowledge and ideas. I have enjoyed great freedom throughout the period of this research work.

My appreciation also goes to Mr. Glyn Wyman and Dr. Richard Jones both of BAE Systems, for their immeasurable support and contributions to the success of this work. I am also grateful to Dr. Ifiok Utong of the School of Electronics for taking the time to explain some useful engineering terms. I would like to thank Dr. Kari Karkkainen of the University of Oulu for sending a copy of his thesis to me.

I want to express my appreciation of the management of the Division of Mathematics and Statistics for giving me the opportunity to develop myself as a teacher. My best regards also to the administrative staff of the School of Technology. Finally to my colleagues who have shared my working time in room G501.

Abstract

This thesis is concerned with spreading code construction and assignment in Direct Sequence Code Division Multiple Access radio systems. Particular emphasis is given to Quasi-Synchronous Code Division Multiple Access. The frequency spectrum required for these systems depends on the length of the spreading code used. Frequency spectrum is an important and expensive resource of a radio system. Short spreading codes are desirable but can lead to an inadequate number of codewords, requiring undesirable codeword re-use. It is shown that careful assignment of spreading codes can effectively minimise interference, manage the spreading codewords available and ultimately ensure that the spectrum is used efficiently.

Interference in Direct Sequence Code Division Multiple Access systems depends on the correlations of the spreading code used. Spreading codes are designed to have low correlation. It is possible to achieve lower correlations (or even zero correlations) within a window in a quasi-synchronous system than is possible in an asynchronous system. The problem of limited availability of codewords can be overcome using a scrambling code, but the low inter-cell correlations are then lost.

As an alternative, it is shown that lower interference can be achieved for a given length of spreading code if assignment techniques which take account of both correlations and codeword re-use are adopted. This is particularly true in a quasi-synchronous system. An interference measure based on the signal-to-interference ratio is adopted, similar to that recently proposed for frequency assignment algorithms. The interference component of this measure depends on the aperiodic correlations of the code used.

The codes compared for assignment include well known Gold, Kasami and Kerdock codes as well as modifications of Lin Chang Simplex and Loosely Synchronous codes. These last two codes are designed for quasi-synchronous operation. The modifications double the number of codewords, at a cost of allowing some higher correlations. Tabu search and simulated annealing algorithms of the type used for frequency assignment are adapted for spreading

code assignment. They are able to avoid or minimise interference arising from these higher correlations.

The modified Lin Chang Simplex and (in particular) Loosely Synchronous codes are shown to present better system performances (in terms of higher signal-to-interference ratios and greater network coverage) than the combination of Walsh Hadamard code and Gold code used in third generation mobile telephone systems. They are also better than the other codes considered. This is true for both mean square and peak planning, and is particularly true for the most demanding networks considered.

Contents

List of Figures

List of Tables

List of Symbols and Abbreviations

$AIC\%$	-	Average Inter-Code percentage
AIP	-	Average Interference Parameter
$ACC\%$	-	Average Co-Code percentage
BER	-	Bit-error-rate
$C(x, y)$	-	Aperiodic correlation
$C_{\max}$	-	Peak aperiodic correlation
$d_{r,i}$	-	Distance between receiver r and transmitter i
DS-CDMA	-	Direct Sequence Code Division Multiple Access
DSSS	-	Direct Sequence Spread Spectrum
FDMA	-	Frequency Division Multiple Access
FHSS	-	Frquency Hopping Spread Spectrum
FM	-	Frequency Modulation
GF	-	Galois Field
GO	-	Generalised Orthogonal
GQO	-	Generalised Quasi-Orthogonal
$IP_{\max}$	-	Maximum Interference Potential
ISI	-	Inter-Symbol-Interference
LCS code	-	Lin-Chang Simplex code
LFSR	-	Linear Feedback Shift Register
$L_{r,i}$	-	The signal path loss of transmitter i at reception point r
LS code	-	Loosely Synchronous code
MAI	-	Multiple Access Interference
$MSA\hat{Z}$	-	Mean Square of Autocorrelation of $\hat{Z}$
$MSC\hat{Z}$	-	Mean Square of Crosscorrelation of $\hat{Z}$
P_i	-	The signal power of transmitter i
PN	-	Pseudo Noise
$PSA\hat{Z}$	-	Peak Square of Autocorrelation of $\hat{Z}$
$PSC\hat{Z}$	-	Peak Square of Crosscorrelation of $\hat{Z}$
SIR	-	Signal-to-Interference Ratio
SNR	-	Signal-to-Noise Ratio
TDMA	-	Time Division Multiple Access

WCDMA	-	Wideband Code Division Multiple Access
τ	-	Signal time delay
$\tau_{\max}$	-	Maximum signal time delay
$\theta(x, y)$	-	Even (periodic) correlation
$\hat{\theta}(x, y)$	-	Odd correlation
$\theta_{\max}$	-	Peak even (periodic) correlation
$\hat{\theta}_{\max}$	-	Peak odd correlation
$\hat{Z}$	-	Output of a correlation receiver
$\hat{Z}_{x,y}$	-	Interference part of the output of a correlation receiver
ψ_{x,y_j}	-	Value of any of $MSC\hat{Z}, MSA\hat{Z}, PSC\hat{Z}$ or $PSA\hat{Z}$ between wanted spreading codeword x and unwanted spreading codeword y_j assigned to an interfering transmitter j
$\Phi_{x,y}$	-	Estimated measure of $\hat{Z}$ required for a problem at a certain SIR threshold
σ	-	SIR threshold
σ'	-	Code reuse SIR threshold
$\sigma_{\min}$	-	Minimum code reuse SIR achieved by an assignment
λ	-	Penalty for unsatisfactory code reuse during optimisation process

Chapter 1

Introduction

1.1 Spread-Spectrum radio systems

The history of Spread Spectrum dates back to the mid-1950's. Spread spectrum is a method of transmitting signals by spreading them over a range of frequencies wider than the bandwidth that is normally required to transmit a message signal [60], see also [62, 63]. The early applications of spread spectrum were for military communications [49, 69]. Spread Spectrum was attractive for military antijamming tactical communications, combating signal multi-path, protecting signals against eavesdropping, etc. Another advantage of Spread Spectrum is its multiple access capability [11, 60, 81]. All these advantages and its ability to co-exist with existing commercial applications have made spread spectrum communications popular for commercial applications. The first commercial application of spread spectrum is IS-95. The current third generation mobile telephone system is developed on spread spectrum technology [3, 12, 49, 52] . It has also been extensively studied for future generations of mobile telephone systems.

Two most common methods of spreading signals are *Direct Sequence* Spread Spectrum (DSSS) and *Frequency Hopping* spread spectrum (FHSS). In DSSS, each signal is spread by the use of a pseudo-random sequence (called a *spreading codeword* [1]) to give a spread spectrum signal. This results in the spread spectrum signal having a statistical nature similar to noise. The receiver is able to recover the original narrow band message signal by generating an exact replica of the spreading code used for spreading at the transmitting end and performing an inner product (for signals that are either $+1$ or -1) with the received spread spectrum signal. This collapses the bandwidth of the received spread spectrum signal to give a narrow band signal (the original message signal)

[1] Each digit of a spreading codeword is called a chip. A set of spreading codewords is called a *spreading code*.

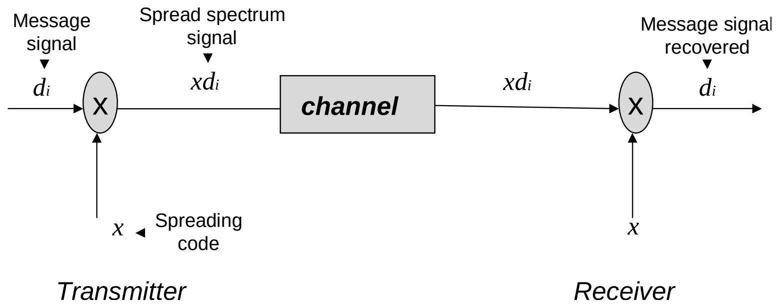


Figure 1.1: *A simple illustration of a Direct Sequence Spread Spectrum (DSSS) system.*

[13, 60, 68, 69, 81]. This process is graphically displayed is figure 1.1 and figure 1.2. In FHSS, a message signal is allowed to hop over a pseudo-random set of frequencies. Codes are used to design the frequency hopping set. An intended receiver is able to recover the message signal by locking-on to the set of hopping frequencies assigned to its transmitter.

The introduction of spread spectrum into commercial applications has made resource management a critical issue. More users are supported in commercial applications that in most military applications. Examples of resources that need to be managed are the frequency spectrum, the *spreading codes* and transmitter power [60, 81, 82]. The number of *codewords* available in a spreading code determines the number of users that could be supported by the system. For most spreading codes, e.g. Gold codes, the number of spreading codewords in a spreading code class depends on the length of the spreading code. In Spread Spectrum systems, the frequency spectrum needed is a function of the spreading code length used. A critical resource management issue arises when a large number of users are to be supported with limited resources.

There are other applications used in the past that appear to spread signals over a range of frequency spectrum without the use of spreading codes [60] (e.g. frequency modulation, FM and Pulse-code modulation, PCM). Such systems are not examples of spread spectrum systems. For a system to be classified as a spread spectrum system, spreading must be independent of the signals. For example in direct sequence code-division multiple-access, the spreading code used for spreading is independent of the signal and it is pseudo-noise in nature.

1.2 Code-Division Multiple-Access

Multiple access defines the method by which more than one user is allowed to share the available resources. Three major multiple access techniques available

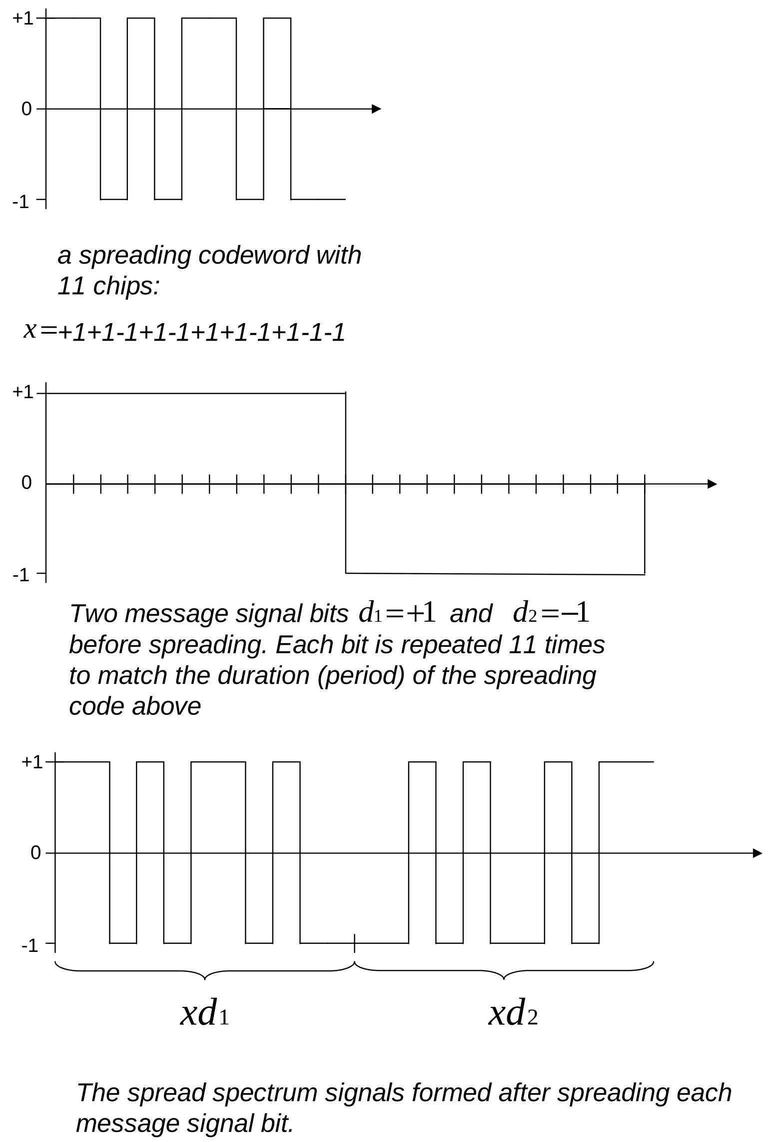


Figure 1.2: *An illustration of a spreading codeword, two message bits and the spread spectrum signal formed.*

in the literature are Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA). The multiple access capability of Direct Sequence spread spectrum communication is of the latter type. In FDMA, each user signal is transmitted over a single or narrow band channel (frequency) which is sufficiently separated from adjacent users' channels in order to avoid (or reduce) interference. A frequency can be re-used when two users are a sufficient distant apart. In TDMA, signals are transmitted over the entire frequency bandwidth at different times. In DS-CDMA, all signals are transmitted and spread over the same available wide band frequency spectrum at the same time. To differentiate signals, each transmitter is assigned a spreading codeword (spreading sequence) which is different from the spreading codewords assigned to close transmitters (users). An intended receiver is able to differentiate its wanted spread spectrum signal from other unwanted spread spectrum signals as a result of the correlations between the spreading codewords used. See figure 1.3 for a simple illustration.

Consider a DS-CDMA system with two transmitters, each with a single receiver. Suppose message signal $d(\tau_1)$, with time reference τ_1, is spread using spreading codeword x to give spread spectrum signal $xd(\tau_1)$, and message $d'(\tau_2)$ with time reference τ_2, is spread using spreading codeword y, to give a spread spectrum signal $yd'(\tau_2)$. Suppose both signals are transmitted with equal powers and that the channel is considered to be a perfect channel, that is without channel noise and multi-path. The two signals combine on the channel to give $xd(\tau_1) + yd'(\tau_2)$. Let $R_{(\tau_1)}$ represents a synchronous receiver for the first transmitter with time reference (τ_1), in which case spreading codeword x is the wanted code. The output of the correlator attached to $R_{(\tau_1)}$ is:

$$R_{(\tau_1)} = d\delta_{x,x}(0) + d'\delta_{x,y}(\tau). \tag{1.1}$$

where $\delta_{x,y}(\tau)$ is the correlation between spreading codewords x and y and $\tau = \tau_2 - \tau_1$. The first term contains the wanted signal, d and the correlation, $\delta_{x,x}(0)$, between wanted spreading codeword x and its replica, assuming chip [2] synchronisation. The second term contains the interfering signal d' and the correlation, $\delta_{x,y}(\tau)$, between the wanted spreading codeword x and the unwanted spreading codeword y with relative time difference τ. Note that here correlation refers to normalised correlations between 0 and 1 with $\delta_{x,x}(0) = 1$. Now, if the value $\delta_{x,y}(\tau)$ in equation (1.1) is zero then the interference contribution due to multiple access is zero. In this case, spreading codewords x and y are orthogonal, i.e. completely uncorrelated. Correlation values close to zero are in some cases adequate for interference in a DS-CDMA system to be tolerable. It is easy to see that interference due to multiple access (or *multiple access interference* (MAI)) is mitigated by designing spreading codes to have low *crosscorrelation*[3]. In the same way, interference from signals due to multi-path or *Inter-Symbol-Interference* (ISI) is mitigated using low off-peak autocorrelation [4] (definitions are given in chapter 2). Interference in DS-CDMA then depends on the correlation properties of the spreading code used. A limiting factor on non-orthogonal spreading codes is that interference increases with the number of users.

[2]each digit of a spreading codeword is called a chip
[3]correlation between two different spreading codewords
[4]correlation between a spreading codeword and its delayed version

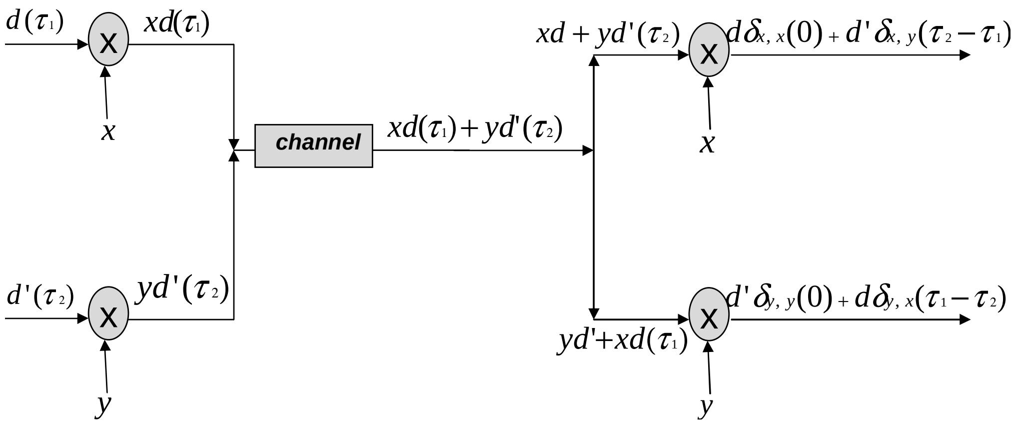


Figure 1.3: *A simple illustration of a Direct Sequence Code Division Multiple Access (DS-CDMA) system with two transmitters, each with one receiver. The channel is assumed to be free of noise and multi-path.* τ_1 *and* τ_2 *are the different time references of spread spectrum signals* $xd(\tau_1)$ *and* $yd'(\tau_2)$ *respectively.*

1.3 Spreading codes for DS-CDMA systems

Spreading codes are one of the important resources of a DS-CDMA system. Each element of a spreading code (or spreading code class) is called a *spreading codeword*. Each digit of a spreading codeword is called a *chip*. A spreading codeword is used for spreading the signal at the transmitting end and for despreading the signal at the receiving end. Interference from signals of other users is a function of the correlation between the respective spreading codewords. The choice of a good spreading code is therefore important for DS-CDMA system planning. The choice depends on:

1. The low correlation of the spreading code - which implies tolerable interference in the system.

2. The implied length of the spreading code - the frequency spectrum required for a DS-CDMA system is directly proportional to the length of spreading code used.

3. The number of codewords available in a spreading code - the number of spreading codewords available in a spreading code determines the number of users that could be supported by the system, even with re-use of spreading codewords when tolerable.

These three properties of a spreading code need to be considered when deciding on a suitable spreading code for a particular DS-CDMA system. A spreading code may have low correlation but with too few spreading codewords or the codewords may be too long for the frequency spectrum available.

Spreading codes are classified based on the level of system synchronisation. System operations in DS-CDMA are classified into *synchronous*, *asynchronous* and *quasi-synchronous* operations. In a synchronous CDMA operation, the i^{th} chip of all signals arrive at the receiver at the same time. One example is a satellite CDMA system. Another example is the downlink transmission (transmission from a base station to a user equipment) in a single cell of a CDMA cellular mobile telephone system [13]. All signals emanate from their respective transmitters with the same timing references. In an asynchronous system, the timing references of signals are essentially lost. An example is the uplink transmission (i.e. signals transmitted from a user device to a base station) of a CDMA cellular mobile telephone system. Equally, interfering signals from other base stations different from the base station of the receiver, in the downlink of a cellular CDMA system, are asynchronous in nature. The receiver does not have a knowledge of the timing references of such signals. Another level of system synchronisation is quasi-synchronous operation. In a quasi-synchronous operation, signals are received with different timing references but for all receivers there is a maximum timing uncertainty of any signal in the network.

In the downlink of a CDMA cellular system, it is possible to maintain synchronisation among signals within the same cell as all the signals are transmitted from the same base station and path lengths do not differ. The main example of a spreading code available in the literature for synchronous operation is a Walsh-Hadamard code. A limiting factor affecting synchronous operation is the imperfect synchronisation with delayed versions of signals in a multi-path channel. This introduces a non-zero correlation (and therefore more interference) into the system.

Examples of spreading codes for asynchronous operation are Gold codes, Large Kasami sets and Small Kasami sets. These spreading codes have low non-zero correlation properties. According to the *Welch bounds* [83] and other theoretical limits, it is impossible to design spreading codes for asynchronous operation with perfect zero correlation. This explains the interest in new spreading code designs for quasi-synchronous operation [17, 78].

In a Quasi-Synchronous CDMA (QS-CDMA) system, the relative time uncertainties (or delays) of interfering signals are confined to take only small values. There may be a knowledge of a global clock, such as a GPS clock, so that when chip i is sent the actual chip received is within the uncertainty range of $\{i - \tau_{\max} + 1, \ldots, i + \tau_{\max} - 1\}$, where τ_{max} is a small integer. Spreading codes could be designed to have zero or tolerable low correlation within such intervals to combat interference due to multiple access and multi-path. A limiting factor on spreading codes for quasi-synchronous operation is that the number of codewords available is still bounded by a variation of the Welch bound, [36, 80, 83]. Specifically, the number of codewords that could be constructed for a spreading code depends on the length of the spreading code and the interference free window desired [17, 79, 78]. Spreading codewords will have to be satisfactorily re-used in order to support a large number of users, such as in a DS-CDMA commercial cellular system.

1.4 DS-CDMA receivers

Receivers for DS-CDMA systems are a little more complex in design than those used in FDMA and TDMA. Components of a DS-CDMA receiver include a correlator and a filter. The correlator generates an exact replica of the spreading codeword used in spreading the wanted signal in order to despread it. When the spreading codeword is multiplied by the interfering signals, there are no recognisable symbols obtained. The filter allows the signal power of the wanted signal to go through unscaled but scales down the signal power of the unwanted signals by a factor of the square of the crosscorrelation observed (scaled between 0 and 1). The ratio of the filter output of the wanted signal power to the sum of the filter outputs of unwanted signal powers is called the *signal to interference ratio* of the receiver.

DS-CDMA receivers are classified based on their response to signals due to the multi-path property of the channel. It will be necessary to note that in a DS-CDMA system, fading becomes more frequency selective as the spreading bandwidth increases [11, 81]. Reflected signals on different frequencies have different phases and paths. Two basic types of DS-CDMA receivers that will be discussed here are the *correlation receiver* and the *rake receiver*.

A correlation receiver has only one correlator. It treats signals due to multipath as independent interference or as noise. In a rake receiver, some delayed versions of the wanted signal are independently resolved or despread and later their powers are combined in order to increase the *Signal-to-Interference Ratio* (SIR) or *Signal-to-Noise Ratio* (SNR). In IS-95, a rake receiver has three or more fingers for resolving delayed signals due to multipath. Each finger consists of a correlator and is assigned to a multipath from which a relatively strong multipath signal is expected. The use of a rake receiver increases DS-CDMA capacity but a correlation receiver, which is the basic receiver, is used for the receiver model constructed for this work. There may be disadvantages in using a rake receiver when codeword re-use must be tolerated.

1.5 Code acquisition techniques in existing CDMA systems

CDMA systems must be able to synchronise a local replica of the spreading code with the one used by the transmitter. This process is known as code acquisition. There are some common features in the way that this is done and some features are unique to each CDMA system.

The downlink of cellular CDMA systems may consist of several channels, known as the pilot channel, the synchronisation channel, the paging channel and the traffic channel [21]. Prior to the transmission of user information or data, the spreading codeword of the base station is first sent through the pilot channel. A mobile receiver identifies the base station with the maximum signal power, demodulates the pilot code and performs channel estimation using the pilot code. These channel estimations include phase reference of the code, coherent demodulation of other channels and measurement of instantaneous interference (or SIR).

In IS-95, the pilot code has a period of 32,768 chips [13]. Pilot codes of other base stations consists of unique phase offsets of the same m-sequence (see section 3.1). Code offsets between base stations are adjusted by global positioning system (GPS) timing. The default value of code offset is 64 chips in duration. The receiver only needs to find the phase of the m-sequence (pilot code) used by the base station. This is achieved by correlating the pilot signal with the local m-sequence generated at the receiver. The phase which results in the peak auto-correlation is the phase which gives the codeword of the base station. If the receiver is a rake receiver, the same process is performed with every multipath signal. The rake receiver then tracks the amplitude and phase of the multipath pilot code.

In W-CDMA, the pilot code is a Gold code (see section 3.2) truncated to length 2^{16} [13] which is a part of a Gold code with period $2^{41} - 1$. Each base station is assigned a unique Gold codeword. This code is known as a scrambling code as it does not actually perform spreading. Code acquisition is more complex than in IS-95. The number of Gold codewords is fixed at 512. The code is divided into groups (usually 32 groups). The receiver first demodulates the synchronisation channel in order to synchronise the pilot code frame size and the code group. This is done by correlating the synchronisation signal with all the 512 codewords. Since the cyclic shifts of the code are unique, the code group and the frame synchronisation are determined. The phase and amplitude of the pilot code is then determined by correlating the pilot code with all possible phases of the codewords in the code group already identified [3]. A threshold value is used to decide whether the code has been identified and the synchronisation timing uncertainty range is determined. This is further fine tuned as the communication between the transmitter and receiver goes on.

Another channel used in CDMA systems is the paging channel. The paging channel provides broadcast information, paging registration, traffic channel assignment, short message service and authentication. Examples of the broadcast information transmitted through this channel include phase offset information of neighbouring base stations and system time. Once the mobile device detects a page channel, it sends a message of origination and authentication. The base station then assigns the traffic channel and the receiver continuously monitors (or tracks) transmission timing of the frame within one chip.

1.6 The model used for quasi-synchronous CDMA

The proposed model for quasi-synchronous CDMA system is quite different from the models used for W-CDMA. In the downlink, each base station is assigned a number of spreading codewords corresponding to the number of transmitters available at the base station. Each pilot channel will then have a different pilot code. Another complexity is the possible high values of correlation outside the low (or zero) correlation zone. For example, suppose that the maximum correlation value is about half of the peak auto-correlation value. This may lead to a false alarm i.e. detecting the wrong signal. Xiaoxu *et al.* in [85] proposed a two-stage search scheme to avoid this problem. In the first stage, a code phase is predicted and then coarsely searched. The second stage involves a fine search of the code phase using the unique feature of the code. The method requires a prior knowledge of the code beyond the zero or low correlation zone to formulate an appropriate threshold on the correlation output.

Another complexity in the implementation of a quasi-synchronous CDMA system is the dependency on GPS for base station timing. GPS has been successfully used in maintaining an external timing and hence to synchronise base stations. However, sole dependency on a US-owned GPS system for synchronisation is a major concern. A GPS synchronised system also suffers from loss of service in underground or deep in-building deployment. Self-synchronisation of base stations is an alternative [7, 9]. Each receiver, after maintaining synchronisation with the pilot signal, broadcasts the system timing (system frame number) of the base station as measured at the receiver through the broadcast channel. The network can filter measurements received from several mobile users to form an estimate of the relative time difference between base stations. Each base station then broadcasts the relative time difference with its neighbours in its System Information Message. The relative timing uncertainty introduced by this method can be reduced by allowing the base station to increase or decrease its timing periodically until the timing difference between all base stations is sufficiently small.

1.7 The transmission and reception process in W-CDMA systems

The process involved in the transmission and reception of W-CDMA signals includes protection of message signals against radio propagation impairments, performing spread spectrum modulation and providing multiple access [34]. At the transmitting end, the message signal is first protected using a forward error control code (convolutional code). Each 10ms frame is then time-interleaved to protect against burst errors. The message is Binary Phase Shift Keying (BPSK) spread using a Walsh Hadamard codeword. It is also scrambled by Quaternary Phase Shift Keying (QPSK) modulation. The spread signal is then power amplified before transmission in the 2 GHz carrier frequency [3]. At the receiver de-scrambling is achieved using a replica of the Gold codeword and de-spreading is performed using a replica of the Walsh orthogonal codeword. If the receiver is a rake receiver each multi-path channel already recognised during code acquisition is correlated with each finger of the rake receiver. The correlator output SIR is then measured and compared with the target SIR [3]. After SIR measurement, the correlator output is transformed into a soft-decision message sample and de-interleaved for further soft-decision Viterbi decoding to recover the message signal.

1.8 Spreading codes and spreading code assignment - Literature review

1.8.1 Literature review of spreading code design

The major motivation behind different advances in spreading code design is the improvement of correlation properties. Spreading codes are generally designed in such a way that:

1. The correlations between a spreading codeword and time shifted (or delayed) versions of itself take on small values.
2. The correlations between any pair of codewords from a spreading code take on small values.

Sequences with these properties are referred to as *pseudo-random* or *pseudo-noise* sequences (PN sequences) [68]. The word "sequence" is mostly used interchangeably with "codeword". The history of spreading codes spans over 50 years. The first spreading code to be published was the binary *Maximal-length* sequence (m-sequence). The theory of m-sequences was developed and applied by Gilbert, Golomb, Welch, Zierler. The details of this theory are well documented in [27, 68, 70, 89]. At this time, much of the research was concerned with the autocorrelation properties and the pseudo-random nature of m-sequences [14, 26, 27, 58, 72].

Sarwate and Pursley in [68] and some early authors, gave attention to the crosscorrelation properties of m-sequences. In particular, methods of selecting sets of m-sequences with good crosscorrelation properties were developed by the late 1960's [25, 38, 68]. Some of the resulting spreading codes are Gold codes, Large sets of Kasami sequences, Small sets of Kasami sequences, Gold-like sequences and Dual-BCH sequences. These code families are designed for asynchronous operation. They are pseudo-random and periodic in nature. They can be generated using a *linear feedback shift register* (LFSR) with finite binary state. Equally, purely aperiodic codes exist, e.g. Barker and Golay complementary sequences [24, 26, 59], but they can be repeated in a periodic way [39].

The increasing interest in spread spectrum communication for commercial applications, especially cellular mobile systems, led to consideration of code families for synchronous operation. The most common of such codes is the Walsh-Hadamard code derived from the rows of a Hadamard matrix [13, 19, 20, 22]. These codes lose their orthogonal nature when synchronisation cannot be maintained, for example, in the case of a multipath channel and in the uplink of a cellular mobile telephone system. For this reason, there is a need to design spreading codes with zero or close to zero correlation properties over some signal time delays (measured in chips) in order to mitigate interferences due to the multipath property of the channel and asynchronous multiple access. Equally, it was found to be theoretically impossible to design spreading codes with zero correlation over all possible time delays [17]. To meet this challenge, Generalised Orthogonal (GO) and Generalised Quasi-Orthogonal (GQO) codes were proposed for Quasi-Synchronous CDMA (QS-CDMA) systems. Examples of such codes are described in [8, 15, 17, 18].

Generally, spreading codes are designed to be either binary, complex or ternary in nature. M-sequences, Gold codes, Large sets of Kasami and Small sets of Kasami are examples of binary codes. Examples of complex codes are the complex 4-phase sequences described by Bostaz et. al in [5] and the Song-Park (SP) sequences described by Park et. al in [57]. Examples of ternary codes are Large Area (LA) codes [8, 43, 44] and Loosely Synchronous (LS) codes [78].

These codes have been proposed for QS-CDMA operation.

Other important advances in spreading code design are the derivation of correlation bounds for spreading codes by Sidelnikov [71] and Welch [83] as part of the code selection criteria. The realisation of the importance of the *odd correlation* in [47] is also worth mentioning. The development of system analysis using *average Signal-to-Noise Ratio* (SNR) as a function of *Average-Interference-Parameter* (AIP), which in turn is a function of the aperiodic correlation [63], brought about a different code selection criterion from the usual maximum modulus of correlations.

1.8.2 Literature review of spreading code assignment

As it has been noted in the previous sections, spreading codes are essential components of a DS-CDMA system. Interference between users is a function of the correlation properties of the spreading code used. In small networks, spreading code assignment is just a simple code selection problem. Only a small number of codewords may be needed and these may be short (requiring only limited spectrum). In large networks, for example commercial CDMA networks, a large family of codewords is desirable. An increase in the code family size implies an increase in the code length for most spreading codes available in the literature.

To avoid excessive use of the implied frequency spectrum, a re-use of spreading code approach has been proposed. In [32, 37, 84], spreading code assignment is modelled as a graph colouring problem. In this case, crosscorrelations of a spreading code were taken to be equal for all pairs of codewords. Re-use of spreading codewords is subject to a satisfactory re-use distance. Arbitrary spreading code families with equal crosscorrelations were used to test the algorithms proposed. This type of spreading code assignment problem is similar to a frequency assignment problem with co-channel constraints only. This approach may be useful when the maximum of the correlations of a code family (for example the *even correlations* of Gold codes, Small Kasami sets and Large Kasami sets) is used for network planning (in which re-use of spreading codewords is necessary). Network planning using the even correlations can be considered as planning a network performance 50% of the time. Of equal importance are the odd correlations (which occur 50% of the time) and the aperiodic correlation (which is the basic correlation) [68]. A similar straightforward graph colouring type of assignment problem arises with the ternary Loosely Synchronous (LS) code (described by Stańczak et. al in [78]) and Lin-Chang-Simplex code (LCS) for quasi-synchronous operation (based on the construction by Lin and Chang in [45] and modified by Jones et. al in [35]). The LS code has a zero correlation window while a single LCS code has an absolute value of 1 for (unnormalised) even correlation within the low correlation window. Fan et. al in [17] and Jones et. al in [36] independently showed that these code families are limited by a variation of the Welch bound.

For most families of codes (e.g. Gold codes, Small Kasami sets and Large Kasami sets) the odd correlations and the aperiodic correlations vary for different pairs of codewords. In this case, the problem is then potentially different from a straightforward graph colouring problem. This type of problem was considered in [36]. The authors considered spreading code assignment problems with variable correlations using a simulated annealing algorithm. A careful selection of codewords from a Large Kasami set was carried out. Penalties were defined for transmitter pairs. For example, define $\{1000, 10, 1, 0\}$ as the penalty list for a pair of close transmitters. Penalties $0, 1, 10$ are associated with low, high and very high correlations respectively. A penalty value of 1000 is used when a codeword is re-used in an unsatisfactory way. A similar problem arises with the polyphase codes described in [57]. The even correlations between pairs of codewords are zero but the odd correlations (and consequently, the *aperiodic correlations*) vary from 2 to approximately $\frac{N}{\pi}$, where N is the length of code.

Equally, in [35], in order to avoid unsatisfactory re-use of LS codes (as described in [78]) and LCS codes (as described in [45]) in relatively large networks and consequently to conserve the frequency spectrum, a reasonable trade off on the zero correlation properties (with LS codes) or low correlation (with LCS codes) for increase in code size was proposed. For LCS codes another set of these code classes is introduced. The zero or low correlations are maintained between pairs of codewords within the same set but are worse for pairs of codewords between sets. Ideally, a pair of transmitters that are close together will be assigned a pair of codewords within the same set and a pair of transmitters that are farther apart could be assigned a pair of transmitters from different sets. A codeword may be re-used between transmitters that are much farther apart. A similar extension of LS codes is also proposed. The spreading code assignment method proposed to solve this problem is similar to that of [36]. No penalty is incurred between a pair of codewords from the same set. A penalty is associated with a pair of codewords from different sets and a much more stringent penalty is incurred when a codeword is re-used.

The work presented here is concerned with the assignment of spreading codes with similar correlation properties to those addressed in [36] (i.e. the correlations vary for different pairs of codewords). Penalties are modelled in terms of the Signal-to-Interference Ratio (SIR).

1.9 Objectives of this work

The main objective of this work is concerned with the potential adaptation to spreading code assignment of the very successful approaches used in the past for frequency assignment problems. Enormous time and money have been devoted to developing algorithms for frequency assignment problems. A comprehensive survey of relevant literature is available in [1, 74]. Some recent work can be found in [77, 50]. It is necessary to explore the potential benefits of adapting successful methods to code assignment.

Frequency assignment based on an SIR measure has been successfully used (and is still of current interest). This method was found to be a more accurate approach than methods based on binary constraints. A set of *reception points* (usually at boundaries of cells) at which services are required is specified. At each reception point, the *signal strength* of the wanted signal and the signal strengths of all the interfering signals (unwanted signals) are made available to the algorithm [6, 33, 74]. SIR is measured as the ratio of the signal strength of the wanted signal to the sum of all the interfering signals. The cost associated with the ratio is zero if the corresponding SIR is more than a user set threshold. Otherwise, a cost, determined in some way (usually as a function of the SIR at that reception point), is incurred. This cost is then summed over all reception points.

In frequency assignment problems, adjacent transmitters are assigned adequately spaced frequencies (channels) in order to reduce interference. Close channels can be assigned to transmitters that are a sufficient distance apart. A channel is re-used over a satisfactory re-use distance. In DS-CDMA, interference depends on the low correlations of the spreading code used. If these correlations differ from pair to pair, then, to reduce interference, a pair of close transmitters should be assigned a pair of codewords with low correlations. Let x and y represent a pair of codewords of length N. Codewords x and y may be assigned to a pair of adjacent transmitters, if their corresponding correlations, $\delta_{x,y}(\tau)$, are low (i.e. with normalised absolute values close to zero) for all $1-N \leq \tau \leq N-1$. A pair of codewords with high correlations can only be assigned to a pair of transmitters that are sufficiently far apart. A codeword may be re-used if a user set codeword re-use threshold for SIR is exceeded.

In the work presented here, a cost function based on SIR is used. A correlation receiver (which is the basic receiver) is used for the receiver design. The emphasis will be on types of spreading codes that have been proposed for 4th generation mobile telephone systems and other radio systems. These codes exhibit variations in aperiodic correlations.

1.10 Contributions of this work

The generations of CDMA systems can be differentiated based on the level of synchronisation. These range from synchronous to asynchronous with the intermediate concept of quasi-synchronous. Spreading codes used in these practical systems are designed to have uniform (or uniform occurrence of) correlations. In this case, the peak and mean of correlations will be approximately the same for all pairs of codewords. The usual aim is to design spreading codes with correlations that come close to the Welch bounds. Such codes do not require careful code assignment. However, there exist some spreading codes (which have not been implemented in practical systems) with non-uniform (or non-uniform occurrence of) correlations. These have mainly low correlations, but some high correlations for pairs of codewords that need not be assigned where there is any major potential for interference. The work presented here:

1. Describes the implementation of these codes for the first time.

2. Demonstrates how to assign these codes in such a way that the variation in correlation is exploited to achieve better system performances (measured in SIR and area coverage) than codes in use today. This is a major contribution to the design and planning of CDMA systems.

3. Gives an estimate both of the number of codewords necessary for a satisfactory code re-use and also of the necessary length required for such codes in a particular network.

1.11 Discussion

This chapter has briefly discussed the long history of spread spectrum systems in the military and more importantly the reason why it is attractive for current and future generations of mobile telephone systems. A particular emphasis was placed on the multiple access capabilities of Direct-Sequence Spread Spectrum System, Code-Division Multiple-Access (CDMA). Two important components of DS-CDMA systems discussed are the spreading codes and DS-CDMA receivers. The low correlations, the length and the size of the spreading code are the properties of a spreading code that need to be taken into consideration when choosing a spreading code for a DS-CDMA system. Spreading codes have been classified based on their correlation for different levels of system synchronisation. There are spreading codes for synchronous, asynchronous and quasi-synchronous operation. DS-CDMA receivers have been classified based on their response to signals due to multipath - two types are the correlation receiver and the rake receiver.

A re-use of spreading codewords is subject to a satisfactory re-use threshold. Spreading code assignment is potentially a straightforward graph colouring problem when the correlations of spreading codewords are the same for all pairs. This is analogous to a frequency assignment problem with co-channel constraints only. A more difficult assignment problem arises when correlations of spreading codewords differ from pair to pair. This work will concentrate on the latter type by modifying some of the algorithms that have been successfully used in the past for frequency assignment problems using a cost function based on Signal-to-Interference Ratio (SIR).

Chapter 2

Spreading Codes: Definitions and Properties

2.1 Code Definitions

This section reviews some mathematical definitions and theories that are relevant to spreading codes.

A *binary block* code of length N is a code having all its codewords (binary vectors) of the same length N. The *weight* of a codeword is the number of non-zero components. The *(Hamming) distance* between two words is the number of positions in which they differ.

A *cyclic shift* Tx of a codeword $x = (x_0, x_1, \ldots, x_{N-1})$ is the word $(x_1, x_2, \ldots, x_{N-1}, x_0)$ obtained from x by taking the first digit of x and moving it to the end, all other digits shifting one position to the left. Further cyclic shifts $T^k x$ are obtained by applying T k times. i.e.

$$T^k x = (x_k, x_{k+1}, \ldots, x_{N-1}, x_0, x_1, \ldots, x_{k-1}) \quad 0 \leq k \leq N-1. \tag{2.1}$$

It is easy to see that $T^N x = x$.

Similarly $T^{-1}x$ is obtained by taking the last digit of x and moving it to the beginning, all other digits shifting one position to the right. $T^{-k}x$ is obtained by applying $T^{-1}x$ k times. A useful relation [68] is:

$$T^{-k} = T^{N-k} \quad 0 < k \leq N. \tag{2.2}$$

2.2 Spreading code properties

2.2.1 Basic code properties

Spreading codes are generally designed to reflect the following basic signal behaviours:

1. Each signal is easy to distinguish from a time-shifted version of itself.
2. Each signal is easy to distinguish from (a possibly time-shifted version of) every other signal in the set.

In practical applications of CDMA systems binary block codes of 0s and 1s are transmitted as +1s and -1s using the conversion

$$x_i = (-1)^{b_i} : b_i \in \{0,1\};\ x_i \in \{-1,1\}.$$

The properties that follow then apply to spreading codes of -1s and +1s. They are also applicable to ternary spreading codes with elements {-1, 0, +1}. The properties are easily generated for complex sequences by replacing every y_i in the correlation equations below by a complex conjugate of y_i. Each bit of a spreading codeword is usually referred to as a *chip*.

Period of a code

A codeword x of length N is said to be periodic of period t if for all i, $x_i = x_{(i+t) \bmod N}$. The period P of a code is then the lowest common multiple of the periods of all the codewords. Using the cyclic shift T, P is the least integer for which $T^P x = x$ for all codewords x.

We shall for convenience adopt the notation x_{i+k} for $x_{(i+k) \bmod N}$ throughout this work. For most spreading codes of interest for DS-CDMA systems, the period of a code is equal to the length of the code. That is $P = N$.

Periodic (even) autocorrelation

The periodic (even) autocorrelation function denoted as $\theta(x,x)(\tau)$ between a codeword x of length N and a time-shifted version of itself $T^\tau(x)$ as shown in figure 2.1) is the inner product:

$$\theta(x,x)(\tau) = \sum_{i=0}^{N-1} x_i x_{i+\tau} \qquad 0 \le \tau \le N-1. \tag{2.3}$$

The expression in equation (2.3) is called the *peak* autocorrelation when $\tau = 0$ and *off-peak* (or *out-of-phase*) autocorrelation when $\tau \neq 0$. For all the types of

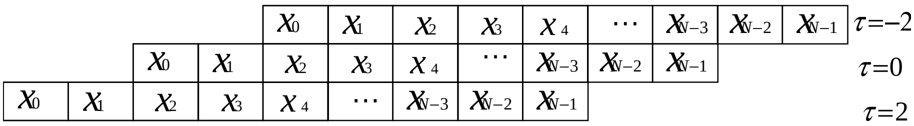


Figure 2.1: *A graphical definition of signal time delay in chips of a spreading codeword x. This definition holds for all time delays throughout this work.*

correlation (i.e. even, odd, and aperiodic correlations) described here, the peak correlation is the same but the off-peak autocorrelation may differ considerably.

Periodic (even) crosscorrelation

The even crosscorrelation function $\theta(x, y)(\tau)$ between two codewords x and y (or a time shifted version of y) of the same length N is the inner product:

$$\theta(x, y)(\tau) = \sum_{i=0}^{N-1} x_i y_{i+\tau} \qquad 0 \leq \tau \leq N - 1. \tag{2.4}$$

Odd autocorrelation

The need for odd correlation functions is explained in section 2.4. The odd autocorrelation function $\hat{\theta}(x, x)(\tau)$ between x and $x(\tau)$ is expressed as:

$$\hat{\theta}(x, x)(\tau) = \sum_{i=0}^{N-\tau-1} x_i x_{i+\tau} - \sum_{i=N-\tau}^{N-1} x_i x_{i+\tau} \qquad 0 \leq \tau \leq N - 1. \tag{2.5}$$

Odd crosscorrelation

The odd crosscorrelation function $\hat{\theta}(x, y)(\tau)$ between two codewords x and y is expressed as:

$$\hat{\theta}(x, y)(\tau) = \sum_{i=0}^{N-\tau-1} x_i y_{i+\tau} - \sum_{i=N-\tau}^{N-1} x_i y_{i+\tau} \qquad 0 \leq \tau \leq N - 1. \tag{2.6}$$

Aperiodic correlation

As noted in chapter 1, the aperiodic correlation is the basic correlation in CDMA work. The need for aperiodic correlation in measuring the level of interference at the output of a correlation receiver will appear in equations (2.31) and (2.32). The aperiodic crosscorrelation is defined as:

$$C(x,y)(\tau) = \begin{cases} \sum_{i=0}^{N-1-\tau} x_i y_{i+\tau} & 0 \leq \tau \leq N-1 \\ \sum_{i=0}^{N-1+\tau} x_{i-\tau} y_i & 1-N \leq \tau < 0 \\ 0 & \mid \tau \mid \geq N. \end{cases} \tag{2.7}$$

The aperiodic autocorrelation is obtained by changing every y into an x in equation (2.7). For $0 \leq \tau \leq N-1$, the aperiodic correlation function could also be defined as the mean of the even and odd correlation functions.

Welch and related correlation bounds

Welch, in [83], published a collection of lower bounds on peak correlations of spreading codes. The method used was a familiar one. Welch found the bounds based on Root Mean Square (RMS) of correlations and asserted that the peak correlations over all pairs of codewords could not be smaller than these bounds. The Welch bounds relating to spreading codes for asynchronous operation will initially be stated.

For a set $\mathcal{S}$ containing K codewords, define the peak even crosscorrelation θ_c as

$$\theta_c = \max\{\mid \theta(x,y)(\tau) \mid: 0 \leq \tau \leq N-1, x \in \mathcal{S}, y \in \mathcal{S}, x \neq y\}. \tag{2.8}$$

and the peak out of phase even autocorrelation as

$$\theta_a = \max\{\mid \theta(x,x)(\tau) \mid: 1 \leq \tau \leq N-1, x \in \mathcal{S}\}. \tag{2.9}$$

It is shown by Sarwate in [66] that for $K \leq N$

$$\left(\frac{\theta_c^2}{N} \right) + \frac{N-1}{N(K-1)} \left(\frac{\theta_a^2}{N} \right) \geq 1 \tag{2.10}$$

and defining $\theta_{\max} = \max\{\theta_a, \theta_c\}$,

$$\theta_{\max} \geq N \left[\frac{K-1}{NK-1} \right]^{1/2}, \tag{2.11}$$

a result first proved by Welch in [83]. For $K > N$, it is shown by Sidelnikov in [71] that

$$\theta_{\max} > (2N-2)^{1/2}. \tag{2.12}$$

Bounds on peak odd correlation are obtained by replacing every θ in equations (2.10), (2.11), (2.12) with $\hat{\theta}$. Defining the peak aperiodic crosscorrelation as

$$C_c = \max\{\mid C(x,y)(\tau) \mid : 0 \leq \tau \leq N-1, x \in \mathcal{S}, y \in \mathcal{S}, x \neq y\}. \tag{2.13}$$

and the peak out of phase aperiodic autocorrelation as

$$C_a = \max\{\mid C(x,x)(\tau) \mid : 1 \leq \tau \leq N-1, x \in \mathcal{S}\}. \tag{2.14}$$

It is shown by Sarwate in [66] that if x contains $K \leq N$ codewords then

$$\tfrac{(2N-1)}{N} \left(\tfrac{C_c^2}{N} \right) + \tfrac{2(N-1)}{N(K-1)} \left(\tfrac{C_a^2}{N} \right) \geq 1. \tag{2.15}$$

which implies that:

$$C_{\max} = \sqrt{\max\{C_c^2, C_a^2\}} \geq \left[\frac{N^2(K-1)}{2NK-K-1}\right]^{1/2}, \tag{2.16}$$

again first found by Welch in [83]. The bounds presented by equations (2.11) and (2.16) are called the Welch bounds.

Corresponding correlation bounds for generalised orthogonal codes

Generalised orthogonal codes are codes proposed for quasi-synchronous operation. Let the integer $\tau_{\max} < N$ be one greater than the maximum delay for a zero or low correlation zone. Zero or low crosscorrelations are maintained within a confined delay interval $0 \leq\mid \tau \mid< \tau_{\max}$. Similarly, zero or small autocorrelations are maintained in the delay interval $0 <\mid \tau \mid< \tau_{\max}$. For K codewords, corresponding bounds to the Welch lower bounds of equations (2.11) and (2.16) are obtained as:

1. For even correlation, it is shown by Tang et. al in [79, 80] that:

$$\theta_{\max} \geq \sqrt{\theta_{x,x}(0)^2 \frac{K\tau_{\max} - N}{(K\tau_{\max} - 1)N}}. \tag{2.17}$$

   Bounds for the odd correlation are obtained by changing $\theta_{\max}$ and $\theta_{x,x}(0)$ to $\hat{\theta}_{\max}$ and $\hat{\theta}_{x,x}(0)$ respectively. In the case of a zero correlation zone i.e. $\theta_{\max} = 0$ or $\hat{\theta}_{\max} = 0$, equation (2.17) shows that $K\tau_{\max} > N$ is impossible. Thus, in such cases we have:

$$K\tau_{\max} \leq N \tag{2.18}$$

2. For aperiodic correlation [79]:

$$C_{\max} \geq \sqrt{C_{x,x}(0)^2 \frac{(K-1)\tau_{\max} - N + 1}{(K\tau_{\max} - 1)(N + \tau_{\max} - 1)}}. \tag{2.19}$$

For a zero correlation zone, i.e $C_{\max} = 0$, it is clear by a similar argument as equation (2.18) that:

$$(K-1)\tau_{\max} \leq N - 1. \tag{2.20}$$

Comments on the Welch and other related bounds

The Welch bounds and other related bounds of equations (2.11),(2.16), (2.17) and (2.19) have been useful correlation criteria for spreading code design. Spreading codes for which the RMS of correlations over all pairs of codewords is equal to these lower bounds, especially the lower bound of equation (2.11), are referred to as Welch Bound Equality (WBE) codes. Spreading codes are considered to be uncorrelated when either the RMS or peak of correlations over all pairs of codewords satisfies the Welch bounds. This may be the reason why early spreading codes are designed to have uniform (or uniform occurrence of) correlations. Examples of WBE codes are the M-sequences, Gold codes, Small and Large sets of Kasami sequences.

Generally, a new spreading code with higher RMS or peak of correlations than for a Gold code or Small or Large set of Kasami sequences is not considered a significant advance in code design. However, there exist some codes (e.g. modified LS codes described in section 3.10) for which the number of codewords indicated by the variations of the Welch bounds in equations (2.17) and (2.19) is doubled but with correlations very dependent on the pair of codewords considered. Correlations between many pairs of codewords are zero but the RMS and peak of correlations between other pairs of codewords are higher than for the Gold code or Small or Large sets of Kasami sequences. The work presented here is able to show that if these new type of codes are carefully assigned to exploit the pairwise dependency, better system performance can be achieved than for codes in use today (which are examples of WBE codes). This presents a new approach to spreading code design.

2.3 Relationships between correlation properties

For a time-delay τ (measured in chips), $0 \leq \tau \leq N-1$, the following relations are true [68]:

$$\theta(x,y)(-\tau) = \theta(x,y)(N-\tau) = \theta(y,x)(\tau), \tag{2.21}$$

$$\mid \hat{\theta}(x,y)(-\tau) \mid \;=\; \mid \hat{\theta}(x,y)(N-\tau) \mid \;=\; \mid \hat{\theta}(y,x)(\tau) \mid, \tag{2.22}$$

$$C(x,y)(-\tau) = C(y,x)(\tau), \tag{2.23}$$

$$\theta(x,Ty)(\tau) = \theta(x,y)(\tau+1), \tag{2.24}$$

$$\hat{\theta}(x,Ty)(\tau) = \hat{\theta}(x,y)(\tau+1) + 2x_{N-\tau-1}y_0, \tag{2.25}$$

$$\theta(x,y)(0) = \hat{\theta}(x,y)(0) = C(x,y)(0), \tag{2.26}$$

$$C(x,Ty)(\tau) = \begin{cases} C(x,y)(\tau+1) + x_{N-\tau-1}y_0 & 0 \leq \tau \leq N-1 \\ C(x,y)(\tau+1) - x_{-\tau-1}y_0 & 1-N \leq \tau < 0 \end{cases}, \tag{2.27}$$

$$C(Tx,Tx)(\tau) = \begin{cases} C(x,x)(\tau) - x_0x_\tau + x_{N-\tau}x_0 & 0 \leq \tau \leq N-1 \\ C(x,x)(\tau) - x_{-\tau}x_0 + x_0x_{N+\tau} & 1-N \leq \tau < 0 \end{cases}, \tag{2.28}$$

$$\theta(x,y)(\tau) = C(x,y)(\tau) + C(x,y)(\tau-N), \tag{2.29}$$

$$\hat{\theta}(x,y)(\tau) = C(x,y)(\tau) - C(x,y)(\tau-N). \tag{2.30}$$

It follows from equations (2.24), (2.25), (2.27), (2.28) that the periodic correlation functions are independent of the phase of a code while the odd correlation functions and aperiodic correlations depend on the phase of the code.

2.4 Reasons for requiring low correlation

The two basic signal design problems presented at the beginning of section 2.2.1 were reduced in [68] to a problem of designing spreading codes that satisfy the following properties:

1. Let x be a codeword. The correlations $\theta(x,x)(\tau)$, $\hat{\theta}(x,x)(\tau)$ and $C(x,x)(\tau)$, for all $\mid \tau \mid > 0$, must take on small values.

2. For two codewords x and y, $\theta(x,y)(\tau)$, $\hat{\theta}(x,y)(\tau)$ and $C(x,y)(\tau)$, for all $\mid \tau \mid \geq 0$, must take on small values.

These two properties will be further illustrated by modifying equation (1.1) to include the periodic, odd and aperiodic correlations as follows.

Let d_n be the n^{th} bit of a wanted signal with spreading codeword x. Let

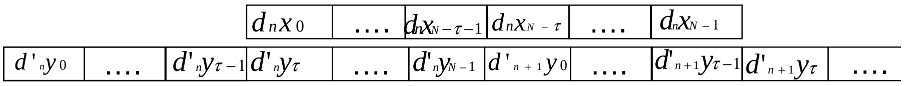


Figure 2.2: *This figure illustrates a correlation between spreading codeword x with wanted signal bit d_n and spreading codeword y with interfering message bits d'_n and d'_{n+1} at time delay τ.*

d'_n and d'_{n+1} be the n^{th} and $(n+1)^{\text{st}}$ bits of an unwanted signal with spreading codeword y. Let $\tau \geq 0$ be the number of left cyclic shifts of codewords y (in chips) relative to x arising as a result of synchronisation uncertainty. The output of a correlation receiver which is synchronised with spreading codeword x, as in equation (1.1), for the n^{th} wanted bit is represented by:

$$\hat{Z}_n = d_n \sum_{i=0}^{N-1} x_i x_i + \left[d'_n \sum_{i=0}^{N-\tau-1} x_i y_{i+\tau} + d'_{n+1} \sum_{i=N-\tau}^{N-1} x_i y_{i-N+\tau} \right]. \tag{2.31}$$

The first term arises from the desired signal and is d_n times the peak autocorrelation of its spreading codeword x. The term in square-brackets represents the interference due to the presence of another transmitted message signal with n^{th} bit d'_n and $(n+1)^{\text{st}}$ bit d_{n+1} in the channel. It is clear from equation (2.31) that the first $N-\tau$ elements of the wanted spreading codeword x are correlated against the last $N-\tau$ elements of the unwanted spreading codeword y, and the last τ elements of x are correlated against the first τ elements of y. This is graphically illustrated in figure 2.2.

Comparing equation (2.31) with the definitions of periodic autocorrelation and aperiodic crosscorrelation in equations (2.3) and (2.7) respectively, it follows that:

$$\hat{Z}_n = d_n \theta(x,x)(0) + \left[d'_n C(x,y)(\tau) + d'_{n+1} C(x,y)(\tau - N) \right]. \tag{2.32}$$

If there is no bit transition (say $d'_n = d'_{n+1} = +1$), equations (2.31) and (2.4) then imply:

$$\hat{Z}_n = d_n \theta(x,x)(0) + d'_n \theta(x,y)(\tau), \tag{2.33}$$

which also follows from equations (2.29) and (2.32).

On the other hand, if there is a bit transition (i.e. $d'_n = -d'_{n+1}$), then from equations (2.31) and (2.6) it follows that:

$$\hat{Z}_n = d_n \theta(x,x)(0) + d'_n \hat{\theta}(x,y)(\tau), \tag{2.34}$$

which also follows from equations (2.30) and (2.32).

It is easy to see from equations (2.29) and (2.30) that for $\tau \geq 0$:

$$C(x,y)(\tau) = \frac{1}{2}\{\theta(x,y)(\tau) + \hat{\theta}(x,y)(\tau)\}. \tag{2.35}$$

If d'_n is a sequence of independently, identically distributed, binary random variables, then $P(d'_n \neq d'_{n+1}) = \frac{1}{2}$ and so both the even and odd correlation functions are equally important in CDMA systems. It could be further deduced from equation (2.35) that the aperiodic correlation at each τ is the mean modulus of the even and the odd correlations. It is therefore referred to as the basic correlation.

The second term of each of equations (2.32), (2.33) and (2.34) represents the interference contribution due to multiple access at the output of the correlation receiver and it increases with the number of interferers (users). To mitigate this interference it is necessary for the expressions representing interference to take on zero (or tolerably low) values. The same reason applies to delayed versions of the wanted signal which occur due to the multi-path nature of the channel. Such signals are mitigated by zero (or low) off-peak autocorrelations of the spreading code.

2.5 Discussion

In this chapter, a brief review of code theories that are most relevant to properties of spreading codes has been presented. A straightforward representation of different correlation formulae for real valued spreading codes (taking on values between +1 and -1) has been shown. These representations are easily generalised for complex valued codes [68]. The review also highlighted some known bounds on the spreading code correlations. In the case of codes for asynchronous operation, large values of code size K and code length N, implies that the lower bounds of equations (2.11) and (2.16) approximate to $\theta_{\max} = \sqrt{N}$ and $C_{\max} = \sqrt{N/2}$ respectively [64]. In the case of codes for quasi-synchronous operation, equations (2.17) and (2.19) imply that if $\tau_{\max}$ takes on achievable values the number of codewords satisfying the low or zero correlation property is small. In particular, from equations (2.18) and (2.20) the number of codewords with a zero correlation zone is $K \leq N/\tau_{\max}$ or $K-1 \leq (N-1)/\tau_{\max}$. This means that in a relatively large network (which requires a big $\tau_{\max}$), codeword re-use distances will be forced to be too small for satisfactory operation. Methods for increasing the number of codewords by trading off the low or zero correlations between some pairs of codewords are described in [35]. This project is then basically concerned with minimising any resulting interference.

The reason why low correlations are required for a spreading code from an interference perspective was discussed. Note that the model presented for the output of a correlation receiver is assumed for a perfect channel that is without noise and multi-path. A correlation receiver treats interfering signals due to multipath as additional background noise. The model can easily be generated for a rake receiver by including the outputs (or terms) of the rake fingers.

All the expressions presented in this chapter are only valid for time uncertainty $\tau \geq 0$. Modification for $\tau < 0$ for even correlation can easily be generated using equations (2.21) and (2.29). Aperiodic correlation is well defined for $\tau < 0$ from equations (2.7) and (2.23). The sign of the odd correlation depends on a chosen convention. $\hat{\theta}(x, y)(-\tau)$ could be equal to either $\hat{\theta}(y, x)(\tau)$ or $-\hat{\theta}(y, x)(\tau)$. For this reason, odd correlation is not generally defined for $\tau < 0$ but a useful relation is $\hat{\theta}(x, y)(-\tau) = \hat{\theta}(x, y)(N - \tau)$. This difficulty does not arise if we use the aperiodic correlation in the interference measure for our cost function model.

Chapter 3

Examples of Spreading Codes

In this chapter, examples of spreading codes that are available in the literature will be described and their respective assignment implications discussed. The terms *codeword* and *sequence* shall be used interchangeably.

A code C is *linear* if the sum (mod 2) of any two codewords in C is also a codeword. The *dimension* of the code is the number of codewords in a basis. A code of length N is an $[N, k, d]$ code if it has dimension k and minimum distance d [31].

A *cyclic code* is a linear code in which any cyclic shift of a codeword is also a codeword. Cyclic codes can be represented using *binary polynomials*. A codeword $c = (c_0, c_1, \ldots, c_{N-1})$ is written as a polynomial $c(x) = (c_0 + c_1 x + ... + c_{N-1} x^{N-1})$. The *generator polynomial* $g(x)$ of a cyclic code C is the unique non-zero polynomial of minimum degree in the set of code polynomials $C(x)$. If $g(x)$ has degree $N-k$ then the words $g(x), xg(x), \ldots, x^{k-1}g(x)$ form a basis for the code. Every word $v(x) = u(x)g(x)$, where $u(x)$ is a word in $C(x)$ of degree less than k. A generator polynomial could be obtained by factorising $x^N - 1 = g(x)h(x)$ [76], where $h(x)$ is the *parity check polynomial* and for every word $v(x)$ in $C(x)$, $v(x)h(x) \equiv 0 \bmod (x^N - 1)$.

An *irreducible polynomial* $h(x)$ over K ($K = \{0, 1\}$) of degree r, $r > 1$ is said to be *primitive* if it does not divide $1 + x^m$ for any $m < 2^r - 1$. An element α of the finite Galois Field GF(2^r) is primitive if every non-zero element in GF(2^r) could be expressed as a power of α. An irreducible polynomial $h(x)$ with degree r in GF(2^r) is also primitive if it has a primitive element α as a root.

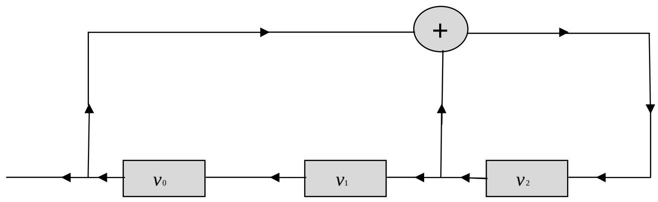


Figure 3.1: *An example of a linear shift-register using a primitive polynomial* $h(x) = 1 + x^2 + x^3$*. Note that the operator "+" denotes addition modulo 2 .*

3.1 Maximal length sequence

Consider a parity-check polynomial $h(x)$ and any word $v(x)$ in the binary code $C(x)$ of length N. From the definition of a parity-check polynomial in section 2.1, we have $v(x)h(x) \equiv 0 \bmod (x^N - 1)$. This implies that for all integers j:

$$\sum_{i=0}^{k} v_{i+j} h_i = 0 \tag{3.1}$$

or

$$v_{k+j} = \sum_{i=0}^{k-1} v_{i+j} h_i \tag{3.2}$$

If $h_k = h_0 = 1$ and $h(x)$ is a primitive polynomial of degree k, then the solution of the recurrence relation in (3.2) is the periodic sequence $v_0, v_1, \ldots, v_N, v_0, v_1, \ldots,$ of length and period $N = 2^k - 1$. From knowledge of $v_0, v_1, \ldots, v_{k-1}$, the value of v_k can be found and so forth. The periodic sequence is called a maximal-length sequence or simply an m-sequence.

A linear sequential switching device with k cells which calculates the sum indicated in equation (3.2) and hence calculates v_i from the previous k values is called a *Linear Feedback Shift-Register* (LFSR). An example of a linear shift register is given in figure 3.1. A primitive polynomial $h(x) = 1 + x^2 + x^3$ of degree $k = 3$ is used. For example, when the initial state of the LFSR of figure 3.1 is set as $\{v_0, v_1, v_2\} = \{1, 0, 1\}$, the m-sequence generated is $\{1010011\}$ with length $N = 2^3 - 1 = 7$. In the practical applications of interest here, binary sequences of 0's and 1's are transmitted as +1's and −1's by replacing every 0 with a +1 and every 1 with a −1.

3.1.1 Properties of binary shift-register codes

The following properties of m-sequences are taken from [68]

1. The period of a shift-register code V is $N = 2^k - 1$ where k is the degree of the primitive polynomial $h(x)$.

2. There are N codewords generated by $h(x)$ and these can be written as $v, Tv, T^2v, \ldots, T^{N-1}v$.

3. $T^iv \oplus T^jv = T^kv$, for $0 \le i,\ j < N$ and some k, $0 \le k < N$, where $\oplus$ denotes addition modulo 2.

4. The weight $wt(v)$ of any sequence v in C is the number of 1's in a codeword containing 0's and 1's and is given by

$$wt(v) = 2^{k-1} = \frac{N+1}{2} ,$$

which is the number of 1's in a codeword containing 0's and 1's.

5. There is one more 1 than 0.

6.

$$\theta(v,v)(l) = \begin{cases} N & \text{if } l \equiv 0 \mod N \\ -1 & \text{if } l \not\equiv 0 \mod N \end{cases}$$

7. For some initial setting of the shift register, $v_i = v_{2i}$ for all $i > 0$. The unique sequence satisfying this property is called the characteristic sequence or the characteristic phase of the m-sequence v.

8. For $q = 1, 2, \ldots, N-1$ the decimation of a non-zero m-sequence v, denoted as $v[q]$, is obtained by taking every q^{th} element of v mod N. It has a period $N/gcd(N,q)$ and generator polynomial (say $\hat{h}(x)$) with roots that are q^{th} powers of the roots of $h(x)$.

9. If v is generated by primitive polynomial $h(x)$ of degree k and $u = v[q]$ is the decimation of v by $q = 2^j + 1$ or $q = 2^{2j} - 2^j + 1$ such that:

$$e = gcd(k, j)$$

and $\frac{k}{e}$ is odd, then the periodic crosscorrelations $\theta_{x,y}(\tau)$: $0 \le \tau \le N-1$, for such pairs are three valued $(-1, t(k)-2, -t(k))$ with:

$$\begin{cases} -1 & \text{occurs} & 2^k - 2^{k-e} - 1 & \text{times} \\ t(k)-2 & \text{occurs} & 2^{k-e-1} + 2^{(k-e-2)/2} & \text{times} \\ t(k) & \text{occurs} & 2^{k-e-1} - 2^{(k-e-2)/2} & \text{times,} \end{cases} \tag{3.3}$$

where

$$t(k) = \begin{cases} 2^{\frac{k+1}{2}} + 1 & \text{for } k \text{ odd} \\ 2^{\frac{k+2}{2}} + 1 & \text{for } k = 2 \mod 4 \end{cases} .$$

Such a pair of m-sequences (which was first formulated by Gold in [25]) is called a *preferred pair* of m-sequences. A list of polynomials of the preferred pairs can be found in [14].

10. From equation (2.11) θ_{max} for m-sequences satisfies :

$$\theta_{max} \geq N^{\frac{1}{2}}.$$

M-sequences are of great interest in CDMA cellular networks. In particular, suppose that all base stations use a common time reference using a Global Positioning System (GPS) [3, 13, 21, 87]. Each base station can use a unique phase offset of the m-sequence. This combination can be used in both the forward link and reverse link of IS-95. Assignment of the m-sequence can be done in an arbitrary way.

3.2 Gold sequences

Consider two preferred pairs of m-sequences (say m_1 and m_2) generated by two primitive polynomials $h(x)$ and $\hat{h}(x)$ such that a shift-register polynomial $f(x)$ factors into $h(x)\hat{h}(x)$ of degree k. The Gold code generated by $f(x)$ [25, 88, 89] is:

$$G(m_1, m_2) = \{m_1, m_2, m_1 \oplus m_2, m_1 \oplus T^1 m_2, m_1 \oplus T^2 m_2, \ldots, m_1 \oplus T^{N-1} m_2\} \quad (3.4)$$

A Gold code contains $N+2 = 2^k+1$ codewords of period N. The even correlation function is three valued [30] (i.e. $-1, t(k)-2, -t(k)$ where $t(k)$ is as defined in property 9 of shift-register codes) and the peak even correlation parameters θ_c and θ_a satisfy $\theta_c = \theta_a = t(k)$. That is:

$$\theta_{max} = t(k) \quad (3.5)$$

When k is odd, the Gold code forms an optimal set due to the Sidelnikov lower bound in equation (2.12). The distribution of the correlation spectrum $\{-1, t(k)-2, t(k)\}$ is approximately $\{50\%, 30\%, 20\%\}$ respectively over time delay interval $0 \leq \tau \leq N-1$. However, when k is even (i.e. $k \equiv 2 \bmod 4$) the lower bound is weak - the Gold code is not an optimal set in this case but the distribution of the correlation spectrum $\{-1, t(k)-2, t(k)\}$ is approximately $\{75\%, 15\%, 10\%\}$ respectively over the time delay interval $0 \leq \tau \leq N-1$. Note that, from property 9 of M-sequences, if k is an even number the correlation spectrum achievable by a Gold code of length $2^k - 1$ has the same values as the correlation achievable by a Gold code of length $2^{k+1} - 1$. The peak correlations $t(k)$, $t(k+1)$ are equal and also correlations $t(k)-2$, $t(k+1)-2$ are equal. However, the frequency of occurrence of the best correlation value -1 reduces from 75% to 50% and the frequencies of occurrence of $t(k)$ and $t(k)-2$ increase from 10% and 15% respectively to 20% and 30% respectively. It then follows that there may be little advantage in using a Gold code generated by a primitive polynomial of an odd degree k (requiring more frequency spectrum) over a Gold code generated by a primitive polynomial of an even degree $k-1$ (requiring less frequency spectrum). This of course is subject to the number of users that can be supported.

An example of two shift-registers generating a Gold code is shown in figure 3.2. Let $h(x) = 1 + x^2 + x^5$, be the primitive polynomial defining a linear feedback shift register f_1 which generates m-sequence m_1 and $\hat{h}(x) = 1 + x^2 + x^3 + x^4 + x^5$, be the primitive polynomial defining a linear feedback shift register f_2 which generates m-sequence m_2. When the initial state of f_2 is set as $\{v_0, v_1, v_2, v_3, v_4\} = \{0, 0, 0, 0, 0\}$ the output of the combined LFSR is m_1 and equally, the output is m_2 when the state of f_1 is set to zero. In particular, $\{m_1 = 1111100011011101010000100101100\}$ is obtained when the initial state of f_1 is set to 1 all through and f_2 is set to 0. In the same way, $\{m_2 = 1111100100110000101101010001110\}$ is obtained when the initial state of f_2 is set to 1 all through and f_1 is set to 0. The Gold code generated from this example has length $N = 2^5 - 1 = 31$. A total of 33 codewords can be constructed by changing the initial states of the LFSR's or by different shift combinations of m_1 and m_2 as in equation (3.4). The absolute correlation spectrum is $\{1, 7, 9\}$ occurring $15, 10, 6$ times respectively over all time delays $0 \leq \tau < 31$.

A Gold code is actively used in the forward link of a Wideband Cellular CDMA (WCDMA) network as a scrambling code [13]. Each base station is assigned a Gold sequence, which is common to all users within the same cell but different from those assigned to neighbouring cells. Unlike the use of m-sequences in IS-95, there is no need for any external timing source such as a GPS clock to maintain common timing between cells - the system, in this case, is inter-cell asynchronous [3, 13]. If system planning is based on the even correlation alone, assignment of a Gold code can be done arbitrarily in small networks. A careful re-use of codewords, similar to a graph colouring approach with co-channel constraint, is necessary in a network where re-use of codewords must be tolerated. The variation in the odd and aperiodic correlation of a Gold code is small. For this reason, when Gold codes are used algorithms for careful spreading code assignment may be of relatively little importance.

3.3 Small set of Kasami sequences

Let k be even and let m_1 denote an m-sequence of period $N = 2^k - 1$ generated by $h(x)$ of degree k. Consider the sequence $m_3 = m_1[s(k)] = m_1[2^{\frac{k}{2}}+1]$ obtained by decimating m_1 by $s(k) = 2^{\frac{k}{2}} + 1$. It follows from property 8 of m-sequences that m_1 has period $2^{\frac{k}{2}} - 1$. It is generated by a primitive polynomial (say $\hat{h}(x)$)

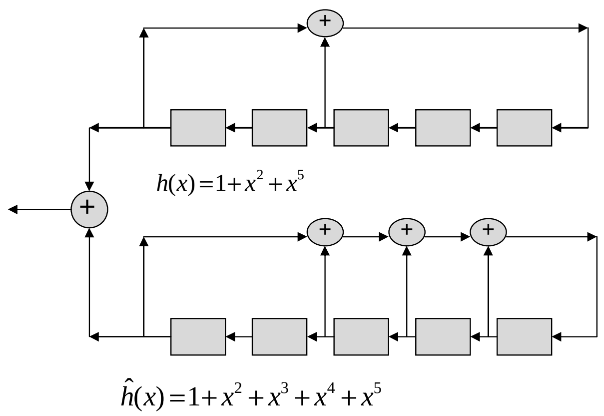


Figure 3.2: *An illustration of the generation of a Gold code set*

of degree $\frac{k}{2}$ and has roots that are $s(k)^{th}$ powers of the roots of $h(x)$. The Small set of Kasami sequences generated by the polynomial $f(x) = h(x)\hat{h}(x)$ of degree $\frac{3k}{2}$ is defined by:

$$K_s(m_1) = \{m_1, m_1 \oplus m_3, m_1 \oplus T^1 m_3, \ldots, m_1 \oplus T^{(2^{\frac{k}{2}}-2)} m_3\} \tag{3.6}$$

Kasami discovered that the even correlation functions for sequences belonging to $K_s(m_1)$ take on values in the set $\{-1, -s(k), s(k) - 2, \}$ [38, 68]. The total number of sequences in the set $K_s(m_1)$ is $2^{\frac{k}{2}}$ and $\theta_{max} = s(k)$. $K_s(m_1)$ then form an optimal set due to the Welch lower bound in equation (2.10). Assignment of a Small set of Kasami sequences is similar to that of a Gold code. The peak even correlation is the same for all pairs of codewords and little variation is experienced in the odd and aperiodic correlations from pair to pair of codewords. Note that $s(k)$ is less than $t(k)$ the peak of a Gold code but the least correlation value -1 which occurs at 75% when k is even or 50% when k is odd in a Gold code has the least occurrence in a Small set of Kasami sequences.

3.4 Gold-like sequences

Let k be even, $k \equiv 0 \bmod 4$, and let q be an integer such that the $\gcd(q, 2^k - 1) = 3$. Suppose m_1 is an m-sequence of period $N = 2^k - 1$ generated by a primitive polynomial $h(x)$, and let $m_4^{(l)}$, $l = 0, 1, 2$, denote the result of decimating $T^l m_1$ by q. From property 8 of m-sequences, $m_4^{(l)}$ are sequences of period $N' = N/3$ generated by a primitive polynomial $h'(x)$ whose roots are the q^{th} powers of the roots of $h(x)$. The set of sequences generated by $f(x) = h(x)h'(x)$ is:

$$
\begin{aligned}
H_q(m_4) = \{&m_1, m_1 \oplus m_4^{(0)}, m_1 \oplus Tm_4^{(0)}, \ldots, m_1 \oplus T^{N'-1}m_4^{(0)}, \\
&m_1 \oplus m_4^{(1)}, m_1 \oplus Tm_4^{(1)}, \ldots, m_1 \oplus T^{N'-1}m_4^{(1)}, \\
&m_1 \oplus m_4^{(2)}, m_1 \oplus Tm_4^{(2)}, \ldots, m_1 \oplus T^{N'-1}m_4^{(2)}\}.
\end{aligned} \tag{3.7}
$$

These sequence are called Gold-like sequences. There are a total of $N + 1 = 2^k$ sequences. The even correlation spectrum of sequences belonging to $H_q(u)$ is $-1, -t(k), t(k) - 2, -s(k), s(k) - 2$ where $t(n)$ and $s(n)$ are as defined in section 3.1.1 (9) and section 3.3 respectively. The maximum absolute correlation is $\theta_{\max} = t(k)$ [68].

3.5 Large set of Kasami sequences

Consider a set generated by $f(x) = h(x)h'(x)\hat{h}(x)$ where $h(x)$ is the primitive polynomial that generates an m-sequence m_1, $h'(x)$ is the primitive polynomial that generates the m-sequence m_2 (as in the case of a Gold sequence) or m_4 (as in the case of a Gold-like sequence) and $\hat{h}(x)$ is the primitive polynomial that generates the m-sequence m_3. The *Large set of Kasami sequences* K_L is constructed as follows [68]:

- if $k \equiv 2 \bmod 4$ then:

$$
K_L = G(m_1, m_2) \bigcup \left[\bigcup_{i=0}^{2^{k/2}-2} \{T^i m_3 \oplus G(m_1, m_2)\} \right] \tag{3.8}
$$

where $m_2 = m_1[t(k)]$, $m_3 = m_1[s(k)]$ and $G(m_1, m_2)$ is as defined in equation (3.4). K_L in this case contains $2^{k/2}(2^k + 1)$ sequences.

- if $k \equiv 0 \bmod 4$, then:

$$
\begin{aligned}
K_L = H_{t(k)}(m_4) \bigcup \left[\bigcup_{i=0}^{2^{k/2}-2} \{T^i m_3 \oplus H_{t(k)}(m_4)\} \right] \\
\bigcup \{m_4^{(j)} \oplus T^l m_3 : 0 \le j \le 2, 0 \le l < (2^{k/2} - 1)/3\}
\end{aligned} \tag{3.9}
$$

where $m_4^{(j)}$ is the result of decimating $T^j m_2$ by $t(k)$ and $H_{t(k)}(m_4)$ is as defined in (3.7). K_L in this case contains $2^{k/2}(2^k + 1) - 1$ sequences.

In either case the correlation spectrum for K_L takes on values in the set $\{-1, -t(k), t(k) - 2, -s(k), s(k) - 2\}$ and the maximum absolute value

$\theta_{\max} = t(k)$. The Large set of Kasami sequences contains the Small set of Kasami sequences and the set of Gold sequences or the set of Gold-like sequences.

The Large set of Kasami sequences is one of the families of spreading codes with a large number of sequences. In WCDMA, a Large set of Kasami sequences is used as a spreading code in the uplink direction [13]. Each mobile user is assigned a unique sequence different from those assigned to close mobile users. Since the Large set of Kasami sequences is generated from the Gold code and the Small set of Kasami sequences, its assignment implication is similar to that of a Gold code and a Small set of Kasami sequences. The peak periodic correlation is the same for all pairs of codewords, but the aperiodic and odd correlations experience little variations from pair to pair of codewords. When codeword re-use is not a critical problem, little advantage is achievable by careful code assignment.

3.6 Codes derived from Kerdock codes

Kerdock codes are non-linear binary codes derived from linear codes over the set $\mathbf{Z}_4 = \{0, 1, 2, 3\}$ (i.e. with arithmetic mod 4)[46]. Specifically, for an odd positive integer $m > 1$ the Kerdock code $K(m)$ has length $n = 2^{m+1}$, minimum distance $(n - \sqrt{n})/2$ and n^2 codewords. As noted in [28] and [51] certain codes derived from Kerdock codes have the same even correlation properties as the Small Kasami set but many more codewords. The construction described in [35] will be further described here.

Given an odd positive integer $m > 1$, a monic primitive irreducible polynomial $f_m x^m + \ldots + f_1 x + f_0$, $(f_m = 1)$ over the integers mod 4 (referred to as a *characteristic polynomial*) is used. A method of constructing such polynomials is given in [29]. Let $\mathbf{h}$ be a vector with coordinates h_i, $i = 1, \ldots, m+1$ defined by

$$h_i = (-1)^{i-1} f_{m-i+1} (\text{mod } 4).$$

Let $\mathbf{g} = (g_0, g_1, g_2, \ldots, g_{m-1})$ be a vector with entries from $\mathbf{Z}_4$ (a *seed vector*), and define an infinite sequence $\{p_i\}$ $(p_i \in \mathbf{Z}_4,\ i = 0, 1, 2, \ldots)$ by:

$$p_i = g_i \ (i = 0, 1, 2, \ldots, m-1) \text{ and } p_i = -\sum_{k=2}^{m+1} h_k p_{i-k+1} \ (i \geq m), \qquad (3.10)$$

with all arithmetic mod 4. This $\mathbf{Z}_4$ sequence $\{p_i\}$ is a linear recurrent sequence of period $2^{m+1} - 2$ for almost all initial choices of seed vector $\mathbf{g}$. The $\mathbf{Z}_4$ linear recurrence can be represented as a shift register using mod 4 multiplication and mod 4 addition. Consider the first $2^{m+1} - 2$ elements $\{p_i\}$

$(p_i \in \mathbf{Z_4}, \mathbf{i} = \mathbf{0}, \mathbf{1}, \mathbf{2}, \ldots, \mathbf{2^{m+1}} - \mathbf{3})$ of each sequence. Each element of p_i is converted to binary and the most significant bit is chosen. The binary sequence $\mathbf{b} = \{b_i\}(b_i \in \mathbf{Z}_2, i = 0, 1, 2, \ldots, 2^{m+1} - 3)$ is obtained. Now consider four binary vectors of length $2^{m+1} - 2$:

$$\begin{aligned}
\mathbf{a}_1 &= (00000000000000000\ldots000) \\
\mathbf{a}_2 &= (11111111111111111\ldots111) \\
\mathbf{a}_3 &= (01010101010101010\ldots101) \\
\mathbf{a}_4 &= (10101010101010101\ldots010).
\end{aligned}$$

Then the 4^{m+1} binary vectors $\mathbf{b} + \mathbf{a_1}$, $\mathbf{b} + \mathbf{a_2}, \mathbf{b} + \mathbf{a_3}, \mathbf{b} + \mathbf{a_4}$ obtained for all distinct initial states $\mathbf{g}$ of the shift register are the codewords of the cyclic binary code. Note that not all of these codewords are useful. The sequences obtained when $\mathbf{g}$ has all components 0 or 2 have period $2^m - 1$ and so have a large autocorrelation. Furthermore, the complement of each codeword is also present. These codewords must be removed from the set of sequences. An algorithm that can be used to achieve this is described in [35]. The Kerdock derived code obtained has length $2^{m+1} - 2$ with 2^m cycles of codewords of full period. Note that $\mathbf{a_1}$ and $\mathbf{a_3}$ are the complements of $\mathbf{a}_2$ and $\mathbf{a}_4$ respectively. Exactly one of $\mathbf{a}_1$ and $\mathbf{a}_2$ and exactly one of $\mathbf{a}_3$ and $\mathbf{a}_4$ is then used to generate a complement-free cyclic code.

For a Kerdock derived code of length $N = 2^{m+1} - 2$, the number of codewords available is 2^m (which is half the number of codewords available in a Gold code) and the maximum even correlation is $2^{(m+1)/2} - 2$. The peak odd (and consequently the peak aperiodic) correlation differ from pair to pair of Kerdock codewords. The method of code assignment for Kerdock derived codes is the same as that described for Gold codes. The variations in correlations are small and relatively little advantage may be gained if codeword re-use is not a critical issue.

3.7 Walsh-Hadamard codes

Walsh-Hadamard codes are examples of orthogonal codes. They can be generated by using a special square matrix called a *Hadamard matrix* [13, 42]. Each row of the Hadamard matrix represent a Walsh-Hadamard codeword. The Hadamard matrix of desired length can be created using the recursive procedure:

$$H_1 = [+1], \quad H_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix} \quad H_4 = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}$$
$$H_{2k} = \begin{bmatrix} H_k & H_k \\ H_k & \overline{H_k} \end{bmatrix} \tag{3.11}$$

where $\overline{H_k}$ represents the binary complement of matrix H_k and k is a power of 2. There are 2^k codewords of length 2^k. Other constructions are possible. Crosscorrelations between codewords of a Walsh-Hadamard code are 0 at $\tau = 0$ but unsatisfactorily high for other τ's.

Walsh-Hadamard codes are used as spreading codes in cellular communication systems e.g. IS-95, third generation mobile telephone systems and other systems where synchronisation to $\tau = 0$ can be maintained. Walsh-Hadamard codes do not require careful assignment since the crosscorrelations are the same when synchronisation is maintained.

3.8 Lin-Chang-Simplex (LCS) codes

Let m, n be two integers with m a divisor of n and let $\tau_{\max} = \frac{2^n-1}{2^m-1}$. Lin and Chang [45] created a family of $\frac{\binom{2^m-1}{2^{(m-1)}}}{(2^m-1)}$ cyclically distinct sequences of length $N = 2^n - 1$ with $\theta_{x,y}(\tau) = -1$ for $\tau \neq 0 \pmod{\tau_{\max}}$. However, for some pairs x, y of codewords $\theta_{x,y}(0) > 1$ when $m > 3$, and so the family of codes as presented by Lin and Chang does not contain a low correlation zone including $\tau = 0$ with $\theta_{\max} = 1$. A modification using seed vectors from *simplex codes* is proposed in [35] where the modified codes are named *Lin-Chang-Simplex* codes. A code with $2^m - 1$ codewords is constructed by performing shift operations on some balanced seed vectors [35, 46]. Each seed vector then generates an LCS codeword. Seed vectors of LCS codes are expected to have the following properties:

1. Each vector has one more 1 than 0.
2. Any two vectors are cyclically distinct.
3. Each vector has length $2^m - 1$.
4. The periodic correlation for any pair (say e, f) is -1 except when $e = f$ and delay $\tau = 0$.

Let α be a primitive element of the finite field $GF(2^n)$ and $Tr_m^n(x) = \sum_{j=0}^{n/m-1} x^{2mj}$ be the trace function from $GF(2^n)$ to $GF(2^m)$ [46]. Each element s_k of a *shift sequence* $S = (s_0, s_1, \ldots, s_{2^n-2})$ is defined by the trace function as:

$$s_k = \begin{cases} i & if\ Tr_m^n(\alpha^k) = \alpha^{Ti}\ i \in \{0, 1, \ldots, 2^m - 2\} \\ \\ \infty & if\ Tr_m^n(\alpha^k) = 0 \end{cases} \tag{3.12}$$

Let $e = (e_0, e_1, \ldots, e_{2^m-2})$ be a binary $\{0, 1\}$ seed vector. Shift operations are performed on e to generate a $(2^m - 1) \times \tau_{\max}$ array E with columns labelled $0, 1, \ldots, \tau_{\max} - 1$ as follows. If s_i is ∞ then column i of E is a column of zeros. If $s_i \neq \infty$ then column i of E is the transpose of $(e_{s_i}, e_{s_i+1}, \ldots, e_{(s_i+2^m-2) \bmod (2^m-1)})$ and thus is a cyclic shift of e. The sequence X_e (of length $2^n - 1$) generated by seed vector e can be obtained by scanning the rows of the array E, starting from the top left hand corner. The sequence can be written as:

$$x_e = (e_{s_0}, e_{s_1}, \ldots, e_{s_{\tau_{\max}-1}}, e_{s_{\tau_{\max}}}, \ldots, e_{s_{2(\tau_{\max}-1)}}, e_{s_{2\tau_{\max}}}, \ldots, e_{s_{2^n-2}}) \tag{3.13}$$

Let x_e and y_f be two LCS sequences generated by balanced seed vectors e and f. The periodic crosscorrelation function $\theta_{x_e,y_f}(\tau)$ of x_e and y_f is expressed as:

$$\theta_{x_e,y_f}(\tau) = \begin{cases} -1 & \text{for } \tau \neq 0 \pmod{\tau_{\max}} \\ 2^{n-m} - 1 + 2^{n-m}\theta_{ef}(d) & \text{for } \tau = d\tau_{\max} \end{cases} \tag{3.14}$$

where $\theta_{ef}(\tau)$ is the crosscorrelation of e and f. The seed vectors must be chosen from a *cyclically permutable* simplex code, i.e. a simplex code with the zero codeword omitted and with all vectors cyclically distinct. Thus the vectors are of full period and cyclically distinct. Such a code can be obtained (for $m > 3$) by a simple hill climbing algorithm [35]. Columns of the generator matrix are transposed if the number of cyclically distinct codewords increases.

An LCS code has suitable properties for quasi-synchronous operation. However the number of codewords available based on the lower bound of equation (2.17) is small for realistic networks. For example, there are only 31 codewords in the LCS code of length 1023. A method of increasing the number of codewords by using two or more distinct sets of LCS was introduced in [35]. The sets use the same seed vector but different shift sequences. Low correlations are maintained between codewords within the same set but the correlations are worse between codewords from different sets. A method of mitigating the resulting interference is addressed in this work. A pair of close transmitters should be assigned a pair of codewords from the same set; a pair of transmitters that are farther apart can be assigned a pair of codewords from different sets. A codeword can be re-used where a re-use threshold is satisfied.

3.9 Loosely Synchronous codes

A Loosely Synchronous (LS) code is an example of a spreading code suitable for quasi-synchronous operation. The method of LS code construction described by Stańczak et. al in [78] will be presented here. Let $\tau_{\max}$ be one chip greater than the maximum synchronisation uncertainty in a network. LS codes are constructed using Hadamard Matrices and the sequence pairs called Golay pairs, inserting $\tau_{\max} - 1$ zeros in between the Golay sequences. This results in a ternary $\{0, -1, +1\}$ type of spreading code. As a result of the concatenation $\theta_{x,y}(\tau) = C_{x,y}(\tau) = 0 : 0 \leq| \tau |\leq \tau_{\max} - 1$ and $\theta_{x,x}(\tau) = C_{x,x}(\tau) = 0 : 0 <| \tau |\leq \tau_{\max} - 1$. It is easy to achieve this property if sufficient numbers of zeros are used. However for efficient implementation spreading codes must have a *duty ratio* (defined as the number of non-zero elements in a period divided by N) close to 1. For simplicity, LS codes are described using generating functions. The sequence $+1, +1, +1, -1, +1, +1, -1, +1$ can be represented as a generating function $G(z) = 1+z+z^2-z^3+z^4+z^5-z^6+z^7$.

Given two binary sequences $C(z)$ and $S(z)$ of length $\mathcal{N}$, $C(z)$ and $S(z)$ form a Golay pair if:

$$C(z)C(z^{-1}) + S(z)S(z^{-1}) = 2\mathcal{N} \tag{3.15}$$

(in the Laurent ring $\mathbf{Z}[z, z^{-1}]$, which has as elements all power series of the form $\sum_{i=-\infty}^{\infty} a_i z^i$). The two sequences separated by $\tau_{\max} - 1$ zeros form a ternary sequence with aperiodic zero autocorrelation if $0 < |\tau| \leq \tau_{\max} - 1$ and peak aperiodic autocorrelation $2\mathcal{N}$. Thus for example, consider two sequences of length 8 with generating functions $C(z) = 1 + z + z^2 - z^3 + z^4 + z^5 - z^6 + z^7$ and $S(z) = 1+z+z^2-z^3-z^4-z^5+z^6-z^7$. Clearly $C(z)C(z^{-1})+S(z)S(z^{-1}) = 16$ and so $C(z)$ and $S(z)$ is a Golay pair. Then the sequence $+1+1+1-1+1+1-1+1 \quad 0 \quad 0 \quad 0 \quad 0+1+1+1-1-1-1+1-1$ with generating function $C(z) + z^{\mathcal{N}+\tau_{\max}-1}S(z) = C(z) + z^{12}S(z)$ is an LS codeword. The off-peak autocorrelation is zero within the interval $0 < \tau < \tau_{\max} = 5$. Note that it is necessary to add 4 further zeros at the end to generate a periodic codeword. Golay pairs exist of lengths $\mathcal{N} = 2^r \times 10^s \times 26^t$ for all non-negative r, s, t. Constructions of sequences whose length is a multiple of 10 and 26 can be found in [16]. The two sets of zeros will be referred to as *external padding* A Golay pair of length 2 is given by $A(z) = 1 + z$, $B(z) = 1 - z$. Given a Golay pair $A(z), B(z)$ of length $\mathcal{N}$; a Golay pair $C(z), D(z)$ of length $2\mathcal{N}$ can be constructed as:

$$C(z) = A(z) + z^{\mathcal{N}}B(z), \quad D(z) = A(z) - z^{\mathcal{N}}B(z). \tag{3.16}$$

Applying t iterations of this rule yields another Golay pair of length $2^t\mathcal{N}$.

Consider two Golay pairs $C_0(z), S_0(z)$ and $C_1(z), S_1(z)$ such that:

$$\begin{aligned} C_0(z)C_0(z^{-1}) + S_0(z)S_0(z^{-1}) &= 2\mathcal{N}, \\ C_1(z)C_1(z^{-1}) + S_1(z)S_1(z^{-1}) &= 2\mathcal{N}, \end{aligned} \tag{3.17}$$

and

$$C_0(z)C_1(z^{-1}) + S_0(z)S_1(z^{-1}) = 0. \tag{3.18}$$

The two Golay pairs $C_0(z), S_0(z)$ and $C_1(z), S_1(z)$ for which equations (3.17) and (3.18) hold are referred to as *cross complementary sequence pairs*, and the second pair is said to be a *mate* of the first. The mate $C_1(z), S_1(z)$ of a Golay pair $C_0(z), S_0(z)$ can be constructed as:

$$C_1(z) = z^{(\mathcal{N}-1)}S_0(z^{-1}), \quad S_1(z) = -z^{(\mathcal{N}-1)}C_0(z^{-1}) \tag{3.19}$$

For example let:

$$\begin{aligned} C_0(z) &= 1 + z + z^2 - z^3 + z^4 + z^5 - z^6 + z^7 \\ S_0(z) &= 1 + z + z^2 - z^3 - z^4 - z^5 + z^6 - z^7 \end{aligned}$$

The mate $C_1(z), S_1(z)$ can be constructed as:

$$\begin{aligned} C_1(z) = z^7 S_0(z^{-1}) &= -1 + z - z^2 - z^3 - z^4 + z^5 + z^6 + z^7 \\ S_1(z) = -z^7 C_0(z^{-1}) &= -1 + z - z^2 - z^3 + z^4 - z^5 - z^6 - z^7 \end{aligned}$$

It is easy to see that the Golay pair $C_0(z), S_0(z)$ with mate $C_1(z), S_1(z)$ form a cross complementary sequence pair.

Now, given cross complementary Golay pairs $C_0(z), S_0(z)$ and $C_1(z), S_1(z)$. Then the two sequences:

1. the sequence represented by $C_0(z)$ followed by $W_0 \geq \tau_{\max} - 1$ zeros followed by the sequence represented by $S_0(z)$;

2. the sequence represented by $C_1(z)$ followed by $W_0 \geq \tau_{\max} - 1$ zeros followed by the sequence represented by $S_1(z)$;

form two sequences with the desired properties. Let $\pi \in \{0, 1\}$ be a binary element with complement $\pi^* = (\pi + 1)_{\text{ mod } 2}$. The two LS sequences can be represented as

$$\begin{aligned} C_\pi(z) + z^{\mathcal{N}+\tau_{\max}-1}S_\pi(z) \\ C_{\pi^*}(z) + z^{\mathcal{N}+\tau_{\max}-1}S_{\pi^*}(z). \end{aligned}$$

The periodic or aperiodic correlations (i.e. both the autocorrelation and cross-correlation) are zero for the given value of $\tau_{\max} - 1$. The W_0 zeros ensure that a sequence represented by a $C(z)$ does not overlap with a sequence represented by a $S(z)$ (in quasi-synchronous operation). To obtain a periodic sequence it is also necessary to add a further W_0 zeros of external padding after the sequence represented by $S_\pi(z)$ or $S_{\pi^*}(z)$. This ensures that this sequence does not overlap with the first sequence in the next period.

Now to obtain K LS sequences of length $L = K\mathcal{N} + W_0$, a $[p \times p]$ Hadamard matrix is used such that $p = K/2$ [10] . Hadamard matrices can only exist if $p = 1, 2$ or $4k$ for some integer k (see section 3.7) and certainly do exist for values $p \leq 428$ [10]. Let $\pi = \{\pi_1, \pi_2, \ldots, \pi_p\}, \pi_k \in \{0, 1\}$ be a binary vector of length p with binary complement $\pi^* = \{\pi_1^*, \pi_2^*, \ldots, \pi_p^*\} : \pi_k^* = (\pi_k + 1) \bmod 2$. Then the $2p$ sequences of an LS code of length $2p\mathcal{N} + W_0$ can be described by [78]:

$$\begin{aligned} G_k(z) &= \textstyle\sum_{i=1}^{p} h_{k,i} \left[C_{\pi_i}(z) + z^{p\mathcal{N}+W_0} S_{\pi_i}(z)\right] z^{(i-1)\mathcal{N}} \\ G_{k+p}(z) &= \textstyle\sum_{i=1}^{p} h_{k,i} \left[C_{\pi_i^*}(z) + z^{p\mathcal{N}+W_0} S_{\pi_i^*}(z)\right] z^{(i-1)\mathcal{N}}. \end{aligned} \tag{3.20}$$

The off-peak aperiodic autocorrelation and aperiodic crosscorrelation is zero for $|\tau| \leq \min(\mathcal{N} - 1, W_0)$. Thus with the choice $W_0 = \mathcal{N} - 1$, $2p$ sequences of length $2p\mathcal{N} + 2(\mathcal{N} - 1)$ have been constructed with zero correlation window $\tau_{\max} - 1 = W_0$. Note that the correlation at $\tau = \mathcal{N}$ is $\mathcal{N}(p - 1)$.

For example, consider a Golay pair $C_0(z) = 1 + z + z^2 - z^3 + z^4 + z^5 - z^6 + z^7$, $S_0(z) = 1+z+z^2-z^3-z^4-z^5+z^6-z^7$ with mate $C_1(z) = z^7 S_0(z^{-1}) = -1+z-z^2-z^3-z^4+z^5+z^6+z^7$, $S_1(z) = -z^7 C_0(z^{-1}) = -1+z-z^2-z^3+z^4-z^5-z^6-z^7$ of length $\mathcal{N} = 8$. Choosing $W_0 = \mathcal{N} - 1 = 7$ and $\pi = \{01\}$ so that its complement $\pi^* = \{10\}$. Let the Hadamard matrix be a 2×2 matrix so that 4 codewords of length $L = 2.2.8 + 14 = 46$ and 0.7 duty ratio are constructed. The sequences constructed are:

$$\begin{array}{llllll} G_1(z) = & C_0(z) & C_1(z) & 0000000 & S_0(z) & S_1(z) \\ G_2(z) = & C_0(z) & -C_1(z) & 0000000 & S_0(z) & -S_1(z) \\ G_3(z) = & C_1(z) & C_0(z) & 0000000 & S_1(z) & S_0(z) \\ G_4(z) = & C_1(z) & -C_0(z) & 0000000 & S_1(z) & -S_0(z) \end{array}$$

The zero correlation window has a $\tau_{\max} = 8$ between any pair of codewords. A bigger Hadamard matrix can be used to generate more codewords. For example the use of a 16×16 Hadamard matrix will generate 32 codewords but the length of the LS code increases to 270 with duty ratio 0.95. It then suffices that as implied by the bound in equation (2.20) the length of an LS code increases with the increase in the number of codewords desired. A method to double the number of codewords with minimal increase in the length of the LS code is described in the next section.

3.10 LS codes with internal padding

The approach of using internal padding in LS codes was introduced by Jones et. al in [35]. The main aim of this improvement is to double the number of LS codewords achievable by the approach in section 3.9 and indicated by the bound in equation (2.20) without significant increase in the implied length of the LS code. The LS code zero correlation zone is modified so that the property is only partially maintained.

Departing from the construction of section 3.9, suppose the length of the Golay pair $\mathcal{N}$ is halved and p doubled. The new Golay pair has length $\mathcal{N}' = \mathcal{N}/2$ and 4p LS sequences of the same length N can then be constructed. Good correlation properties are then required for $|\tau| \leq \min(2\mathcal{N}' - 1, W_0)$. But using the construction of equation (3.20) high values of the aperiodic correlation may be obtained for $|\tau| = \mathcal{N}'$.

Now define a set of *internal padding lengths* $\{L_1, L_2, L_3, \ldots, L_{\mathcal{P}-1}\}$ such that $\mathcal{P} = 2p$. Then compute the cumulative padding lengths $Q_1 = 0$, $Q_i = \sum_{j=1}^{i-1} L_j$ $(2 \leq i \leq \mathcal{P})$. Let $H = [h_{i,j}]$ be a $\mathcal{P} \times \mathcal{P}$ Hadamard matrix, and use a cross-complementary sequence pair given by $C_0(z), S_0(z)$ and $C_1(z), S_1(z)$. Let $\pi = [\pi_1, \pi_2, \ldots, \pi_{\mathcal{P}}]$ be a binary vector of length $\mathcal{P}$ and let π^* be the complement of π. Then the $2\mathcal{P}$ sequences of an *LS code with internal padding* for $1 \leq k \leq \mathcal{P}$ can be described by:

$$\begin{aligned} G_k(z) &= \sum_{i=1}^{\mathcal{P}} h_{k,i} \left[C_{\pi_i}(z) + z^{\mathcal{P}\mathcal{N}+W_0+Q_{\mathcal{P}}} S_{\pi_i}(z)\right] z^{(i-1)\mathcal{N}+Q_i} \\ G_{k+\mathcal{P}}(z) &= \sum_{i=1}^{\mathcal{P}} h_{k,i} \left[C_{\pi_i^*}(z) + z^{\mathcal{P}\mathcal{N}+W_0+Q_{\mathcal{P}}} S_{\pi_i^*}(z)\right] z^{(i-1)\mathcal{N}+Q_i}. \end{aligned} \tag{3.21}$$

Note that if all $L_j = 0$ the construction reduces to that of equation (3.20). Now, in order to avoid a small duty ratio, distinct internal padding lengths are repeated a small number of times. This involves considering transitions and non-transitions in the elements of vectors π and π^*. A *transition* occurs when:

1. one Golay pair C member is followed by a different Golay pair C member, i.e. C_0 followed by C_1 or C_1 followed by C_0;

2. one Golay pair S member is followed by a different Golay pair S member, i.e. S_0 followed by S_1 or S_1 followed by S_0.

A *non-transition* occurs when:

1. one Golay pair C member is followed by the same Golay pair C member, i.e. C_0 followed by C_0 or C_1 followed by C_1;

2. one Golay pair S member is followed by the same Golay pair S member, i.e. S_0 followed by S_0 or S_1 followed by S_1.

For each transition, the corresponding increase in the aperiodic correlation is zero (as a result of equation (3.18)) for two codewords generated by π or two codewords generated by π^*. For each non-transition, the corresponding increase in the aperiodic correlation is zero (as a result of equation (3.18)) for one codeword generated by π and one codeword generated by π^*. Note that the total number of transitions and non-transitions is equal to $\mathcal{P} - 1$. Let (Tr,NTr) denote the number of transitions and the number of non-transitions. Three cases (Tr,NTr) can be considered.

Case 1: $(Tr,NTr) = (\frac{\mathcal{P}}{2}, \frac{\mathcal{P}}{2} - 1)$.

In this case vector π is chosen with $\frac{\mathcal{P}}{2}$ transitions and $\frac{\mathcal{P}}{2} - 1$ non-transitions. Transitions are assigned padding lengths

$$\{L_1, L_1, L_2, L_2, L_3, L_3, \ldots, L_{\mathcal{P}-2}, L_{\mathcal{P}-2}, L_{\mathcal{P}-1}, L_{\mathcal{P}-1}\}$$

(in some order) and non-transitions are assigned padding lengths

$$\{L_1, L_1, L_2, L_2, L_3, L_3, \ldots, L_{\mathcal{P}-2}, L_{\mathcal{P}-2}, L_{\mathcal{P}-1}\}.$$

Some Golay pairs could give a non-zero contribution to the aperiodic correlation (referred to here as a *coincidence*) for each value of τ with $|\tau| < 2\mathcal{N}$. If we let $W_0 = 2\mathcal{N}' - 1$ then the aperiodic crosscorrelation and off-peak autocorrelation is 0 for $0 \leq |\tau| \leq \mathcal{N}' - 1$ and $\frac{3\mathcal{N}'}{2} \leq \tau \leq 2\mathcal{N}' - 1$. The aperiodic correlations have non-zero values in the interval $\mathcal{N}' \leq \tau \leq \frac{3\mathcal{N}'}{2} - 1$ as a result of the coincidences. Note that even when the aperiodic correlation is non-zero, it is still smaller than the maximum for a Gold code with a similar number of non-zero entries.

Case 2: $(Tr,NTr) = (\mathcal{P} - 1, 0)$.

In this case all internal padding lengths are of necessity assigned to transitions. If $W_0 = 2\mathcal{N}' - 1$ the aperiodic correlation for any two codewords from the same set is 0 for $0 \leq |\tau| \leq \mathcal{N}' - 1$ and $\frac{3\mathcal{N}'}{2} \leq \tau \leq 2\mathcal{N}' - 1$. It gives some non-zero values in the delay $\mathcal{N}' \leq \tau \leq \frac{3\mathcal{N}'}{2} - 1$. This non-zero correlation is still small for a pair of codewords both generated by π or π^*. The non-zero correlation is worse and in fact more than the peak correlation of a Gold code of approximately the same length.

Case 3: $(Tr,NTr) = (\frac{3\mathcal{P}}{4} - 1, \frac{\mathcal{P}}{4})$.

In this case three instances of each internal padding length are assigned to transitions and one to non-transitions. If $W_0 = 2\mathcal{N}' - 1$ the aperiodic correlation for any two codewords from the same set is 0 for $0 \leq |\tau| \leq \mathcal{N}' - 1$ and $\frac{3\mathcal{N}'}{2} \leq \tau \leq 2\mathcal{N}' - 1$. It gives some non-zero values that are far less than $p\mathcal{N}'$ in the delay $\mathcal{N}' \leq \tau \leq \frac{3\mathcal{N}'}{2} - 1$. $\tau_{\max}$ is equal to $2\mathcal{N}'$. In the same way as in case 2, this non-zero correlation is still small for a pair of codewords both generated by π or π^*. The non-zero correlation is worse and in fact more than the peak correlation of a Gold code of relatively the same length.

The method of code assignment presented in this work will be used to mitigate the resulting interference to achieve better system performance than when a Gold code or a related asynchronous code is used. A list of examples of LS codes with internal padding can be found in [65].

For example, consider a network with the following requirements:

1. $\tau_{\max}$ must be greater than or equal to 32
2. Required code length (or available frequency spectrum) ≤ 1400.
3. Minimum number of codewords = 60

The construction without internal padding will be initially analysed. To achieve the required $\tau_{\max} = 32$ a Golay code of length $\mathcal{N} = 32$ (so that $W_0 = 31$) can be constructed. To generate a minimum of 60 codewords a 32×32 Hadamard matrix can be constructed but this will result in an LS code of length $2.32.32 + 62 = 2110$ which is greater that 1400 required. This will require more frequency spectrum.

Now, using the approach of including internal paddings, a Golay code of length $\mathcal{N}' = \mathcal{N}/2 = 16$ (so that $W_0 = 2\mathcal{N}' - 1 = 31$) can be constructed, with a 32×32 Hadamard matrix. If the maximum repetition for the internal padding lengths is 4 such that the total internal padding length is 105. The length of the LS code constructed is then 1296 which is less than 1400. For the case (Tr,NTr) = (16,15), the peak correlation which occurs within the interval ($16 \leq |\tau| \leq 23$) is 64. For case (Tr,NTr) = (31, 0), the peak correlation is 96 (only between two codewords for which one is generated by π and the other is generated by π^* in the interval ($16 \leq |\tau| \leq 23$)). In the case of (Tr,NTr) = (24,7), the peak correlation is 32 for $16 \leq |\tau| \leq 23$ for two codewords both generated by π or both generated by π^*. For two codewords, one generated by π, one generated by π^* the peak aperiodic crosscorrelation is at most 96 for $16 \leq |\tau| \leq 23$.

The LS code (with internal padding) using any of the cases presented clearly presents a spreading code assignment problem that is more challenging than a straightforward graph colouring problem. To take advantage of the variation in the correlations of the LS code (with internal padding) a pair of close transmitters need to be assigned a pair of codewords such that either both are generated by π or both generated by π^*. In this case the peak correlation is smaller than that of a Gold code. A pair of transmitters that are farther apart can be assigned a pair of LS codewords such that one is generated using π and the other using π^*.

3.11 Discussion

In this chapter some examples of spreading codes available in the literature have been reviewed including their assignment implications. This chapter forms the basis for this project. It is clear that the assignment implications vary for different spreading codes. It is in fact critical in some cases, for example, the LS code (with internal padding) and the LCS code. These codes are examples of codes suitable for quasi-synchronous operation. Codeword re-use is a critical problem, as is the variation experienced in the correlations. The method of code assignment presented in this work will be used to mitigate the resulting interference to achieve better system performance than when a Gold code or a related asynchronous code is used.

It is clear that codes for quasi-synchronous operation present challenging tasks both in code design and assignment. An area of research interest is the problem of how to further increase the number of such codewords without significant increase in the implied length of the code.

Chapter 4

Performance Measures for CDMA Systems

Two important measures of spreading code performance are available in the literature. The first is the maximum absolute value of the correlations and the second is the mean square value of the correlations. These two measures are important but are of little meaning to a system designer whose interest is more on how these measures directly relate to standard system measures such as the signal-to-interference ratio (SIR), the signal-to-noise ratio (SNR) and the bit-error-rate (BER). Consequently the aim of this chapter is to give a simple account of these system measures in a spread spectrum CDMA application. The way by which spreading code assignment can be used to mitigate the effect of interference using each system measure is discussed.

4.1 Signal-to-Noise-Ratio

Signal-to-Noise Ratio (SNR) is an important measure in radio systems. It is expressed as the ratio of the wanted signal power to the noise power at the output of the correlator. A communication system is limited by both the interference due to multiple access and the channel noise. If the noise is white Gaussian noise with infinite power, SNR will be approximately equal to zero. Thus, the spreading (and consequently spreading code assignment) offers no advantage over such noise [60]. If the noise is white Gaussian noise with finite power N_0 (one-sided spectral density of noise), the SNR [11] at the output of a correlation receiver is:

$$SNR = \frac{2E_b}{N_0} \tag{4.1}$$

where E_b is the energy per bit of the wanted signal. Note that this ratio is only valid when the spreading codeword of the wanted signal is known by the receiver. This assumption holds for all system performance measures discussed here.

The multiple access interference may be viewed as being random in nature and so could be treated as additional noise [63]. Let $|T|$ be the total number of transmitters. The average variance of the combination of $|T| - 1$ multiple access interference and the channel noise, in particular *Average White Gaussian Noise* (AWGN), is used to compute average SNR at the output of a correlation receiver. That is, at reception point r served by transmitter i [39, 63]:

$$SNR_r^{(\mathrm{AV})} = \left[\frac{N_0}{2E_b} + \frac{1}{6N^3} \sum_{j=1, j \neq i}^{|T|} \sum_{\tau=1-N}^{N-1} \{2.C_{x_i,y_j}^2(\tau) + C_{x_i,y_j}(\tau)C_{x_i,y_j}(\tau+1)\} \right]^{-1}, \tag{4.2}$$

where $\frac{E_b}{N_0}$ is the bit energy to noise ratio and C_{x_i,y_j} is the aperiodic correlation between spreading codeword x_i assigned to transmitter i and spreading codeword y_j assigned to transmitter j. Note that equation (4.2) does not include the case where codeword re-use must be tolerated or the channel is a multipath channel. Taking $\frac{N_0}{2E_b}$ as a constant, SNR_r then depends on the second term of equation (4.2) which is a function of the aperiodic correlation. For some families of codes e.g. Gold code, M-sequence, Small and Large sets of Kasami sequences, etc. the degree of variation in the aperiodic correlation is very small. Little advantage might be gained in adopting careful spreading code assignment if the aim is to obtain better SNR. If the degree of the variation is high, and including distance related path loss in the SNR model, better system performance (measured in SNR) can be obtained by careful spreading code assignment.

4.2 Interference and Signal-to-Interference-Ratio

Consider two types of interference - intentional interference and interference due to multiple access. Intentional interference (jamming) aims to disrupt the communication system. Intentional interference could be narrowband or wideband interference. Narrowband interference with high power only affects a small part of the spread spectrum signal leaving other parts undisturbed. At the receiver, the narrowband interference (with high power) is spread to form a wideband interference by the use of the spreading codeword generated by the receiver. The receiver at the same time demodulates the spread spectrum wanted signal (with low power before the correlator) to give the original narrowband signal (before spreading) with high power. This is illustrated in figure 4.1. In this case, SIR is just a power advantage of the despread message signal (with a narrow bandwidth) over the now spread interference (with a wide bandwidth); that is the ratio of the narrow bandwidth of the message signal to the wide bandwidth of the interference [60]. This is equivalent to the ratio of the spreading code length to a message bit, called the *processing gain*. This technique also affords a spread spectrum signal a protection against interfering signals from narrowband communication systems such as a GSM. The spread spectrum signal (with very low power) then only contributes very little interference (which can be regarded as background noise) at the receiver of the narrowband signal. Thus, the power advantage allows a spread spectrum communication system to co-exist with narrowband communication systems. In the same way, a spread spectrum message signal has a power advantage over a wideband interference (which is generated and spread in some way different from using a member of the spreading code of the message signal). Assignment of a spreading code does not have any effect in improving system performance against these types of interference as only the length of the code gives the power advantage.

Better understanding of how to mitigate interference due to multiple access can be realised if it is not treated as a random process. Multiple access interference depends on whether the spreading codewords used are orthogonal or not. If the spreading code is an orthogonal code, the SIR relates to the processing gain. If the signals are just uncorrelated (but with small correlations as in the case of a Gold code), SIR does not directly relate to the processing gain. In this case, when an interfering signal is multiplied by the wanted spreading codeword (generated by the receiver) there are no recognisable symbols obtained. SIR is then a power ratio of the wanted signal to the unwanted spread spectrum signals which depends on the correlations of the spreading code. This is further explored in section 5.2 of the next chapter. Again, if small variations are experienced in the correlations of the spreading codewords little advantage may be gained from careful spreading code assignment. If the correlations vary from pair to pair of codewords and the path loss is distance related in the SIR model, interference can be mitigated by careful code assignment.

4.3 Bit-Error-Rate

Bit-Error-Rate (BER) refers to the probability of incorrectly demodulating a message bit in the presence of interference or noise. The most cited and widely used approximation for BER in a spread spectrum system is the approximation given by Pursley in [63]. This approximation is based on the Standard Gaussian Approximation (with zero mean and unit variance). For K users, BER at a reception point r is approximated as $1 - \Phi(\sqrt{SNR_r})$, where Φ is the standard Gaussian cumulative distribution function and SNR_r is as represented in equation (4.2). In general, BER will not be exactly equal to $1 - \Phi(\sqrt{SNR_r})$ but it has been found to be a good approximation for large values of K. Quantitative results on this approximation are given in [86]. $\Phi(SNR_r)$ increases as SNR_r increases and so BER by direct implication decreases.

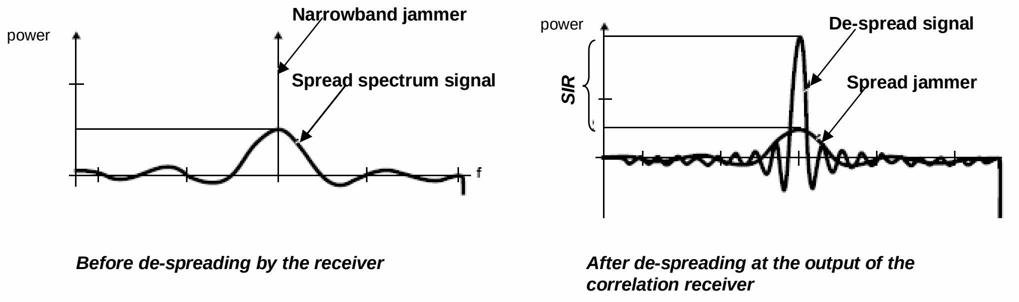


Figure 4.1: *This figure illustrates how spread spectrum signals combat narrowband interference, e.g., a jammer. It is easy to see that the SIR is the power advantage that a spread spectrum signal has over the narrowband interference*

BER does not have a direct relationship with the correlations of spreading codes. Additional complexity of cost function calculation will be introduced if BER is used as a system performance measure in a spreading code assignment problem. Sufficient knowledge of the noise environment is also required to compute the SNR. For these reasons it is not advisable to use a BER derived cost function for solving spreading code assignment problems.

4.4 The aim of spreading code assignment

The aim of spreading code assignment is to mitigate the effect of interference, which results from the correlation of spreading codes, in order to obtain better system performance. Considering the three system performance measures that have been described, SNR, SIR and BER, the SIR has the most direct relationship with multiple access interference in a CDMA system especially when considerable variations are experienced between correlations of codewords.

4.5 Discussion

In this chapter a simple description of system performance measures has been given. These system performance measures are the signal-to-noise ratio (SNR), the signal-to-interference ratio (SIR) and the bit-error-rate (BER). SIR and SNR have a close relationship with the correlations of spreading codes. However, the derivation of SNR is based on the assumption that the correlations follow a random process and so interference is considered as additional background noise. This may not be true with some spreading codes, e.g., LS codes, for which the correlations are deterministic. Additionally, an accurate value for $\frac{2E_b}{N_0}$ must be formulated in order to accurately measure the actual effect of the correlations. The work presented here improves on the approximation of SNR presented in [63] (also decribed in equation (4.2)) by developing a model (based on SIR) which takes into consideration the average correlation for each pair of codewords. If the interference term varies from pair to pair of codewords, a distance-based path loss may be used in order to obtain a better SNR approximation. This will then necessitate careful code assignment as one of the code selection criteria.

BER is an important performance measure but it is not a good choice for the cost function if the aim is to minimise interference due to correlations of spreading codes. The look-up table for the standard Gaussian cumulative distribution function would have to be made available to the optimisation algorithm. The optimisation process will then take longer than necessary. However, it is still a good measure for simple code comparison.

The major aim of the spreading code assignment problems addressed in this work is to mitigate interference that directly results from correlations of the codes. For this reason the model for the cost function adopted in this work will be based on SIR. SIR shall be properly exploited (to include the distance-based path loss) in section 5.2.

Chapter 5

Methodology for Spreading Code Assignment

In this chapter a description of the models adopted for spreading code assignment is presented. Methods available in the literature assume arbitrary correlation values. This assumption is based on the use of a common correlation value that is low enough to ensure that pairs of codewords are uncorrelated. In some cases, correlations between certain codewords may be low enough for the codewords to be near orthogonal while others are high but still regarded as uncorrelated. The advantages of the method proposed here are most apparent when the correlations of a spreading code vary from pair to pair of spreading codewords. The aim of spreading code assignment is therefore to:

1. assign pairs of codewords with low correlation values to pairs of transmitters that are close together,
2. assign pairs of codewords with high correlation values only to pairs of transmitters that are farther apart,
3. re-use a codeword only where a codeword re-use threshold can be satisfied.

The model presented incorporates relevant measures of spreading codes into the interference model. Performance measures of spreading codes are therefore described initially. The correlation values of spreading codes have a close relationship with *signal-to-interference ratio* SIR. It is therefore used as the system measure in the cost function for optimisation. Some remarks on codeword re-use and the evaluation of an assignment are presented. Mathematical techniques for predicting the minimum number of codewords and average correlation required for a problem are also described.

5.1 Performance measures for spreading codes

In a CDMA communication system, resistance to interference depends on the correlation observed between spreading codewords. System performance then depends on low spreading code correlation. Performance measures for asynchronous operation will first be analysed. Performance measures for quasi-synchronous operation will be considered subsequently.

5.1.1 Spreading code measure using maximum absolute values of correlations

The traditional spreading code performance measure for asynchronous operation is the maximum modulus of the correlations. This approach is open to the usual criticism that too much emphasis is placed on worst-case parameter values. However, the approach is warranted for certain systems and so will be further described. Define $\theta_{x,y}(\tau)$, $\hat{\theta}_{x,y}(\tau)$, and $C_{x,y}(\tau)$ $(\tau > 0)$ as the even, odd, and aperiodic crosscorrelations between spreading codewords x and y as in equations (2.4), (2.6) and (2.7). Definitions for $\tau < 0$ can be obtained from the relations $\theta_{x,y}(-\tau) = \theta_{y,x}(\tau)$, $C_{x,y}(-\tau) = C_{y,x}(\tau)$ and in particular for this work, the relation $\hat{\theta}_{x,y}(-\tau) = -\hat{\theta}_{y,x}(\tau)$ is adopted for the odd correlation. For a spreading code of length N, the measures:

$$\lambda_{x,y} = \max\{\gamma_{x,y}, \hat{\gamma}_{x,y}\} \tag{5.1}$$

where

$$\gamma_{x,y} = \max_{\tau} \mid \theta_{x,y}(\tau) \mid \,, \;\; \hat{\gamma}_{x,y} = \max_{\tau} \mid \hat{\theta}_{x,y}(\tau) \mid : 1 - N \leq \tau \leq N - 1$$

and

$$\varsigma_{x,y} = \max_{\tau} \mid C_{x,y}(\tau) \mid : 1 - N \leq \tau \leq N - 1 \tag{5.2}$$

are mostly used for worst-case performance of a pair of spreading codewords [63]. Measurements using the maximum of the off-peak autocorrelation (of say codeword x) are achieved by replacing every y with an x and excluding $\tau = 0$. Note that a consequence of equation (2.35) is:

$$\lambda_{x,y} \leq 2\varsigma_{x,y} \tag{5.3}$$

The worst-case measure using $\lambda_{x,y}$ for a spreading code $\mathcal{S}$ with K codewords is:

$$\lambda = \max_{x,y}\{\lambda_{x,y}\} \leq 2\max_{x,y}\{\varsigma_{x,y}\} \;\; (x, y \in \mathcal{S}). \tag{5.4}$$

Equally, the worst-case measure using $\varsigma_{x,y}$ is:

$$\varsigma = \max_{x,y}\{\varsigma_{x,y}\} \ (x, y \in \mathcal{S}) \tag{5.5}$$

From equations (5.4) and (5.5) [64]:

$$\varsigma \leq \lambda \leq 2\varsigma \tag{5.6}$$

The measures of equations (5.4) and (5.5) are useful in system planning for the worst interference that could be received from any interfering signal.

It is necessary to note that for most families of codes, $\lambda_{x,y}$ and $\varsigma_{x,y}$ differ for different pairs of codewords. A less stringent measure which takes account of these variations would be to find the average of the peak values. That is:

$$\lambda_{average} = \frac{1}{K^2}\sum_{i=0}^{K-1}\sum_{j=0}^{K-1}\lambda_{x_i,y_j} \tag{5.7}$$

and

$$\varsigma_{average} = \frac{1}{K^2}\sum_{i=0}^{K-1}\sum_{j=0}^{K-1}\varsigma_{x_i,y_j} \tag{5.8}$$

It can be inferred from equations (2.21), (2.22) and (2.23) that $\lambda_{x,y} = \lambda_{y,x}$ and $\varsigma_{x,y} = \varsigma_{y,x}$. Thus equations (5.7) and (5.8) are equivalent to:

$$\lambda_{average} = \frac{2}{K(K+1)}\sum_{i=0}^{K-1}\sum_{j=i}^{K-1}\lambda_{x_i,y_j} \tag{5.9}$$

and

$$\varsigma_{average} = \frac{2}{K(K+1)}\sum_{i=0}^{K-1}\sum_{j=i}^{K-1}\varsigma_{x_i,y_j} \tag{5.10}$$

The work presented here will improve on the worst-case measures of equations (5.4) and (5.5) (and described in [63]) by considering the pairwise dependency of peak correlations in the spreading code assignment methods presented.

5.1.2 Spreading code measure using mean-square of correlations

In recent times, the mean-square measure of the correlations has been found more appropriate [40, 67, 55]. Taking a Gold code and a Small set of Kasami sequences of length 1023 as an example, the sum of the squares of $\hat{\theta}_{x,y}(\tau)$ for all $\tau > 0$ is less for Gold codes than it is for Kasami codes [41]. In addition the frequency of occurrence of each correlation in the correlation spectrum of a code requires consideration. Compare a Small Kasami set with a subcode of a Gold code of the same length. With a Gold code of length 1023, a periodic correlation value of -1 occurs 761162 times which is about a 75 percent contribution. The same correlation value occurs in 30690 times (a 3 percent contribution) in the Kasami code, even though a Kasami code is believed to be better than a Gold code based on the maximum modulus of crosscorrelation. Since all correlation values, not just maximum values, affect resistance to interference, the mean-square method seems more appropriate for calculation of SIR.

For two spreading codewords x, y, the mean-square measures (normalised between 0 and 1) of the periodic, odd and aperiodic correlations are:

$$MSCC_{x,y} = \begin{cases} \frac{1}{N^2(2N-1)} \sum_{\tau=1-N}^{N-1} \mid \theta_{x,y}(\tau) \mid^2 \\ \frac{1}{N^2(2N-1)} \sum_{\tau=1-N}^{N-1} \mid \hat{\theta}_{x,y}(\tau) \mid^2 \\ \frac{1}{N^2(2N-1)} \sum_{\tau=1-N}^{N-1} \mid C_{x,y}(\tau) \mid^2 \end{cases} \tag{5.11}$$

$$MSAC_{x,x} = \begin{cases} \frac{1}{N^2(2N-2)} \sum_{\tau=1-N,\tau\neq 0}^{N-1} \mid \theta_{x,x}(\tau) \mid^2 \\ \frac{1}{N^2(2N-2)} \sum_{\tau=1-N,\tau\neq 0}^{N-1} \mid \hat{\theta}_{x,x}(\tau) \mid^2 \\ \frac{1}{N^2(2N-2)} \sum_{\tau=1-N,\tau\neq 0}^{N-1} \mid C_{x,x}(\tau) \mid^2 \end{cases} \tag{5.12}$$

where $MSCC_{x,y}$ is the mean-square of crosscorrelation and $MSAC_{x,x}$ is the mean-square of autocorrelation.

For code selection purposes, the mean-square values over all K codewords (again normalised between 0 and 1) may be expressed [53, 54, 55] in terms of the aperiodic correlation as:

$$R_{CC} = \frac{1}{K(K-1)(2N-1)N^2} \sum_{i=0}^{K-1} \sum_{j=0, j \neq i}^{K-1} \sum_{\tau=1-N}^{N-1} \mid C_{x_i,x_j}(\tau) \mid^2 \tag{5.13}$$

$$R_{AC} = \frac{1}{K(2N-2)N^2} \sum_{i=0}^{K-1} \sum_{\tau=1-N, \tau \neq 0}^{N-1} \mid C_{x_i,x_i}(\tau) \mid^2 \tag{5.14}$$

5.1.3 Corresponding spreading code measures for Quasi-synchronous CDMA (QS-CDMA) operation

Corresponding measures for Quasi-synchronous CDMA (QS-CDMA) can be obtained by adjusting the measures defined for asynchronous CDMA operation. In QS-CDMA, small correlation values are expected within a constrained time delay interval, $\{-\tau_{\max}+1, \ldots, \tau_{\max}-1\}$, where $\tau_{\max}$ is the maximum time delay (measured in chips) of the channel. Corresponding peak measures (in QS-CDMA operation) to $\lambda_{x,y}(\tau)$ and $\varsigma_{x,y}(\tau)$ between two codewords x and y in equations (5.1) and (5.2) respectively are defined below:

$$\lambda_{x,y} = \max\{\gamma_{x,y}, \hat{\gamma}_{x,y}\}, \tag{5.15}$$

such that:

$$\gamma_{x,y} = \max_{\tau} \mid \theta_{x,y}(\tau) \mid, \;\; \hat{\gamma}_{x,y} = \max_{\tau} \mid \hat{\theta}_{x,y}(\tau) \mid : (1-\tau_{\max} \leq \tau \leq \tau_{\max}-1), \tag{5.16}$$

and

$$\varsigma_{x,y} = \max_{\tau} \mid C_{x,y}(\tau) \mid : (1-\tau_{\max} \leq \tau \leq \tau_{\max}-1). \tag{5.17}$$

Similarly, corresponding normalised mean-square measures to $MSCC_{x,y}$ and $MSC\hat{Z}_{x,y}$ between two codewords x and y in equations (5.11) and (5.12) respectively are given as:

$$MSCC_{x,y} = \begin{cases} \frac{1}{N^2(2\tau_{\max}-1)} \sum_{\tau=1-\tau_{\max}}^{\tau_{\max}-1} \mid \theta_{x,y}(\tau) \mid^2 \\ \frac{1}{N^2(2\tau_{\max}-1)} \sum_{\tau=1-\tau_{\max}}^{\tau_{\max}-1} \mid \hat{\theta}_{x,y}(\tau) \mid^2 \\ \frac{1}{N^2(2\tau_{\max}-1)} \sum_{\tau=1-\tau_{\max}}^{\tau_{\max}-1} \mid C_{x,y}(\tau) \mid^2 \end{cases} \tag{5.18}$$

$$MSAC_{x,x} = \begin{cases} \frac{1}{N^2(2\tau_{\max}-2)} \sum_{\tau=1-\tau_{\max},\tau\neq 0}^{\tau_{\max}-1} \mid \theta_{x,x}(\tau) \mid^2 \\ \frac{1}{N^2(2\tau_{\max}-2)} \sum_{\tau=1-\tau_{\max},\tau\neq 0}^{\tau_{\max}-1} \mid \hat{\theta}_{x,x}(\tau) \mid^2 \\ \frac{1}{N^2(2\tau_{\max}-2)} \sum_{\tau=1-\tau_{\max},\tau\neq 0}^{\tau_{\max}-1} \mid C_{x,x}(\tau) \mid^2 \end{cases} \tag{5.19}$$

$$R_{CC} = \frac{1}{K(K-1)(2\tau_{\max}-1)N^2} \sum_{i=0}^{K-1} \sum_{j=0,j\neq i}^{K-1} \sum_{\tau=1-\tau_{\max}}^{\tau_{\max}-1} \mid C_{x_i,x_j}(\tau) \mid \tag{5.20}$$

$$R_{AC} = \frac{1}{K(2\tau_{\max}-2)N^2} \sum_{i=0}^{K-1} \sum_{\tau=1-\tau_{\max}}^{\tau_{\max}-1} \mid C_{x_i,x_i}(\tau) \mid \tag{5.21}$$

5.2 Signal-to-Interference Ratio model

In a DS-CDMA system all signals are transmitted on the same range of frequencies. A receiver is expected to be able to separate a wanted signal from all unwanted signals. There are two stages involved in this demodulation. Firstly, the receiver forms an inner product of a replica of the wanted spreading codeword with all the received signals (see [65]), i.e including the wanted signal. The second stage includes passing the signals through a filter. The filter allows the signal power of the wanted signal to go through unscaled but scales down the signal power of the unwanted signals by a factor of the square of the crosscorrelation observed (scaled between 0 and 1). The ratio of the filter output of wanted signal power to sum of the filter outputs of unwanted signal powers is called the *signal-to-interference ratio* (SIR) of the receiver. The SIR model using the three correlations are modelled in the next subsections. The SIR model presented using the aperiodic correlation (being the basic correlation) is later adopted in the cost function used for optimisation.

5.2.1 SIR model using the even and the odd correlations

In this section the SIR is modelled using the even and the odd correlations. Recall from equations (2.33) and (2.34) that (for unit power) the voltage output of a correlation receiver for the n^{th} bit of the wanted signal (for $\tau \geq 0$) are as follows.

When $b'_n = b'_{n+1}$:

$$\hat{Z}_n = b_n \theta_{x,x}(0) + b'_n \theta_{x,y}(\tau) : 0 \leq \tau \leq N-1. \tag{5.22}$$

When $b'_n = -b'_{n+1}$:

$$\hat{Z}_n = b_n \theta_{x,x}(0) + b'_n \hat{\theta}_{x,y}(\tau) : 0 \leq \tau \leq N-1. \tag{5.23}$$

where b_n and x are the n^{th} message bit and the spreading codeword of the wanted signal, b'_n and y are the interfering message bit and spreading codeword of the unwanted signal respectively.

Thus, if the wanted signal (with codeword x) and a single interfering signal (with codeword y) have equal powers (in watts), the signal-to-interference ratio at the output of the filter is:

$$\left(\frac{\theta_{x,x}(0)}{\theta_{x,y}(\tau)}\right)^2$$

or

$$\left(\frac{\theta_{x,x}(0)}{\hat{\theta}_{x,y}(\tau)}\right)^2$$

with each case expected to arise 50% of the time. This is easily generalised to the case of unequal signal powers at the input to the filter.

Suppose that at receiver point r the signal received from transmitter i is given by $P_i\ L_{r,i}$, where P_i is the power of transmitter i and $L_{r,i}$ is the path loss from transmitter i to receiver r. In this work it is assumed that all *antenna gains* are the same. However, these gains can be incorporated into $L_{r,i}$ if necessary. Equally, if no detailed data is available, $L_{r,i}$ is taken to be $d_{r,i}^{-3.5}$ where $d_{r,i}$ is the distance between receiver r and transmitter i. Thus the signal-to-interference ratio at the output of the filter at receiver r is:

$$\frac{P_i\ L_{r,i}}{P_j\ L_{r,j}} \left(\frac{\theta_{x,x}(0)}{\theta_{x,y}(\tau)}\right)^2 = \frac{P_i\ L_{r,i}}{P_j\ L_{r,j}} \frac{N^2}{(\theta_{x,y}(\tau))^2}$$

or

$$\frac{P_i\ L_{r,i}}{P_j\ L_{r,j}} \left(\frac{\theta_{x,x}(0)}{\hat{\theta}_{x,y}(\tau)}\right)^2 = \frac{P_i\ L_{r,i}}{P_j\ L_{r,j}} \frac{N^2}{\left(\hat{\theta}_{x,y}(\tau)\right)^2}.$$

If more than one interferer is considered:

$$SIR_r = \frac{S_i}{\sum\limits_{j,j\neq i} I_j} = \frac{P_i\ L_{r,i}}{\sum\limits_{j,j\neq i} \left(\Phi_{x,y_j}(\tau_j)\right)^2\ P_j\ L_{r,j}} \tag{5.24}$$

where:

$$\Phi_{x,y_j} = \begin{cases} \frac{1}{N^2}\theta_{x,y_j}(\tau_j) \\ \frac{1}{N^2}\hat{\theta}_{x,y_j}(\tau_j) \end{cases}$$

and y_j is the spreading codeword assigned to transmitter j.

5.2.2 SIR model using the aperiodic correlations

In this subsection an SIR model using the aperiodic correlation is presented. Since the aperiodic correlation is the basic correlation, the model is adopted as the SIR model that will be used in the cost function.

Recall from equation (2.32) the output of a correlation receiver (for $\tau \geq 0$) using the aperiodic correlation:

$$\hat{Z}_n = b_n\theta_{x,x}(0) + \left[b'_n C_{x,y}(\tau) + b'_{n+1} C_{x,y}(\tau - N)\right] : 0 \leq \tau \leq N-1, \tag{5.25}$$

and for $\tau < 0$:

$$\hat{Z}_n = b_n\theta_{x,x}(0) + \left[b'_n C_{x,y}(\tau) + b'_{n-1} C_{x,y}(\tau + N)\right] : 1-N \leq \tau < 0. \tag{5.26}$$

Let $\hat{Z}_{x,y}$ be proportional to the receiver output when the wanted signal has codeword x and the unwanted signal has codeword y.

Case 1: When $\tau > 0$ the interference part of the receiver output $\hat{Z}_{x,y}(\tau)$ is calculated as follows:
50% of the time when $b'_n = b'_{n+1}$:

$$\mid \hat{Z}_{x,y}(\tau) \mid^2 = (C_{x,y}(\tau) + C_{x,y}(\tau - N))^2$$

and 50% of the time when $b'_n \neq b'_{n+1}$:

$$\mid \hat{Z}_{x,y}(\tau) \mid^2 = (C_{x,y}(\tau) - C_{x,y}(\tau - N))^2$$

Then on average:

$$\mid \hat{Z}_{x,y}(\tau) \mid^2 = (C_{x,y}(\tau))^2 + (C_{x,y}(\tau - N))^2 \tag{5.27}$$

Case 2: When $\tau < 0$
50% of the time when $b'_n = b'_{n+1}$:

$$| \hat{Z}_{x,y}(\tau) |^2 = (C_{x,y}(\tau) + C_{x,y}(\tau + N))^2$$

and 50% of the time when $b'_n \neq b'_{n+1}$:

$$| \hat{Z}_{x,y}(\tau) |^2 = (C_{x,y}(\tau) - C_{x,y}(\tau + N))^2$$

On average:

$$| \hat{Z}_{x,y}(\tau) |^2 = (C_{x,y}(\tau))^2 + (C_{x,y}(\tau + N))^2 \tag{5.28}$$

Case 3: When $\tau = 0$

$$| \hat{Z}_{x,y}(\tau) |^2 = (C_{x,y}(\tau))^2. \tag{5.29}$$

The SIR model for more than one interferer j with different powers P_j and path losses $L_{r,j}$ is:

$$SIR_r = \frac{P_i.L_{r,i}.N^2}{\sum_{j,j\neq i} | \hat{Z}_{x,y}(\tau) |^2 .P_j \; L_{r,j}} \tag{5.30}$$

with $| \hat{Z}_{x,y}(\tau) |^2$ taking values from case 1, case 2, or case 3 appropriate to the time delays.

5.2.3 SIR model for mean square measure of $\hat{Z}$

For most families of code the value of $\hat{Z}$ in the SIR model of equation (5.30) varies for different values of τ. For this reason one of the performance measures of spreading codes, the mean-square measure, can be used.

The normalised mean-square measures $MSC\hat{Z}$ and $MSA\hat{Z}$ for asynchronous systems are:

$$MSC\hat{Z}_{x,y} = \frac{1}{N^2(2N-1)} \sum_{\tau=1-N}^{N-1} | \hat{Z}_{x,y}(\tau) |^2 \tag{5.31}$$

and

$$MSA\hat{Z}_{x,x} = \frac{1}{N^2(2N-2)} \sum_{\tau=1-N,\tau\neq 0}^{N-1} | \hat{Z}_{x,x}(\tau) |^2 \tag{5.32}$$

For quasi-synchronous operations they are:

$$MSC\hat{Z}_{x,y} = \frac{1}{N^2(2\tau_{\max}-1)} \sum_{\tau=1-\tau_{\max}}^{\tau_{\max}-1} | \hat{Z}_{x,y}(\tau) |^2 \tag{5.33}$$

and

$$MSA\hat{Z}_{x,x} = \frac{1}{N^2(2\tau_{\max} - 2)} \sum_{\tau=1-\tau_{\max},\tau\neq 0}^{\tau_{\max}-1} \mid \hat{Z}_{x,x}(\tau) \mid^2 \tag{5.34}$$

The signal to interference ratio is then taken as:

$$SIR_r = \frac{P_i \ L_{r,i}}{\sum_{j,j\neq i} P_j \ L_{r,j}\psi_{x,y_j}} \tag{5.35}$$

where: $\psi_{x,y_j} = \begin{cases} MSC\hat{Z}_{x,y_j} & \text{if } x \neq y_j \\ MSA\hat{Z}_{x,x} & \text{if } x = y_j \end{cases}$.

This assumes that if $y_j = x$ the two signals will not have identical phases (see section 5.40).

5.2.4 SIR model for peak square measure of $\hat{Z}$

Recall from equations (5.25) and (5.26) the correlation outputs of a correlation receiver for $\tau < 0$ and $\tau > 0$ respectively. The peak of the interference measure $\hat{Z}_{x,y}(\tau)$ is modelled as follows:
1. When $\tau > 0$:

50% of the time when $b'_n = b'_{n+1}$:

$$\mid \hat{Z}_{x,y}(\tau) \mid_{11} = \mid C_{x,y}(\tau) + C_{x,y}(\tau - N) \mid .$$

50% of the time when $b'_n \neq b'_{n+1}$:

$$\mid \hat{Z}_{x,y}(\tau) \mid_{12} = \mid C_{x,y}(\tau) - C_{x,y}(\tau - N) \mid .$$

Define: $\hat{Z}_1 = \max_\tau \{ \mid \hat{Z}_{x,y}(\tau) \mid_{11} \ , \ \mid \hat{Z}_{x,y}(\tau) \mid_{12} \ : \ \tau > 0 \}$

2. When $\tau < 0$:
50% of the time when $b'_n = b'_{n-1}$:

$$\mid \hat{Z}_{x,y}(\tau) \mid_{21} = \mid C_{x,y}(\tau) + C_{x,y}(\tau + N) \mid .$$

50% of the time when $b'_n \neq b'_{n-1}$:

$$\mid \hat{Z}_{x,y}(\tau) \mid_{22} = \mid C_{x,y}(\tau) - C_{x,y}(\tau + N) \mid .$$

Define: $\hat{Z}_2 = \max\{\mid \hat{Z}_{x,y}(\tau) \mid_{21} \quad , \quad \mid \hat{Z}_{x,y}(\tau) \mid_{22} \quad : \quad \tau < 0\}$

3.When $\tau = 0$:

$$\mid \hat{Z} \mid_3 = \mid (C_{x,y}(0) \mid .$$

The normalised peak value ($PSC\hat{Z}$) between two codewords x, y over all τ such that $1 - N \leq \tau \leq N - 1$ is:

$$PSC\hat{Z}_{x,y} = \frac{1}{N^2} . \max\{\hat{Z}_1^2, \hat{Z}_2^2, \hat{Z}_3^2\} \tag{5.36}$$

In the same way, the normalised off Peak Square Autocorrelation of $\hat{Z}$ ($PSA\hat{Z}$) is calculated by replacing every y with an x in cases 1 and 2 above. Note that, since $C_{x,x}(\tau) = C_{x,x}(-\tau)$, then:

$$PSA\hat{Z}_{x,x} = \frac{1}{N^2} . \hat{Z}_1^2 = \frac{1}{N^2} . \hat{Z}_2^2 \tag{5.37}$$

The signal-to-interference ratio for peak planning at reception point r is:

$$SIR_r = \frac{P_i \ L_{r,i}}{\sum_{j, j \neq i} P_j \ L_{r,j} \psi_{x,y_j}} \tag{5.38}$$

where: $\psi_{x,y_j} = \begin{cases} PSC\hat{Z}_{x,y_j} & \text{if } x \neq y_j \\ PSA\hat{Z}_{x,x} & \text{if } x = y_j \end{cases}$.

5.3 Cost function model

Cost functions based on SIR are of current interest in FDMA frequency assignment [6, 33]. An SIR threshold is set for all receivers (e.g. $14dB$[1] or $15dB$). At each reception point the algorithm optimises some measure of the SIR deficit. The SIR deficit is then raised to a power and summed over all reception points. Total coverage is then measured over all reception points with satisfactory SIR (i.e. SIR $\geq$ threshold).

A cost function model for this over all reception points r is then:

$$cost = \sum_r (\max\{0, (\sigma - SIR_r)\})^a \tag{5.39}$$

where σ denotes an SIR threshold and a is an even integer; usually set to 2. Alternatively, the cost could count 1 for each reception point where the SIR is adequate. Dividing by r and multiplying by 100 then gives the percentage coverage.

[1] dB means decibels. Let y be a real number; xdB is $10 \log_{10} y$

5.4 Remarks on codeword re-use

It is important to note that re-use of spreading codewords is subject to a satisfactory re-use distance. This is analogous to a satisfactory frequency re-use distance in FDMA frequency assignment. If this distance is not satisfied, a DS-CDMA system stands a great danger of either demodulating or incurring an unbearable interference from an unwanted signal using the same codeword. A codeword may only be re-used such that the signal strength of an interfering signal using the same codeword as the wanted signal becomes insignificant at the receiver of the wanted signal.

The unwanted signal I_j in the case of codeword re-use is defined to be:

$$I_j = p_j.L_{r,j}.\psi_{x,x} \tag{5.40}$$

Now, for a constant power p_j, I_j can be reduced if either of $L_{r,j}$ or $\psi_{x,x}$ takes on small values. For example, in the downlink of IS-95, an m-sequence of length $2^{32} - 1$ is re-used in every cell. Neighbouring cells are assigned the same m-sequence but with a phase-shift of 64-chips. This allows $\psi_{x,x}$ in equation (5.40) to take values from the off-peak auto-peak autocorrelation. Then 512 cells can be supported [13]. In WCDMA, there is an adequate number of codewords and so codewords are re-used between very distant cells (or transmitters). In this case $L_{r,j}$ takes on very small values.

In practice when a codeword is re-used, path lengths will be very different and $\psi_{x,x}$ will take on values from the off-peak autocorrelation which are low values over all time delays $\tau \neq 0$ (in the case of asynchronous operation). In the case of quasi-synchronous operation the time delay $\tau \neq 0$ comes from the low (or zero) correlation zone. In other words, $\psi_{x,x} = MSA\hat{Z}_{x,x}$ (for mean planning) or $\psi_{x,x} = PSA\hat{Z}_{x,x}$ (for peak planning) is used in equation (5.40). This prevents the peak autocorrelation value from wrongly penalising any codeword re-use. However, if $\psi_{x,x} = MSA\hat{Z}$ or $\psi_{x,x} = PSA\hat{Z}$ is used during optimisation one of the goals of code assignment is violated. The same spreading codeword is frequently assigned to very close transmitters.

In order to allow the optimization algorithms to take account of this danger of assigning the same spreading codeword to close transmitters, a penalty based approach is adopted. The parameter $\psi_{x,x}$ is set to a multiple of the peak autocorrelation i.e. $\psi_{x,x} = p.\theta_{x,x}(0)$ where p is an integer and the peak autocorrelation is the number of non-zero elements of a codeword. If normalised correlations are used $\psi_{x,x}$ will then be equal to p. Typically, $\psi_{x,x} = 1$ will be used for mean planning and can take on higher values for peak planning. A description of how to choose realistic values of p for peak planning will appear in chapter 8.

This approach of using normalised $\psi_{x,x}$ equal to p whenever a codeword is re-used can be improved by including a codeword re-use threshold in the optimisation process. In this case, $\psi_{x,x} = p$ is used only when the codeword re-use threshold is not satisfied. $\psi_{x,x} = MSA\hat{Z}$ or $\psi_{x,x} = PSA\hat{Z}$ is otherwise used when a codeword is re-used such that the codeword re-use threshold is satisfied. For both methods, the parameter $\psi_{x,x}$ in equation (5.40) is later set to $MSA\hat{Z}$ or $PSA\hat{Z}$ after optimisation in order to evaluate the assignment.

5.5 Evaluation of spreading code assignment

Two approaches will be used for evaluation of a spreading code assignment. Firstly, parameter $\psi_{x,x}$ of equation (5.40) will be changed from $\psi_{x,x} = p$, used during optimisation, to the off-peak autocorrelation $\psi_{x,x} = MSA\hat{Z}$ or $\psi_{x,x} = PSA\hat{Z}$ to calculate the corresponding *true cost*. As explained in section 5.40 path lengths for different signals will be different when synchronisation cannot be maintained and $\psi_{x,x}$ will take on values from the off-peak autocorrelation. The resulting cost of equation (5.39) is then referred to as the true cost. The cost referred to as the *optimisation cost* when $\psi_{x,x} = p$ (during optimisation) is used to prevent assignment of the same code to close transmitters.

Secondly, a spreading codeword re-use will be measured in terms of *signal-to-Co-code-Interference* ratio (SCIR). Let $S = \{x_1, x_2, \ldots, x_T\}$ be an assignment of spreading codewords to T transmitters. At receiver instance r with serving transmitter w, let u be an interfering transmitter such that w and u are assigned the same codeword i. Let S_{wr} be the wanted signal strength at r and S_{ur} be the interfering signal strength at r. Let:

$$\sigma'_{riwu} = \frac{S_{wr}}{S_{ur}}. \qquad (5.41)$$

Now, define $\sigma_{\min}$ as the minimum of σ'_{riwu} over all receiver points r served by a transmitter w with interferer u such that u and w are assigned the same codeword i ($x_w = x_u = i$), i.e.

$$\sigma_{\min} \;=\; \min_{r,i,w,u}\{\sigma'_{riwu} \mid (w \text{ serves } r) \wedge (x_w = x_u = i)\}.$$

An assignment with satisfactory true cost C (using $\psi_{x,x} = MSA\hat{Z}$) will be considered satisfactory if $\sigma_{\min} \geq \sigma'$ for some chosen codeword re-use threshold σ'. Usually σ' will be set to $14dB$.

5.6 Initial prediction of the minimum number of codewords required

To estimate the minimum number of codewords needed, spreading code assignment problems will first be solved using a straightforward graph colouring approach. This can be formulated as follows:
Let G be a graph with vertices representing transmitters. Suppose initially that all transmitters are assigned the same codeword. Given two vertices i and j of the graph, then an edge i, j is present if either:

- for any reception point r_i served by i, $\frac{S_{i,r_i}}{S_{j,r_i}} \leq \sigma'$ or
- for any reception point r_j served by j, $\frac{S_{j,r_j}}{S_{i,r_j}} \leq \sigma'$,
  where S_{ir} is the signal strength of the signal sent from transmitter i to receiver r and σ' is the threshold for codeword re-use.

It can be seen that two adjacent vertices must be assigned different codewords. The minimum number of codewords required is the minimum number of colours required to colour the graph.

The software developed for this project can be adjusted to solve this problem. The normalised $\psi_{x,y}$ between all pairs x, y of codewords are set to zero for distinct codewords and the value $\psi_{x,x}$ is set to one when a codeword x is being re-used. Specifically, the value $\psi_{x,x} = 1$ is applied when two vertices of the graph define above are joined and assigned the same codeword x. Otherwise, $\psi_{x,x} = 0$ is used. For all problems considered in this work σ' is set to $14dB$. The algorithm stops only when the cost is zero and coverage is 100%. This approach is similar to the minimum span problem of FDMA described in [73].

5.7 Estimation of the correlation required for a problem

The approach stated above in section 5.6 could only be used to suggest the minimum number of codewords required but not the correlation required for a satisfactory spreading code assignment. Interference in DS-CDMA depends on the correlation of the spreading code used. Therefore, having an estimate of the correlation (and the implied length) of spreading code required for a problem is important. An approximate estimate of the required correlation using a reception point with maximum interference potential is formulated as follows.

Recall from equations (5.38) and (5.35) the formula for SIR. An SIR_r

at reception point r is considered satisfactory if it is equal to or greater than a set threshold σ, i.e.:

$$SIR_r = \frac{P_i \ L_{r,i}}{\sum\limits_{j,j\neq i} P_j \ L_{r,j}\psi_{x,y_j}} \geq \sigma, \tag{5.42}$$

where:
P_i = the signal power of wanted transmitter i at its reception point r.
P_j = the signal power of unwanted transmitter j at the reception point r.
$L_{r,i}$ = the signal path loss of wanted transmitter i at its reception point r.
$L_{r,j}$ = the signal path loss of unwanted transmitter j at the reception point r.
ψ_{x,y_j} = the $MSC\hat{Z}$, $PSC\hat{Z}$, $MSA\hat{Z}$ or $PSA\hat{Z}$ between wanted spreading code x and unwanted spreading code y.

Let $SIR_{r_{\min}}$ represents the SIR at reception point r with the worst potential for interference. Define *maximum interference potential* by:

$$IP_{\max} = \max\left\{\frac{\sum_{j,j\neq i} P_j L_{r,j}}{P_i L_{r,i}}\right\} \text{ over all reception points } r \tag{5.43}$$

and assume that $\psi_{x,y}$ is taken as the same value $\Phi_{x,y}$ for all pairs x, y of codewords (including when a codeword is re-used). If $SIR_{\min} \geq \sigma$ then SIR's at other reception points are also satisfactory.

Then:

$$SIR_{\min} = \frac{1}{IP_{\max}\Phi_{x,y}} \geq \sigma \tag{5.44}$$

which implies that:

$$\Phi_{x,y} \leq \frac{1}{IP_{\max}\sigma} \tag{5.45}$$

Any value (i.e. $MSC\hat{Z}$, $MSA\hat{Z}$, $PSC\hat{Z}$ and $PSA\hat{Z}$) of a code at an SIR threshold σ should on average satisfy the bound presented in equation (5.45). With good spreading code assignment (as in the case where codeword re-use is necessary) a 100% coverage at a certain threshold σ could be guaranteed if all correlation measures of a code satisfy this bound. In any case the average of all correlation values of a spreading code should satisfy equation (5.45). As a standard adopted in this work the length of code required to make a Gold code (or Large set of Kasami sequences) achieve the bound presented in equation (5.45) will be used as a yardstick. Any longer code (say double the length of the Gold code) will not be considered suitable. This is because too much frequency spectrum will then be required.

5.8 Discussion

This chapter has discussed the methodology adopted for spreading code assignment. SIR models presented are based on the even, the odd and the aperiodic correlations respectively. Spreading code measures using the aperiodic correlation are used in the SIR model for the cost function. The cost function is based on measuring some SIR deficits at each reception point.

Two methods of network planning proposed are mean and peak planning using $MSC\hat{Z}$ and $PSC\hat{Z}$. This research is more interested in mean planning because in some families of codes (e.g. LS code and Gold code) the peak values only occur a few times in the correlation distribution. In particular, the peak even correlation of a Gold code of length 1023 is found to occur 3% of the time. However, an example of peak planning will appear in one of the spreading code assignment examples presented in this work.

The chapter also analyses how re-use of spreading codewords should be treated. During optimisation, the penalty $\psi_{x,x} = 1$ is used as the normalised correlation when a codeword is re-used such that the re-use threshold is not satisfied. $\psi_{x,x}$ is allowed to take on values from $MSA\hat{Z}$ (or $PSA\hat{Z}$) when a codeword is re-used such that the codeword re-use threshold is satisfied. A method of predicting the minimum number of spreading codes needed for a particular network has been proposed. A simple approach to the estimation of the average code correlation required for a network has also been formulated. The length of code required to make a Gold code achieve this estimated average correlation can be used as an indication of the required code length.

Chapter 6

Optimisation Algorithms Adapted for Spreading Code Assignment

6.1 Overview

As with frequency assignment problems, spreading code assignment problems arise when large number of transmitters operating in a region interfere with each other. This occurs as a result of non-zero correlations between their respective codewords. The problem can also arise when available spreading codewords are to be re-used subject to a specified quality of service such as a codeword re-use threshold measured in some way (see section 5.6 for a description).

In this chapter, algorithms of interest for solving spreading code assignment shall be described. These algorithms are examples of metaheuristic algorithms described by Glover et al. in [23]. Given a current solution, small changes are made to generate a neighbourhood of the current solution. The algorithms then decide on the acceptance of a solution in the neighbourhood as the new solution. These algorithms have the ability to escape from local minima of their cost function. Other algorithms used in frequency assignment can be found in [74, 75, 77].

The data structures and cost function updating process are formulated to reflect excellent time efficiency. When a request for a new assignment is made, interference must be calculated at all receiver instances. Total interference experienced at each receiver instance is calculated from the current total interference in a time efficient way. Average interference contributions (due to non-zero correlation and re-use of codewords) to a non-zero total cost are also calculated.

6.2 General description of terminologies

6.2.1 Assignment representation

Let $K = \{x_1, x_2, \ldots, x_{|K|}\}$ be a spreading code such that each $x_i = (x_{i,0}, x_{i,1}, \ldots, x_{i,N-1})$ $(i \in K)$ is a codeword of length N. Let $T = \{Tr_1, Tr_2, \ldots, Tr_{|T|}\}$ be a set of transmitters. An assignment $S = \{s_1, s_2, \ldots, s_{|T|}\}$ of a spreading code to $|T|$ transmitters is a mapping of each transmitter Tr_j to a spreading codeword x_i i.e. $s_j = x_i$ for some $j \in \{1, 2, \ldots, T\}$ and $i \in \{1, 2, \ldots, |K|\}$.

6.2.2 Neighbourhood of an assignment

An assignment $S' = \{s'_1, s'_2, \ldots, s'_{|T|}\}$ is a neighbour of an assignment S if it differs from S in one component. That is, there exists a $j \in T$ such that $s'_j \neq s_j$ and $s'_i = s_i$ for all other $i \neq j$ in T. The full neighbourhood of an assignment consists of all such assignments that differ from the current assignment in the codeword assigned to a single transmitter. The full neighbourhood size is $|T| \times (|K| - 1)$.

6.2.3 Cost function

A cost function C measures the interference associated with an assignment S. The aim of all the algorithms described here is to minimise C. The cost function used for mean and peak planning has been described in section 5.3.

6.3 Simulated annealing algorithm

Simulated annealing is a metaheuristic algorithm originally derived from statistical mechanics. It is governed by a temperature denoted T_t at time t. The system starts with a high temperature T_0 and cools down over time to a low temperature denoted $T_{\min}$ at time t according to a cooling schedule such as a geometric cooling schedule $T_{t+1} = \alpha T_t$ $(0 < \alpha < 1)$. A temperature T_t is fixed for **Nloop** iterations before it can be reduced. Other types of cooling schedule can be found in [74]. The system starts with an initial assignment S_{old} of cost C_{old}; it proceeds by accepting a new assignment in the (full) neighbourhood of S_{old} (with resulting cost C_{new}) if the cost is reduced; but it is sometimes allowed to increase to escape from local minima. Letting $\Delta C = C_{\text{new}} - C_{\text{old}}$, then the new assignment is accepted if $\Delta C \leq 0$ or if $\Delta C > 0$ and a pseudo-random number between 0 and 1 is less than a probability $e^{-(\Delta C / k T_t)}$. Here k is a constant which can be set as 1 as it only scales the temperature $T_{\min}$. The algorithm terminates when the temperature is less than a predefined finishing temperature $T_{\min}$ or the current best cost C_{best} is equal to a predefined minimum cost $C_{\min}$ or no improved solution has been found for a predefined number of iterations. The method itself has a direct analogy with thermodynamics, in the way that liquids freeze and crystallise, or metals cool and anneal. Pseudo-code for simulated annealing is shown in algorithm 1.

For the work presented here, the initial temperature is $T_0 = 1$, **Nloop = 100**, the cooling schedule is $T_{t+1} := T_t \times 0.999999$ and the constant is $k = 1$. The simulated annealing algorithm will be allowed to stop if no better solution than the current best solution is obtained over $C_{\text{limit}} = 10^5$ iterations or if current temperature T_t is less than $T_{\min} = 1.0 \times 10^{-15}$ or if the current best cost C_{best} is equal to $C_{\min} = 0$.

Algorithm 1 (Pseudo-code for a simulated annealing algorithm)

Initialise temperature T_0*;*
Initialise minimum temperature $T_{\min}$
Initialise minimum cost $C_{\min}$
Initialise **Nloop**
Initialise best solution iteration limit C_{limit}
Initialise best solution count $C_{\text{count}} = 0$
Generate a random solution S_{old} *and*
Calculate cost C_{old}*;*
$S_{\text{best}} = S_{\text{old}}$;
$C_{\text{best}} = C_{\text{old}}$;
$C_{\text{count}} = C_{\text{count}} + 1$;
While $(T_t > T_{\min}$ and $C_{\text{best}} \neq C_{\min}$ and $C_{\text{count}} \leq C_{\text{limit}})$
    For $i = 1$ *to* **Nloop** *do*
        generate new solution S_{new} *in the neighbourhood of* S_{old}*;*
        calculate new cost C_{new}*;*
        calculate $\Delta C = C_{\text{new}} - C_{\text{old}}$*;*
        If $\Delta C \leq 0$ *or*
            $\Delta C > 0$ *and* $(\text{random} \in (0,1)) < \text{prob} = e^{-(\Delta C/kt)}$
            then;
            $S_{\text{old}} = S_{\text{new}}$;
            $C_{\text{old}} = C_{\text{new}}$;
            If $C_{\text{old}} < C_{\text{best}}$ *then;*
                $S_{\text{best}} = S_{\text{old}}$;
                $C_{\text{best}} = C_{\text{old}}$;
                $C_{\text{count}} = 0$;
            End if;
        End if;
        $C_{\text{count}} = C_{\text{count}} + 1$;

End for
$T_{t+1} = \alpha T_t$ *where* $0 < \alpha < 1$*;*
$t := t + 1$*;*
End while;
Return S_{best}, C_{best}.

6.4 Tabu search algorithm

Tabu search is a metaheuristic algorithm. It is a local search algorithm but has a mechanism to escape from local minima by marking some *moves* as being forbidden or tabu. A move from a current assignment is defined as an assignment in the neighbourhood of the current assignment. A move is tabu because of either a short-term memory condition or a long-term memory condition. A short-term memory condition specifies that a move cannot be made if it has been made within the last r moves. A long-term memory condition prevents a transmitter from having its spreading codeword changed too large a proportion of the time. However, a long-term memory condition is usually regarded as counterproductive in frequency assignment. The list of moves for which either the short-term memory condition or the long-term memory condition is satisfied is called the *tabu list.*

Given a current solution, the system makes some *candidate* moves in the neighbourhood of the current solution and marks those that are tabu and those that are not tabu. The candidate move with the smallest cost consequence $C_{nontabu}$ that is nontabu and the tabu candidate move with smallest value of cost consequence C_{tabu} are selected. If the best tabu move is better than the best nontabu move, and gives a new best solution over the entire history of the system, then the tabu move is called an aspirate move and is accepted. Otherwise the best nontabu move is always accepted. Pseudo-code for the tabu search algorithm is given in algorithm 2. Throughout the work presented here a random neighbourhood size of $\lceil 0.25 \times |T| \rceil$, where $|T|$ is the number of transmitters, will be used [77]. A move to a new assignment is declared tabu if the move has been made over the last $\lceil 0.4 \times |T| \rceil$ moves [77]. The tabu search algorithm is allowed to stop if no better solution than the current best solution is obtained over $C_{\text{limit}} = 10^5$ iterations.

Algorithm 2 (Pseudo-code for a tabu search algorithm)

Initialise minimum cost $C_{\min}$
Initialise best solution iteration limit C_{limit}
Initialise best solution count $C_{\text{count}} = 0$
Generate a random solution S_{current}
Calculate current cost C_{current}
$S_{\text{best}} = S_{\text{current}}$*;*

$C_{\text{best}} = C_{\text{current}}$;
$C_{\text{count}} = C_{\text{count}} + 1$;
While ($C_{\text{best}} \neq C_{\text{min}}$ and $C_{\text{count}} \leq C_{\text{limit}}$) *do*
Generate a random neighbourhood of size $\lceil 0.25 \times |T| \rceil$
$S_{\text{best non-tabu}}$ = *best non-tabu move*
S_{aspirate} = *aspirate move*
$S_{\text{current}} = S_{\text{best non-tabu}}$ or S_{aspirate}
$C_{\text{current}} = C_{\text{best non-tabu}}$ or C_{aspirate}
Update tabu list
If $C_{\text{current}} < C_{\text{best}}$ *then;*
$S_{\text{best}} = S_{\text{current}}$;
$C_{\text{best}} = C_{\text{current}}$;
$C_{\text{count}} = 0$
End if
$C_{\text{count}} = C_{\text{count}} + 1$;
End while
Return $S_{\text{best}}, C_{\text{best}}$

6.5 Implementations

6.5.1 Implementation of the tabu list

The short-term memory condition is used. A table of dimension $|T| \times |k|$ (where $|k|$ is the number of codewords) is used to store the iteration number of the last iteration at which the move was made. When a move (i.e. involving the change of a codeword x_i assigned to a single transmitter Tr_j) is accepted by the tabu search algorithm, the iteration number is stored in entry (j, i). A move is then declared tabu if the difference between the current iteration number and the content of entry (j, i) is less than r.

6.5.2 Data structures and cost function updating

One of the characteristics of good software is its effectiveness in minimising run time while maintaining accurate execution of instructions. One of the methods of minimising run time is by carefully designing the data structures in such a way that the entire programme can update effectively. It is also important to design a time efficient cost function updating process. The carefully designed data structure and time efficient cost function updating process are vital if the algorithms are to be practicable for this application. They can be described as follows.

Signal strengths from transmitters at each reception point are stored in a two dimensional array W of dimension $R \times |T|$ such that each entry $W_{r,i}$ contains the corresponding signal strength $P_i.L_{r,i}$: $0 \leq r < R, 0 \leq i < |T|$, from transmitter i at receiver instance r. An interference table of dimension $R \times |T|$ (where R is the number of receiver instances and $|T|$ is the number of transmitters) is maintained. Each entry $E_{r,j} = E_{r_i,j} : 0 \leq r < R, 0 \leq j < |T|$, represents an interference contribution $P_j.L_{r,j}.\psi_{x,y_j}$ (i.e $W_{r,j}.\psi_{x,y_j}$) from an unwanted transmitter j (with spreading codeword y_j) at receiver instance r with serving transmitter i (with spreading codeword x). An example can be seen in tables 6.1 and 6.2. An entry in the interference table relating to a wanted (serving) transmitter i is set to have zero interference value. Total interference at receiver instance r is then the sum of row r. A one dimensional array C of length R is used to store the total interference at each receiver instance. Each $C_r = \sum_j E_{r,j}$, is the sum of each row r.

The associated SIR_r of equation (5.35) or (5.38) at receiver instance r with serving transmitter i is:

$$SIR_r = \frac{W_{r,i}}{C_r}.$$

This is repeated over all receiver instances to measure the cost as in equation (5.39); which is:

$$cost = \sum_r \max\{0, (\sigma - SIR_r)\}^2$$

Again the aim of the optimisation process is to minimise this cost.

When an attempt is made to change the current assignment (say S) to an assignment in its neighbourhood (say S'), the SIR is re-calculated at each reception point but it is updated in an efficient way. Let S' represents an attempt to change the wanted codeword assigned to the transmitter i serving receiver instance r. All the entries $E_{r_i,j}$ in row(s) r for receiver(s) served by i are re-calculated completely as the correlations of all the interfering transmitters j are affected. This means total interference C_r is recalculated completely. At other receiver instances (with corresponding serving transmitters j) where transmitter i is considered to be an interfering transmitter only the interference contribution $E_{r_j,i}$ of unwanted transmitter i is re-calculated. At such receiver instances the new total interference (say C'_r) is calculated from the old total interference C_r as follows:

Let $C_r = \sum_{k=0}^{|T|} E_{r_j,k}$ and $C'_r = \sum_{k=0}^{|T|} E'_{r_j,k}$, such that transmitter j is the serving transmitter and transmitters k are the interfering transmitters. When a new assignment is made with the code assigned to transmitter i changed, all

Receiver Instance	Serving Transmitter
0	0
1	0
2	1
3	1
4	2
5	3
6	4

Table 6.1: *This table shows an example of 7 receiver instances with 5 corresponding serving transmitters*

0	$E_{0_0,1}$	$E_{0_0,2}$	$E_{0_0,3}$	$E_{0_0,4}$
0	$E_{1_0,1}$	$E_{1_0,2}$	$E_{1_0,3}$	$E_{1_0,4}$
$E_{2_1,0}$	0	$E_{2_1,2}$	$E_{2_1,3}$	$E_{2_1,4}$
$E_{3_1,0}$	0	$E_{3_1,2}$	$E_{3_1,3}$	$E_{3_1,4}$
$E_{4_2,0}$	$E_{4_2,1}$	0	$E_{4_2,3}$	$E_{4_2,4}$
$E_{5_3,0}$	$E_{5_3,1}$	$E_{5_3,2}$	0	$E_{5_3,4}$
$E_{6_4,0}$	$E_{6_4,1}$	$E_{6_4,2}$	$E_{6_4,3}$	0

Table 6.2: *An example of an interference table with 7 receiver instances and 5 transmitters*

$E_{r_j,k} = E'_{r_j,k}$ for all $k \neq i$, $j \neq i$. It is then easy to see that

$$C'_r = C_r - E_{r_j,i} + E'_{r_j,i}. \tag{6.1}$$

Suppose the spreading codeword assigned to transmitter 1 in table 6.1 is changed. The shaded parts of table 6.2 relating to $i = 1$ are the only entries affected as a result of these changes. That is all entries in table 6.2 apart from the shaded entries are equal to the corresponding entries in table 6.3. Each total interference C'_2 and C'_3 at receiver instances 2 and 3 served by transmitter 1 are re-calculated completely. At other receiver instances $0, 1, 4, 5,$ and 6 not served by transmitter 1, each of $C'_0, C'_1, C'_4, C'_5,$ and C'_6 can be calculated using equation (6.1). The associated SIR_r at such receiver instance can then be calculated and the associated cost is updated.

0	$E'_{0_0,1}$	$E'_{0_0,2}$	$E'_{0_0,3}$	$E'_{0_0,4}$
0	$E'_{1_0,1}$	$E'_{1_0,2}$	$E'_{1_0,3}$	$E'_{1_0,4}$
$E'_{2_1,0}$	0	$E'_{2_1,2}$	$E'_{2_1,3}$	$E'_{2_1,4}$
$E'_{3_1,0}$	0	$E'_{3_1,2}$	$E'_{3_1,3}$	$E'_{3_1,4}$
$E'_{4_2,0}$	$E'_{4_2,1}$	0	$E'_{4_2,3}$	$E'_{4_2,4}$
$E'_{5_3,0}$	$E'_{5_3,1}$	$E'_{5_3,2}$	0	$E'_{5_3,4}$
$E'_{6_4,0}$	$E'_{6_4,1}$	$E'_{6_4,2}$	$E'_{6_4,3}$	0

Table 6.3: *A new interference table generated as a result of a random change in the codeword assigned to transmitter* $i = 1$*.*

6.5.3 Evaluation using average interference contribution of correlations

For assignment evaluation, $\psi_{x,x}$ is set to $MSA\hat{Z}$ for mean planning or $PSA\hat{Z}$ for peak planning. It will be useful to analyse the average interference contribution due to re-use of spreading codewords and due to the correlations between distinct codewords measured by $\psi_{x,y} = MSC\hat{Z}$ or $\psi_{x,y} = PSC\hat{Z}$ (depending on the type of planning).

At a receiver instance r (with wanted transmitter i) where the SIR is inadequate (i.e $SIR < threshold$), percentages of interference due to re-use of the wanted spreading codeword and $\psi_{x,y}$ correlations are calculated. At such row r in the interference table, entries $E_{r_i,j} : j \neq i$ relating to interference contribution(s) due to re-use of the wanted codeword and $\psi_{x,y}$ are denoted as $cocode\ interference$ and $intercode\ interference$ respectively. In other words:

$$cocode\ interference_r = \sum E_{r_i,j_{cocode}}$$

and

$$intercode\ interference_r = \sum E_{r_i,j_{intercode}}.$$

Percentage of interference contribution due to codeword re-use ($cocode\%$) at receiver instance r is then:

$$cocode_r\% = \frac{cocode\ interference_r}{C_r} \times 100$$

and that due to $\psi_{x,y}$ correlations is:

$$intercode_r\% = \frac{intercode\ interference_r}{C_r} \times 100.$$

Let the total number of receiver instances where SIR is inadequate be K and denote these indices as $i_1, i_2, \ldots, i_K$. Average percentages of interference contributions due to codeword re-use (average $cocode\%$ $ACC\%$) and $\psi_{x,y}$ (average $intercode\%$ $AIC\%$) are then:

$$ACC\% = \frac{\sum_{j=0}^{K-1} cocode_{i_j}\%}{K}$$

and

$$AIC\% = \frac{\sum_{j=0}^{K-1} intercode_{i_j}\%}{K}$$

respectively.

Chapter 7

A 458 Transmitters Problem - Mean Planning

In this chapter a 458 transmitter problem is presented. The way that the optimisation algorithms are adapted and some details of the application of the algorithm are given.

This problem consists of 458 transmitters generated using a benchmark generator introduced in [73]. A set of 458 distinct locations were generated using a non-uniform distribution. Reception points are located at the vertices of *Voronoi polygons*. A Voronoi polygon consists of points closer to a transmitter than any other transmitter. A total of 976 distinct receiver locations (which give a total of 2675 receiver instances) are generated. The distribution of the 458 transmitters and 976 receiver locations covers a 20km $\times$ 20km square grid. When the system is operated quasi-synchronously it is assumed that $-15 \leq \tau \leq 15$.

7.0.4 Analyses of problem

This problem is a non-cellular problem. It is assumed that synchronisation can be maintained only with the wanted signal transmitted by the serving transmitter. Some synchronisation uncertainties exist for the interfering signals. The network is then assumed to operate asynchronously when a Gold code or a Small or Large Kasami set is used. It is assumed to operate quasi-synchronously when an LS code or an LCS code is used with $-15 \leq \tau \leq 15$.

The algorithms used are the metaheuristic algorithms described in chapter 6. For the simulated annealing algorithm, the cooling schedule t is set to $t := t \times 0.999999$ (a very slow cooling schedule) and a hundred moves are allowed at each temperature. A full neighbourhood is used. For the tabu search algorithm, a random neighbourhood of size $0.25 \times |T|$ is used [77], where $|T|$ is the number of transmitters, i.e $\lfloor 0.25 \times 458 \rfloor = 115$ is used. As explained in section 6.2.2, a neighbourhood of an assignment is an assignment which differ the code assigned to one transmitter. A move to a new assignment is declared tabu if the move has been made over the last $\lfloor 0.4 \times |T| \rfloor = 183$ moves. The two algorithms are allowed to stop if no better solution than the best solution is obtained over 10^5 iterations.

The solution to the initial assignment experiment in section 5.6 suggested that at least 30 codewords are required for satisfactory codeword re-use at $\sigma' = 14dB$ (i.e. $14dB$ codeword re-use threshold). The value of $IP_{\max}$ (i.e. maximum interference potential) of equation (5.43) is 12.9127 and taking $\sigma = 14dB$ in equation (5.45), $\Phi_{x,y}$ must be less than or equal to 3.08×10^{-3}. As effective frequency spectrum management is part of the aims of this work, the smallest code length to make a Gold code or a Large set of Kasami sequences satisfy the correlation bound is desired. Code length is directly proportional to the frequency spectrum required, so small code lengths are preferable provided the number of codewords is not too small. A sample of 1,000 consecutive codewords from a Large Kasami set of length 255 has an average $MSC\hat{Z}$ of 3.92×10^{-3}. This is close to $\Phi_{x,y} = 3.08 \times 10^{-3}$ at $\sigma = 14dB$ but a 100% coverage may not be obtained. Nevertheless, a code length of 255 will be used initially.

7.0.5 Software specification

The cost function given by equations (5.35), (5.38) and (5.39) is used for this problem. Parameter a in the SIR cost function is set to 2, signal path loss $L_{r,i}$ between receiver r and transmitter i is modelled as $d_{r,i}^{-3.5}$ (where $d_{r,i}$ is the distance between r and i) and equal signal powers are assumed. In other words for this problem (and in subsequent problems),

$$\text{cost} = \sum_r \max\{0, (\sigma - SIR_r)\}^2,$$

where receiver point r has serving transmitter i and

$$SIR_r = \frac{d_{r,i}^{-3.5}}{\sum_{j,j \neq i} d_{r,j}^{-3.5} . \psi_{x_i,y_j}},$$

and $\psi_{x,y_j} = MSC\hat{Z}_{x_i,y_j}$ (mean planning) or $PSC\hat{Z}_{x_i,y_j}$ (peak planning) if $x_i \neq y_j$. For mean planning, $\psi_{x,x} = 1$ is used when a codeword is re-used such that the codeword re-use threshold σ' is not achieved during the optimisation process; $\psi_{x_i,y_j} = MSA\hat{Z}_{x,x}$ is used otherwise. Again, it will be useful to emphasis that the true value of $\psi_{x,x} = MSA\hat{Z}_{x,x}$ or $\psi_{x,x} = PSA\hat{Z}_{x,x}$ is used for evaluation (as justified in sections 5.40 and 5.5). Also, the codeword re-use threshold σ' will be set to $14dB$. Again, the cost and area coverage obtained during the optimisation process (i.e. when using $\psi_{x,x} = 1$) will be referred to as the optimisation cost and coverage respectively. The cost and area coverage obtained during evaluation (i.e. when $\psi_{x,x} = MSA\hat{Z}_{x,x}$ or $PSA\hat{Z}_{x,x}$ is used) is referred to as the true cost and coverage respectively. The two costs are the same for an assignment with satisfactory codeword re-use. Justifications for these differences have been given in sections 5.40 and 5.5.

7.1 Assignment of LCS codes using $MSC\hat{Z}$

The LCS code constructed here is described in section 3.8. Integers m and n are set to 4 and 8 respectively to construct 15 LCS codewords of length 255. Average $MSC\hat{Z}$ is 2.52×10^{-4} which is less than $\Phi_{x,y} = 3.08 \times 10^{-3}$ required. A tabu search algorithm is used to solve this assignment problem. $\psi_{x,x}$ is set to 1 during optimisation and set to $MSA\hat{Z}_{x,x}$ during evaluation. Figure 7.1 and table 7.1 show that $MSC\hat{Z}$ for LCS codewords differs from pair to pair. The degree of the variation may be insignificant as the correlations are very small. The tabu search can still work on this variation to minimise the resulting interference.

At an SIR threshold of $13dB$, a random assignment using $\psi_{x,x} = 1$ gave an optimisation cost of $350,189$ and 40.6355% coverage. σ_{min} at this stage is $0dB$ which is less than σ' and so re-use of spreading codes is unsatisfactory. However, this initial random assignment (when evaluated) is equivalent to a zero true cost and 100% coverage. This is expected as average correlation is far less than $\Phi_{x,y}$. Nevertheless, the assignment cannot be accepted as $\sigma_{\min} = 0dB$ is less than $\sigma' = 14dB$.

The tabu search algorithm continues with the optimisation process and finished with an optimisation cost of $38,264.1$ and total coverage of 73.1589% (obtained using $\psi_{x,x} = 1$). Evaluating this optimised assignment using $\psi_{x,x} = MSA\hat{Z}_{x,x}$, a zero true cost with 100% coverage is obtained. The $\sigma_{\min}$ achieved is $6dB$ which is still less than σ'. The minimum spreading codeword re-use achieved by this assignment is unsatisfactory and so the assignment cannot be accepted.

This indicates that even though the correlations of the LCS code are good, it is certainly hopeless to assign too few codewords to a large number of transmitters. Indeed for this problem 30 codewords or more are required for satisfactory codeword re-use.

An approach introduced in [35] and also described in section 3.8 is then used to achieve more LCS codewords of length 255. Two classes of LCS codes both of length 255 with (n, m) = (8,4) (i.e 15 codewords in each class) are combined to give 30 codewords. The two code classes use the same seed vectors

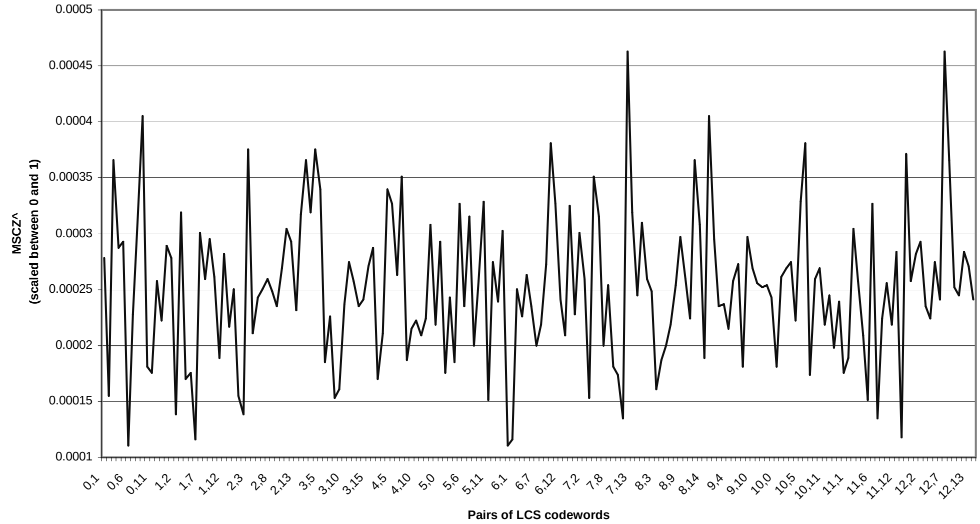


Figure 7.1: *Graph showing how the normalised $MSC\hat{Z}$ of a single set of 15 LCS codewords of length 255 varies for different pairs.*

$MSC\hat{Z}$ Range	$MSC\hat{Z}$ distribution
7 - 10	18
11 - 14	48
15 - 18	86
19 - 22	42
23 - 26	14
27 - 30	2

Table 7.1: *Table to show the distribution of unnormalised $MSC\hat{Z}$ in a single set of LCS codewords (15 codewords) of length 255.*

but different shift sequences. This approach is of course disadvantaged by worse crosscorrelations between code classes than within code classes. But there are 30 codewords and given good code assignment it could lead to lower overall interference in the system than codes in use today.

It is important to use a pair of shift sequences that will result in the best possible correlations both within and between LCS code classes. There are 8 primitive polynomials and so only 8 shift sequences are possible. An exhaustive search for the best pair of shift sequences resulting in the least total mean-square value of $\hat{Z}$ was carried out. It is evident from table 7.2 that the least total $MSC\hat{Z}$ over all codeword pairs occurs when primitive polynomials 551 and 543 (in octal) or $x^8 + x^6 + x^5 + x^3 + 1$ and $x^8 + x^6 + x^5 + x + 1$ are used to generate a pair of shift sequences.

Primitive polynomial pair(in octal)	$\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} MSC\hat{Z}_{x_i,y_j}$
551 and 543	2.54733
453 and 545	2.54995
435 and 703	2.55556
747 and 543	2.64694
543 and 537	2.64694
747 and 537	2.64826
545 and 543	2.64945
435 and 747	2.65016
435 and 453	2.65183
545 and 703	2.65219
551 and 545	2.69586
551 and 453	2.69633
545 and 543	2.69681
453 and 543	2.69729
747 and 545	2.69812
435 and 537	2.69884
551 and 703	2.69932
435 and 551	2.69944
543 and 703	2.70027
435 and 543	2.70039
747 and 453	2.70051
537 and 703	2.70063
551 and 537	2.74716
551 and 747	2.74811
747 and 703	2.7493
453 and 537	2.75014
435 and 545	2.75062
453 and 703	2.75193

Table 7.2: *Pairs of primitive polynomials and total normalised $MSC\hat{Z}$ obtained by the combination of their two corresponding LCS code classes of length $N = 255$.*

Average $MSC\hat{Z}_{x,y}$ between two distinct codewords x and y within the same LCS code classes and between different LCS code classes are 2.5×10^{-4} and 3.4×10^{-3} respectively, which are both less than $\Phi_{x,y}$ for this problem. A pair of transmitters that are close together then ideally need to be assigned LCS codewords from the same LCS code class and pairs of transmitters that are farther apart may be assigned LCS codewords from different LCS code classes.

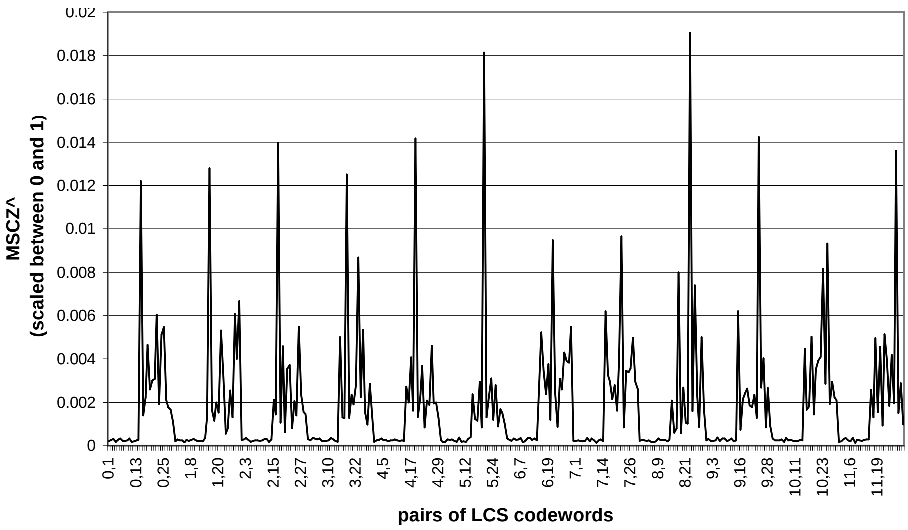


Figure 7.2: *Graph showing how the normalised $MSC\hat{Z}$ for 2 sets of LCS codewords differ between pairs. It also shows that $MSC\hat{Z}$ between LCS code classes are much worse than within the same class. Note that $MSC\hat{Z}$ within each class also differs from pair to pair (see Figure 7.1)*

Evidence from figures 7.1, 7.2 and table 7.3 demonstrates that the assignment method using $MSC\hat{Z}$ is not a type of graph colouring method as the $MSC\hat{Z}$ values differ from pair to pair. The system is expected to cope with both re-use of spreading codewords and better optimisation of the use of $MSC\hat{Z}$.

7.1.1 Assignment of two sets of LCS codewords using $MSC\hat{Z}$

Different SIR thresholds are set for different assignments. The optimisation methods adapted are tabu search and simulated annealing algorithms with $\psi_{x,x} = 1$ for optimisation. Results obtained by simulated annealing will first be examined.

At $14dB$, a random initial assignment using simulated annealing started with an optimisation cost of $269,102$ which resulted in a total coverage of 71.1776%; on average 93.213% ($ACC\%$) of this cost is due to codeword re-use while $AIC\%$ is just 6.787%. Although this assignment has a true cost of zero when evaluated with $\psi_{x,x} = MSA\hat{Z}_{x,x}$, the minimum codeword re-use obtained is very unsatisfactory as $\sigma_{\min}$ is $0dB$ which is less than σ'. Simulated annealing then optimised the optimisation cost with $\psi_{x,x} = 1$ to finish with an optimisation cost of zero and a 100% total coverage. $\sigma_{\min}$ achieved is $16dB$ which is greater than $\sigma' = 14dB$ required. The true cost is zero with 100% coverage. Thus a satisfac-

$N^2.MSC\hat{Z}$ Range	$MSC\hat{Z}$ distribution
6 - 71	474
72 - 137	138
138 - 203	98
204 - 269	54
270 - 335	40
336 - 401	20
402 - 467	6
468 - 533	8
534 - 599	2
600 - 665	6
666 - 731	2
732 - 797	2
798 - 863	6
864 - 929	10
1128 - 1193	2
1194 - 1259	2

Table 7.3: *Table to show the distribution of unnormalised $MSC\hat{Z}$ in the two sets of LCS codewords (30 codewords) of length 255.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	0
15	45.5689	99.8879	0.0264673	99.9735	0
16	2233.19	99.3271	0.251257	99.7487	0
17	12564.8	95.0654	0.131978	99.868	0

Table 7.4: *This table shows the unsatisfactory results of the initial random assignments of two sets of LCS codewords. Note that the cost is the true cost and the codeword re-use threshold σ' is not satisfied.*

tory spreading code assignment is achieved by the simulated annealing algorithm.

The same approach is followed for other SIR thresholds (i.e. $15dB$, $16dB$, $17dB$). The results obtained are presented in tables 7.4 and 7.5. In these cases the true cost obtained by random initial assignments are not zero; much of these costs are due to $MSC\hat{Z}$ between LCS codewords. The resulting codeword re-use values $\sigma_{\min}$ of these assignments are also unsatisfactory. The optimised results presented in table 7.5 however gave zero costs with 100% coverages and satisfactory codeword re-use $\sigma_{\min} \geq \sigma'$.

A similar approach can be followed using the tabu search algorithm. Tabu search

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	16
15	0	100	0	0	16
16	0	100	0	0	14
17	0	100	0	0	14

Table 7.5: *This table shows the satisfactory results of the optimised assignments obtained using the simulated annealing algorithm. All the assignments achieved codeword re-use* $\sigma_{\min} \geq \sigma'$.

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	0
15	89.8909	99.7383	0.0926654	99.9073	0
16	1943.57	98.7664	0.0949261	99.9051	0
17	10434.8	96.0748	0.122077	99.8779	0

Table 7.6: *This table shows the unsatisfactory results of the initial random assignments of the two sets of LCS codewords obtained. Note that the cost is the true cost and the codeword re-use threshold* σ' *is not satisfied.*

started with a random solution which results in an optimisation cost (with $\psi_{x,x} = 1$) of 329,107 and a 65.7196% total coverage. $ACC\%$ and $AIC\%$ interferences are 93.2676% and 6.73244% respectively. Corresponding true cost (with $\psi_{x,x} = MSA\hat{Z}_{x,x}$) obtained is zero but $\sigma_{\min}$ is $0dB$ which is less than σ'. Thus the minimum spreading codeword re-use $\sigma_{\min}$ obtained is unsatisfactory. Tabu search then finished with an optimisation cost (with $\psi_{x,x} = 1$) of zero and a 100% total coverage. $\sigma_{\min}$ achieved is $15dB$ which is greater than σ' and so a satisfactory spreading code re-use is achieved. The true cost is zero and $\sigma_{\min}$ is greater than σ'.

The results obtained at other thresholds are presented in tables 7.6 and 7.7. It clear that the optimised results obtained by the tabu search algorithm gave satisfactory spreading code assignments. The costs are zero with 100% coverages and satisfactory codeword re-use.

Comparing the random initial assignments obtained by both algorithms with the optimised assignments, it is clear that there is a great advantage in careful assignment (such as the assignments presented by simulated annealing and tabu search) when a combination of two sets of LCS codewords is used as a spreading code.

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	15
15	0	100	0	0	16
16	0	100	0	0	15
17	0	100	0	0	14

Table 7.7: *This table shows the satisfactory results of the optimised assignments obtained using the tabu search algorithm. All the assignment achieved* $\sigma_{\min} \geq \sigma'$.

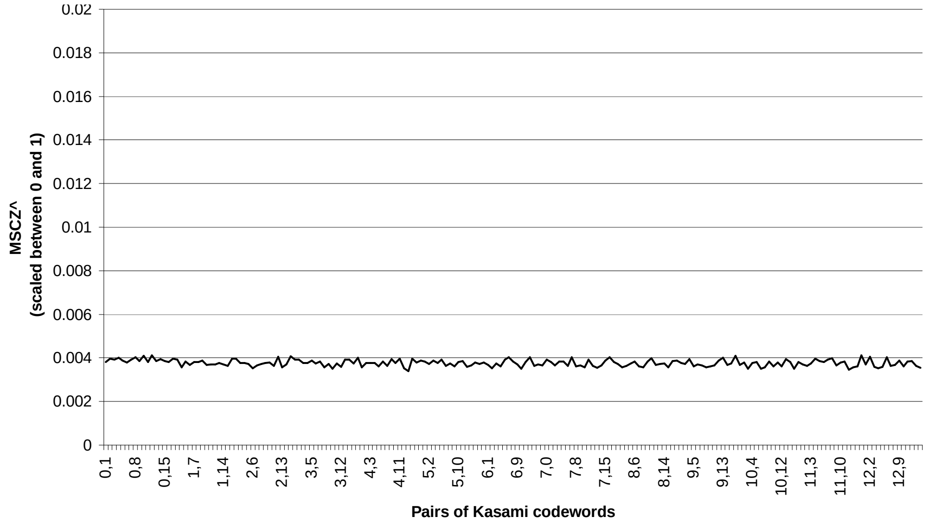


Figure 7.3: *Graph showing* $MSC\hat{Z}s$ *of the Small Kasami set of length 255.* $MSC\hat{Z}$ *experiences little variation compared to the* $MSC\hat{Z}s$ *of 2 sets of LCS codewords as shown in figure 7.2.*

7.2 Assignment of a Small Kasami set

The 458 transmitter problem will now be assigned using a Small Kasami set. The set has 16 codewords of length 255. Too few codewords are available for this problem, as a minimum of 30 codewords are required. However, a Small Kasami set will still be assigned to confirm this claim. The set of Kasami sequences used was constructed using the primitive polynomial 435 (in octal) or $x^8 + x^4 + x^3 + x^2 + 1$. Average $MSC\hat{Z}$ between a pair of codewords is 3.88×10^{-3} which is close to $\Phi_{x,x}$ required. As can be seen from figure 7.3, $MSC\hat{Z}$ for Small Kasami set experiences little variation from pair to pair. Optimising interference due to $MSC\hat{Z}$ is then not a critical issue.

7.2.1 Assignment of a Small Kasami set using $MSC\hat{Z}$

The algorithm employed here is a tabu search algorithm. At $13dB$, a random initial assignment started with an optimisation cost of $663,262$ (using $\psi_{x,x} = 1$)

and a total coverage of 29.9813%. The corresponding true cost with $\psi_{x,x} = MSA\hat{Z}_{x,x}$ is 0. The σ_{min} obtained is $0dB$ which is less than σ' and so spreading codeword re-use is unsatisfactory. The algorithm finished with an optimisation cost of 64,072.8 (with $\psi_{x,x} = 1$) and 62.729% total coverage. On average 72.32% of this cost is due to co-code interference ($ACC\%$). The corresponding true cost is zero but the σ_{min} obtained is $8dB$ which is less than σ'. This demonstrates that 16 Kasami sequences of length 255 are too few for a satisfactory codeword re-use in this optimisation problem.

7.3 Assignment of LS codes

LS code construction requires a careful consideration of the network specification. The specification for this problem is again analysed below:

1. The length of the spreading code must be close to 255.
2. $-15 \leq \tau \leq 15$.
3. A minimum of 30 codewords are required for satisfactory codeword re-use.

A construction by Stańczak, Boche and Haardt in [78] and described in section 3.9 will first be used to construct a suitable LS code. To achieve a $\tau_{\max}$ of 16, a Golay pair of length $\mathcal{N} = 16$ is constructed. Consider a Golay pair $C_8(Z) = 1 + z + z^2 - z^3 + z^4 + z^5 - z^6 + z^7$ and $S_8(Z) = 1 + z + z^2 - z^3 - z^4 - z^5 + z^6 - z^7$ of length 8. A Golay pair of length 16 is constructed as

$$\begin{aligned} C_{16_0}(Z) &= C_8(Z) + Z^8 S_8(Z), \\ S_{16_0}(Z) &= C_8(Z) - Z^8 S_8(Z) \end{aligned}$$

with mate $C_{16_1}(Z) = Z^{16} S_{16_0}(Z^{-1}), S_{16_1}(Z) = -Z^{16} C_{16_0}(Z^{-1})$. To achieve the required length an 8×8 Hadamard matrix is used to construct LS codewords each of length 286 but only 16 codewords can then be constructed. These sequences are orthogonal sequences within the zero correlation zone $-15 \leq \tau \leq 15$ and so both $MSC\hat{Z}_{x,y}$ and $MSA\hat{Z}_{x,x}$ are equal to zero. It is important to note that with ternary codewords $\theta_{x,x}(0) \neq N$; where $\theta_{x,x}(0)$ is the peak autocorrelation and N is the length of code. $\theta_{x,x}(0)$ is the number of non-zero elements of a codeword with elements in $\{-1, 0, +1\}$. The normalised mean-square measures $MSC\hat{Z}$ and $MSA\hat{Z}_{x,x}$ in equations (5.33) and (5.34) for ternary codes become:

$$MSC\hat{Z}_{x,y} = \frac{1}{\theta_{x,x}(0)^2 (2\tau_{max} - 1)} \sum_{\tau=1-N}^{N-1} | \hat{Z}_{x,y}(\tau) |^2 \qquad (7.1)$$

and

$$MSA\hat{Z}_{x,x} = \frac{1}{\theta_{x,x}(0)^2(2\tau_{max}-2)} \sum_{\tau=1-N,\tau\neq 0}^{N-1} \mid \hat{Z}_{x,x}(\tau) \mid^2 \quad (7.2)$$

This assignment problem is a straightforward graph colouring problem. The tabu search algorithm is used to solve this problem. At an SIR threshold of $13dB$, a random initial assignment using $\psi_{x,x} = 1$ started with an optimisation cost of 339,200 and 41.9813% coverage. The true cost with $\psi_{x,x} = MSA\hat{Z}_{x,x}$ is zero but $\sigma_{\min}$ obtained is $0dB$ which is less than σ'. Thus, re-use of LS codes by this assignment is unsatisfactory. The algorithm then reduces the cost with $\psi_{x,x} = 1$ to 25,603.1 and coverage of 77.9065%. $\sigma_{\min}$ achieved is $7dB$ which is less than σ'. The corresponding true cost is zero with 100% coverage. This assignment is unsatisfactory as the $\sigma_{\min} = 7dB$ achieved is less than $\sigma' = 14dB$ required. This demonstrates that even though the code has perfect zero correlation values, it is again hopeless to assign too few codewords to a large numbers of transmitters.

To increase the number of LS codewords, the approach presented in [35] and described in section 3.10 is used. Internal padding lengths are introduced. Three LS codes, each of 32 codewords and length 328 were constructed with transitions (Tr) and non transitions (NTr) as follows:

- $LS1 : (Tr, NTr) = (12, 3);\ \pi = 010100101001010$0.
- $LS2 : (Tr, NTr) = (15, 0);\ \pi = 0101010101010101$.
- $LS3 : (Tr, NTr) = (8, 7);\ \pi = 0110011001100110$.

with total padding lengths of 21. Average $MSC\hat{Z}$ for each of LS1, LS2 and LS3 is 1.89×10^{-3}.

7.3.1 Assignment of LS codes using $MSC\hat{Z}$

For comparison purposes, the same approach used for the LCS codes and the Small Kasami set is maintained. It is clear from figures 7.4, 7.5 and 7.6 that $MSC\hat{Z}$ differs for some distinct pairs of LS codewords. The algorithms then have to cope with optimising interference due to non-zero $MSC\hat{Z}$ interference and interference due to codeword re-use.

Results obtained by simulated annealing will first be examined using LS2. At an SIR threshold of $14dB$, a random initial assignment started with an optimisation cost of $318,418$ (with $\psi_{x,x} = 1$) and a 59.9626% coverage. Average percentages of interference from co-code and inter-code interferences are 89.3181% and 10.6819%

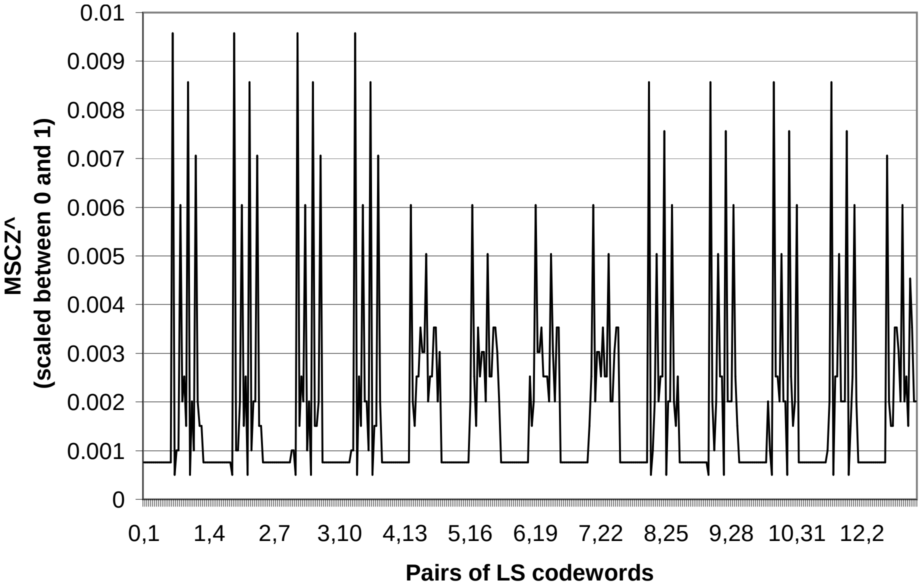


Figure 7.4: *Graph showing how $MSC\hat{Z}$ differs from pair to pair for some distinct pairs of LS codewords using* $(Tr, NTr) = (12, 3)$.

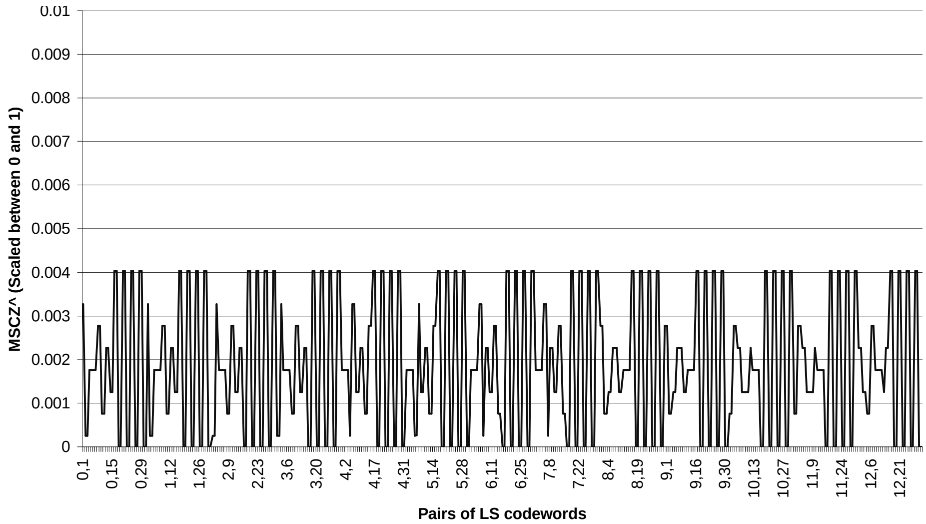


Figure 7.5: *Graph showing how $MSC\hat{Z}$ differs from pair to pair for some distinct pairs of LS codewords using* $(Tr, NTr) = (8, 7)$.

respectively. The corresponding true cost (using $\psi_{x,x} = MSA\hat{Z}_{x,x}$) is also 0 but the $\sigma_{\min}$ is $0dB$ which is less than σ'. This indicates that re-use of spreading codes by this initial random assignment is unsatisfactory. Simulated annealing then finished with a zero optimisation cost (with $\psi_{x,x} = 1$) and 100% coverage. The corresponding true cost (using $\psi_{x,x} = MSA\hat{Z}_{x,x}$) is 0 and the $\sigma_{\min}$ achieved is $15dB$ which is greater than σ'. Thus a satisfactory spreading codeword re-use is

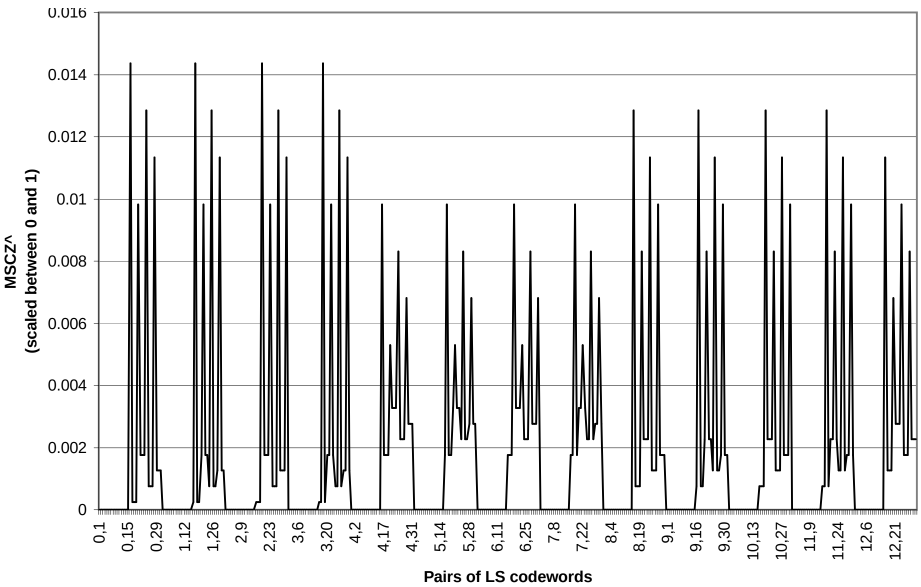


Figure 7.6: *Graph showing how $MSC\hat{Z}$ differs from pair to pair for some distinct pairs of LS codewords using* $(Tr, NTr) = (15, 0)$.

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	0
15	89.8909	99.7383	0.0926654	99.9073	0
16	1943.57	98.7664	0.0949261	99.9051	0
17	10434.8	96.0748	0.122077	99.8779	0

Table 7.8: *A table to show the unsatisfactory random initial assignments of* $LS2 : (Tr, NTr) = (15, 0)$. *All the* $\sigma_{\min}$ *obtained are unsatisfactory. Corresponding optimised results are presented in table 7.9.*

achieved with the assignment presented by simulated annealing. This result and those obtained at other thresholds are presented in tables 7.8 and 7.9. Comparing these two tables it is easy to see the effectiveness of the simulated annealing algorithm in optimising both interference from $MSC\hat{Z}$ and unsatisfactory re-use of codewords.

A similar approach is used to examine results obtained by the tabu search algorithm at an SIR threshold of $14dB$. A random initial assignment started with an optimisation cost of $378,181$ (using $\psi_{x,x} = 1$) resulting in a 56.5234% coverage. Much of the cost (90.6876% average interference) is due to codeword re-use. The corresponding true cost (with $\psi_{x,x} = MSA\hat{Z}_{x,x}$) is 0 and the $\sigma_{\min}$ is $0dB$ which is less than σ'. Again at this stage re-use of spreading codes is unsatisfactory.

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	15
15	0	100	0	0	16
16	0	100	0	0	15
17	0	100	0	0	14

Table 7.9: *A table to show the satisfactory assignments of $LS2 : (Tr, NTr) = (15, 0)$ achieved by the simulated annealing algorithm. These optimised results are obtained from the random assignments in table 7.8*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	0
15	10.0887	99.9252	0.754551	99.2454	0
16	285.15	99.6262	0.6978	99.3022	0
17	4249.69	97.6449	0.668044	99.332	0

Table 7.10: *This table shows the results obtained by random initial assignments of $LS1 : (Tr, NTr) = (12, 3)$ for the tabu search algorithm. Corresponding optimised results are presented in table 7.11.*

Tabu search then optimised this initial solution to give a zero cost (with $\psi_{x,x} = 1$) at 100% coverage. The corresponding true cost (i.e. with $\psi_{x,x} = MSA\hat{Z}_{x,x}$) is 0 and the $\sigma_{\min}$ achieved is $15dB$ which is greater than σ'. The assignment presented by tabu search at $14dB$ is then considered to achieve a satisfactory re-use of LS codewords. It is easy to deduce that both algorithms employed are able to work well on optimising the cost to give satisfactory spreading code assignments.

Tabu search will be further employed to assign LS1 and LS3. The results of the random initial assignments and optimised assignments of $LS1 : (Tr, NTr) = (12, 3)$ are presented in tables 7.10 and 7.11 respectively. It is clear that the tabu search algorithm is able to optimise the unsatisfactory initial assignments (presented in table 7.10) to achieve satisfactory optimised assignments (presented in table 7.11) with zero costs and $\sigma_{\min} \geq \sigma'$.

In the same way the tabu search algorithm is used to assign $LS3 : (Tr, NTr) = (8, 7)$. The results of the random initial assignments and optimised assignments are presented in tables 7.12 and 7.13 respectively. $LS3$ has the worst average correlation and this is evident in the true costs and coverages of the random initial assignments. In particular, at an SIR threshold of $17dB$, a random initial assignment gave a true cost of 179,424 with 84.2991% coverage. $\sigma_{\min}$ obtained

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	15
15	0	100	0	0	16
16	0	100	0	0	14
17	0	100	0	0	14

Table 7.11: *A table to show optimised results of* $LS1 : (Tr, NTr) = (12, 3)$ *with corresponding random initial results in table 7.10. All the results are satisfactory.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	4146.15	96.5234	77.6187	22.3813	0
15	16961.9	89.757	78.6894	21.3106	0
16	56864.4	89.7944	67.8197	32.1803	0
17	179424	84.2991	62.9761	37.0239	0

Table 7.12: *A table to show the unsatisfactory results of the random initial assignments of* $LS3 : (Tr, NTr) = (8, 7)$ *for the tabu search algorithm. Corresponding optimised results are presented in table 7.13.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	0	100	0	0	15
15	0	100	0	0	15
16	0	100	0	0	14
17	0	100	0	0	14

Table 7.13: *A table to show the satisfactory optimised results of* $LS3 : (Tr, NTr) = (8, 7)$ *using the tabu search algorithm.*

is $0dB$. The corresponding optimised true cost is zero with 100% coverage. The codeword re-use threshold $\sigma_{\min}$ is $14dB$ which is greater than σ'. The assignment is therefore a satisfactory assignment.

It is clear that the use of the simulated annealing and tabu search algorithms is important in assigning the modified LS codes. The most significant advantage is demonstrated with the use of the tabu search algorithm in the assignment of $LS3 : (Tr, NTr) = (8, 7)$.

7.4 Assignment of Kerdock derived codes

The construction of the Kerdock derived code used here is as described in section 3.6. The Kerdock derived code used is of length 254 (i.e. using $m = 7$) and has 128 codewords. It was constructed using characteristic polynomial $x^7 + 2x^4 + x + 3$ (in

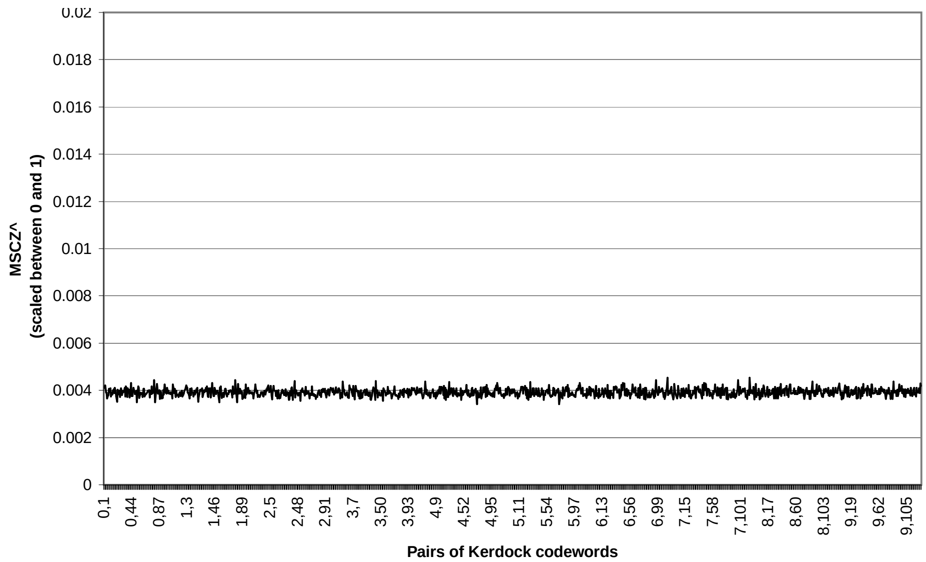


Figure 7.7: *Graph showing $MSC\hat{Z}$ for a Kerdock derived code of length 254. The variation in $MSC\hat{Z}$ is not as significant as with two sets of LCS codewords (in figure 7.2) or an LS code (in figures 7.4, 7.5 and 7.6).*

base 4) which corresponds to hexadecimal 4207 for degree 7 in the characteristic polynomial table presented in [35].

7.4.1 Assignment of Kerdock codes using $MSC\hat{Z}$

A graphical representation of the $MSC\hat{Z}$ of a Kerdock derived code is presented in figure 7.7. The figure reveals that the Kerdock derived code does not have significant variations in $MSC\hat{Z}$s. There are 128 codewords available and so codeword re-use is not a critical problem. Satisfactory codeword re-use can be achieved with a short algorithm run time (2 to 3 hours).

This assignment problem was solved using the tabu search algorithm and the simulated annealing algorithm. It is clear from figures 7.8 and 7.9 (with costs using $\psi_{x,x} = 1$) that there are no significant differences in the performances of the two methods. This may be because numerous codewords are available and so little work is required by the algorithm for a satisfactory codeword re-use. Assignments obtained using simulated annealing will be further examined here.

At $14dB$, a random initial assignment incurred an optimisation cost of 130,416 (using $\psi_{x,x} = 1$) with 80.972% total coverage. Average percentage of interference due to $MSC\hat{Z}$ and codeword re-use are 26.305% and 73.695% respectively. The corresponding true cost (using $\psi_{x,x} = MSA\hat{Z}_{x,x}$) is 289.937 with total coverage of 98.7664%. Average percentage of interference contribution

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	285.212	98.729	0.501994	99.498	0
15	3625.88	96.2991	0.9969	99.0031	0
16	23860	86.4673	1.01637	98.9836	0
17	123725	69.757	0.665775	99.3342	0

Table 7.14: *This table shows the results obtained from random initial assignments of the Kerdock code. All the assignments are unsatisfactory as all* $\sigma_{\min}$ *obtained are less than* $\sigma' = 14dB$ *required.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	186.715	98.9159	0	100	17
15	2990.11	96.8224	0.0003	99.9997	16
16	20694	87.0654	0.00356676	99.9964	20
17	113751	70.729	0.00752606	99.9925	21

Table 7.15: *This table shows the results of the optimised assignments (with corresponding initial assignments in table 7.14) of a Kerdock code of length 254. The values of* $\sigma_{\min}$ *achieved are satisfactory but there are no significant differences in the costs of random assignments in table 7.14 and the optimised costs achieved here.*

due to spreading codeword re-use and $MSC\hat{Z}$ are 0.462761% and 99.5372% respectively. The $\sigma_{\min}$ obtained at this stage is $0dB$ which is less than σ' and so minimum spreading codeword re-use obtained is unsatisfactory.

The system optimized the problem to reduce the cost (with $\psi_{x,x} = 1$) to 186.049 and increased the coverage to 98.4299%. Average percentage of interference due to codeword re-use and $MSC\hat{Z}$ are 10.5331% and 89.4669% respectively. The corresponding true cost (i.e. with $\psi_{x,x} = MSA\hat{Z}_{x,x}$) is 186.045 and total coverage is 98.9159%. This true cost is as a result of interference from $MSC\hat{Z}$ only. The $\sigma_{\min}$ achieved is $17dB$ which is greater than σ' and so a satisfactory spreading codeword re-use is achieved by this assignment at an SIR threshold of $14dB$. Results obtained at other thresholds are presented in tables 7.14 and 7.15. Comparing the results of the random initial assignments in table 7.14 with the optimised results in table 7.15, it is easy to see that apart from satisfactory $\sigma_{\min}$ achieved there are no significant differences in the costs obtained. This is due to the small variations experienced in $MSC\hat{Z}$. Again, little advantage is gained from careful code assignment when variations in correlations are small and codeword re-use is not a critical issue.

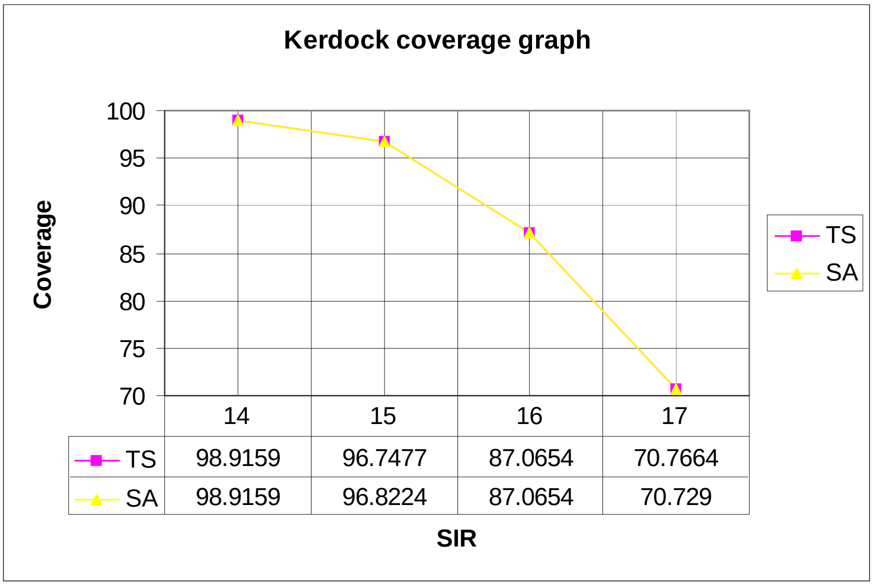


Figure 7.8: *Coverage graph obtained for the Kerdock code using tabu search and simulated annealing.*

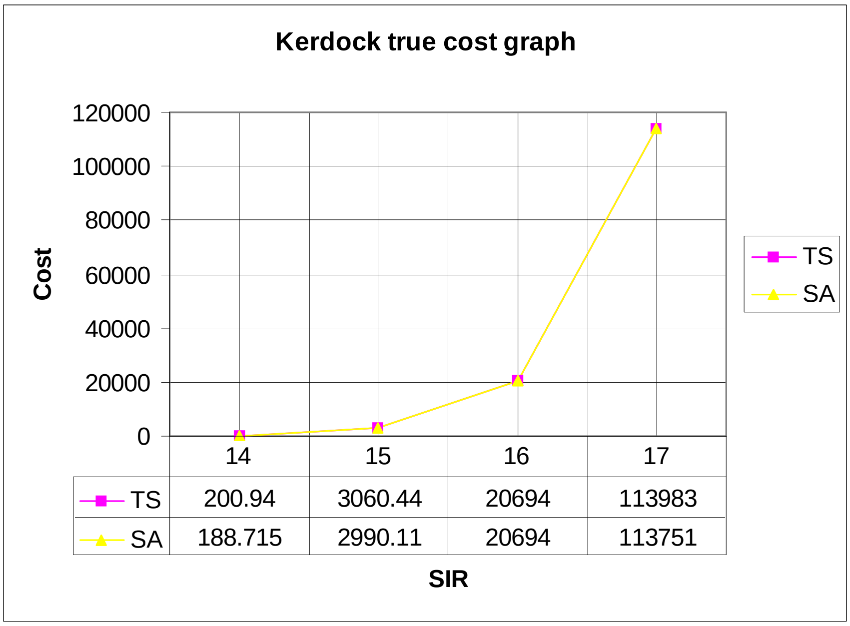


Figure 7.9: *Cost graph (obtained for the Kerdock code using tabu search and simulated annealing.*

7.5 Assignment of a Large Kasami set

A Large Kasami set of length 255 and 4,111 codewords is constructed using primitive polynomial 435 (in octal) or $x^8 + x^4 + x^3 + x^2 + 1$. Due to the large

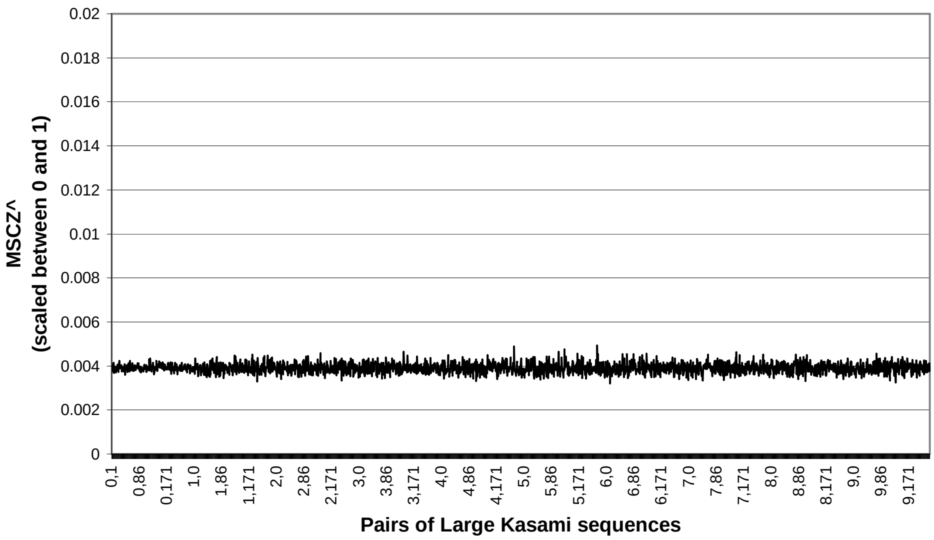


Figure 7.10: *This graph shows $MSC\hat{Z}$s of a Large Kasami set of length 255. The variation in the $MSC\hat{Z}$ is not as significant as with two sets of LCS codewords (in figure 7.2) or in an LS code (in figures 7.4, 7.5 and 7.6).*

memory and time required to compute $MSC\hat{Z}$ values for the 4,111 codewords, only a subset consisting of the first 1000 codewords is made available to the algorithm. In figure 7.10, there is only very small variations in the $MSC\hat{Z}_{x,y}$ between pairs of codewords in the Large Kasami set. This is not a critical code assignment problem. However, the algorithms can be used to take advantage of these variations to reduce cost.

The tabu search and the simulated annealing algorithms are adjusted to cope with the number of codewords available. When a request for a random codeword is made, the algorithms are adjusted to only choose a distinct codeword that is not in use by the current assignment thereby prohibiting re-use of codewords.

7.5.1 Assignment of a Large Kasami sequence using $MSC\hat{Z}$

Note that there are sufficient codewords and so $\psi_{x,x}$ is not used by the algorithm in the cost function. Results obtained by simulated annealing will first be presented. At an SIR threshold of $14dB$, a random initial assignment choosing 458 distinct codewords (i.e. choosing from the 1,000 codewords made available to the algorithm) gave a total cost of 302.65 and 98.6168% coverage. Simulated annealing then optimised this initial cost to finish with a cost of 86.3536 and 99.0654% coverage.

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	302.65	98.6168	0	100	no codeword re-use
15	3737.79	96.2243	0	100	no codeword re-use
16	23746.5	86.3925	0	100	no codeword re-use
17	123743	69.5327	0	100	no codeword re-use

Table 7.16: *This table shows the results of the random initial assignments of a Large Kasami set of length 255.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
14	86.3536	99.1025	0	100	no codeword re-use
15	2211.63	96.8972	0	100	no codeword re-use
16	16315.6	88.7477	0	100	no codeword re-use
17	93943.1	73.0841	0	100	no codeword re-use

Table 7.17: *This table shows the optimised assignments (with corresponding initial assignment in figure 7.16) of a Large Kasami set.*

The same approach can be followed to analyse results obtained by the tabu search algorithm. At an SIR threshold of $14dB$, a random initial assignment gave a total cost of 285.161 and 98.6916% coverage. The tabu search algorithm then optimised the initial cost to give 86.3536 and 99.1025% coverage. These results and those obtained at other SIR thresholds are presented in tables 7.16 and 7.17. It is easy to see that the simulated annealing algorithm is able to optimise the small variations in the $MSC\hat{Z}$s to reduce cost.

A tabu search algorithm can also be used to assign the Large Kasami set. The results obtained are compared with those of the simulated annealing algorithm are presented in figures 7.11 and 7.12. As it is with the assignment of a Kerdock derived code, there is no significant difference in the performances of the two algorithms. It is equally easy to see that both algorithms are able to take advantage of the small variations in the $MSC\hat{Z}$ correlations of the Large Kasami set to minimise the cost.

7.6 Comparison of the optimised spreading code assignments obtained by using $MSC\hat{Z}$

In this section, all the assignment methods with satisfactory codeword re-use presented in sections 7.1.1, 7.3.1, 7.4.1 and 7.5.1 will be compared. For the spreading codes considered almost the same performances are achieved by both the tabu

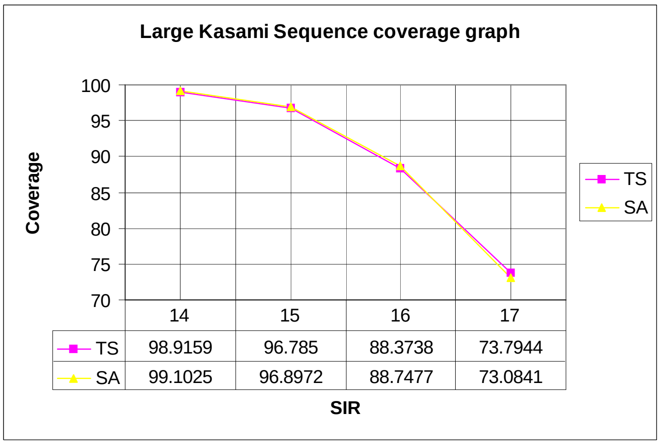


Figure 7.11: *Coverage graphs obtained by the tabu search and simulated annealing algorithms for a Large Kasami set.*

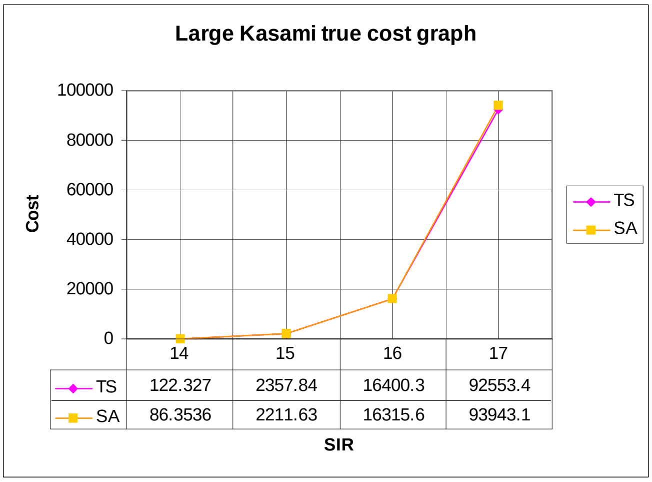
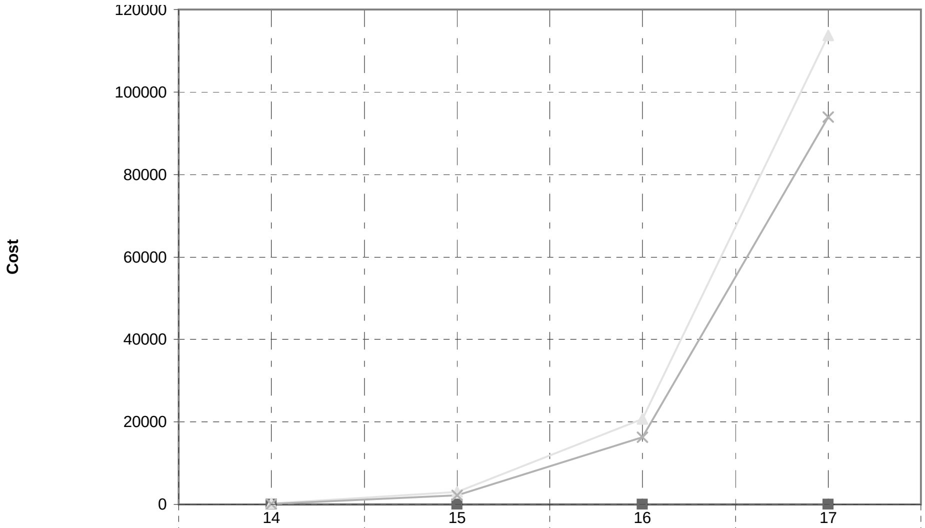


Figure 7.12: *Cost graphs obtained by the tabu search and simulated annealing algorithms for a Large Kasami set.*

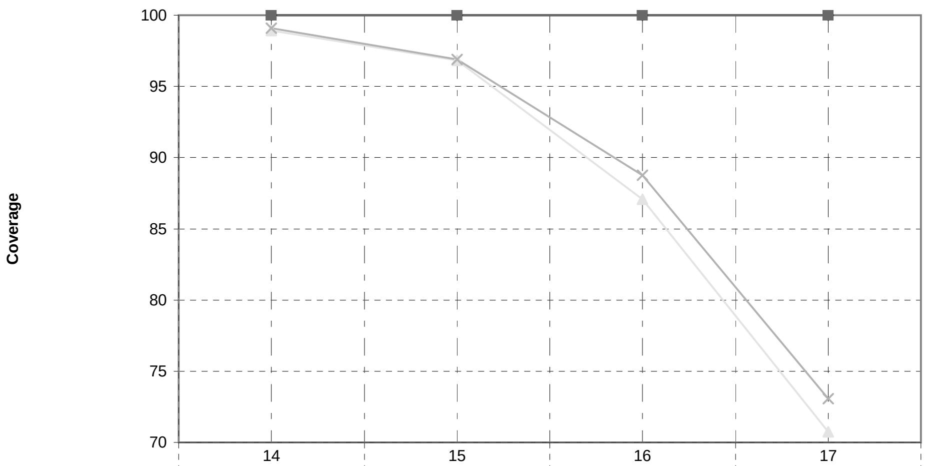
search algorithm and the simulated annealing algorithm. Figures 7.13 and 7.14 present the optimised true cost with the corresponding coverages achieved by the tabu search algorithm (for LCS and LS codes) and the simulated annealing (for the Kerdock derived code and the Large Kasami set). It is easy to see that codes for quasi-synchronous operation with fewer codewords (i.e. LCS and LS codes) present better solutions at all thresholds than codes designed for operating asynchronously (i.e. Kerdock and Large Kasami codes) which usually have adequate numbers of codewords. The effect of the algorithms in mitigating the resulting interferences due to high non-zero correlations in the modified LS and LCS codes has been clearly demonstrated.



SIR	14	15	16	17
LCS (2 sets)	0	0	0	0
LS (T15NT0)	0	0	0	0
Kerdock	186.045	2990.11	20694	113751
Large Kasami	86.3536	2211.63	16315.6	93943.1

SIR

Figure 7.13: *Cost graphs obtained for all the codes assigned.*



SIR	14	15	16	17
LCS (2 sets)	100	100	100	100
LS (T15NT0)	100	100	100	100
Kerdock	98.9159	96.8224	87.0654	70.729
Large Kasami	99.1025	96.8972	88.7477	73.0841

SIR

Figure 7.14: *Coverage graphs obtained for all the codes assigned.*

Chapter 8

A 458 Transmitter Problem - Peak Planning

In this chapter the 458 transmitter problem described in chapter 7 is solved using peak planning ($PSC\hat{Z}$). For peak planning, in order to impose sufficient penalty on unsatisfactory codeword re-use, $\psi_{x,x} = 4$ is used when a codeword is re-used such that $\sigma_{\min} > \sigma'$ is not satisfied; $\psi_{x,x} = PSA\hat{Z}_{x,x}$ is used otherwise. Further reasons to justify this choice are in section 8.1. The same software specification as described in chapter 7 is used. Two sets of LCS codewords of length 255 (described in section 7.1.1), the LS code of length 328 (described in section 7.3.1), the Kerdock derived code of length 254 (described in section 7.4.1) and the Large set of Kasami sequences of length 255 (described in section 7.5.1) will be assigned using $PSC\hat{Z}$. $PSC\hat{Z}$ itself is calculated using the method described in section 5.2.4.

8.1 Remarks on using peak square of correlation $PSC\hat{Z}$

During an initial experiment it was discovered that with codes for which $PSC\hat{Z} >> MSC\hat{Z}$ and $PSA\hat{Z}_{x,x} >> MSA\hat{Z}_{x,x}$, a choice of $\psi_{x,x} = 1$ does not impose sufficient penalty when a code is unsatisfactorily re-used. A good choice of $\psi_{x,x}$ is then necessary for the algorithm to generate an assignment with satisfactory codeword re-use. In particular, to generate a near equivalent result to a particular result obtained using the mean-square of $\hat{Z}$ method at a particular threshold, it is necessary to determine $\psi_{x,x}$ from the peak correlation and mean correlation. Let K be the number of codewords in a code class. The constant λ defined as:

$$\lambda = \left\lfloor \frac{1}{K^2} \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} \frac{MSC\hat{Z}_{x_i,x_i \neq y_j} + MSA\hat{Z}_{x_i,x_i=y_j}}{PSC\hat{Z}_{x,x\neq y} + PSA\hat{Z}_{x,x=y}} \right\rfloor , \tag{8.1}$$

will be used in order to obtain assignments (using the peak square of $\hat{Z}$ method) comparable to those obtained by the mean square of $\hat{Z}$ method. $\psi_{x,x} = \lambda$ is used during optimisation and $\psi_{x,x} = PSA\hat{Z}_{x,x}$ is used for an evaluation of the true cost of an assignment.

The 458 transmitter problem is solved using peak of $\hat{Z}$ planning. The indicated value of λ for the LCS code, the LS code, the Kerdock code and the Large Kasami sequences are 6, 4, 8 and 9 respectively. This is only important when codeword re-use is a critical problem. For this reason, $\lambda = 8$ for the Kerdock code and $\lambda = 9$ for the Large set of Kasami are not relevant to this problem as codeword re-use is not a critical issue. The choice of $\lambda = 4$ will be used for all the assignments presented here.

The average $PSC\hat{Z}$ for the LCS code, the LS code, the Kerdock code and the Large set of Kasami sequences are 1.56×10^{-2}, 1.17×10^{-2}, 3.25×10^{-2} and 3.53×10^{-2} respectively. All these averages are far more than $\Phi_{x,y} = 3.08 \times 10^{-3}$ at $14dB$ and so the peak correlations are too high for satisfactory assignment at an SIR threshold of $14dB$. In particular, if equation (5.45) is rearranged as

$$\sigma \leq \frac{1}{IP_{\max}\Phi_{x,y}}$$

and set $\Phi_{x,y}$ to each of the averages (recall that $IP_{\max}$ for this problem is 12.9127) the SIR threshold σ required will be $7dB$, $8dB$, $4dB$ and $3dB$ respectively. However, for comparison purposes the problem will be solved using $7dB$, $8dB$, $9dB$ and $10dB$ SIR thresholds.

8.2 Assignment of an LCS code using $PSC\hat{Z}$

Two sets of LCS codewords are used. The construction is as described in section 7.1.1. The distribution of the $PSC\hat{Z}$ is presented in figure 8.1. It is easy to see that the correlations vary from pair to pair of the combined sets of LCS codewords. The tabu search algorithm is used to solve this problem. The random initial assignments and optimised assignments are presented in tables 8.1 and 8.2. Comparing these two tables, it is easy to see the effectiveness of the tabu search algorithm in mitigating interference from high correlations (with high $AIC\%$ in the initial solutions of table 8.1). The algorithm is also able to achieve satisfactory codeword re-use. In particular, at SIR threshold of $10dB$ the true cost of the random initial assignment is 11,302.8 with 83.3271% total coverage. The

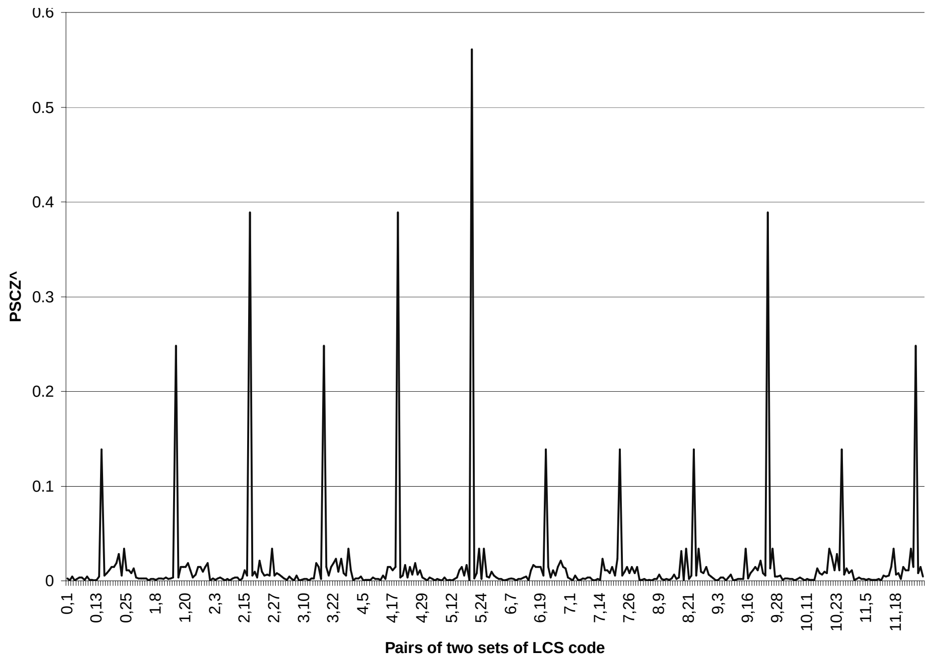


Figure 8.1: *This graph shows the $PSC\hat{Z}$ for the two sets of LCS codewords. It is easy to see that the $PSC\hat{Z}$ varies from pair to pair of LCS codewords.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
7	504.331	95.514	0.0421019	99.9579	0
8	2196.06	89.4579	0.0758289	99.9242	0
9	5114.23	87.4019	0.153106	99.8469	0
10	11302.8	83.3271	0.16237	99.8376	0

Table 8.1: *This table shows the random initial assignments using $PSC\hat{Z}$ of the two sets of LCS codewords of length 255. The assignments are unsatisfactory as $\sigma_{\min}$ is less than $\sigma' = 14dB$. The corresponding optimised assignment is presented in table 8.2.*

tabu search algorithm optimised this initial assignment to achieve an optimised assignment with a zero cost, 100% coverage and satisfactory codeword re-use $\sigma_{\min} = 14$.

8.3 Assignment of an LS code using $PSC\hat{Z}$

For this assignment $LS2 : (\mathrm{Tr}, \mathrm{NTr}) = (15, 0)$ is used to construct 32 LS codewords each of length 328. Figure 8.2 presents the $PSC\hat{Z}$ between pairs of LS codewords; it is easy to see that the assignment of the LS code using $PSC\hat{Z}$ is similar to that using $MSC\hat{Z}$ in section 7.3.1. $PSC\hat{Z}$ varies significantly from pair

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
7	0	100	0	0	14
8	0	100	0	0	14
9	0	100	0	0	14
10	0	100	0	0	14

Table 8.2: *This table shows the corresponding optimised results to the random initial assignments in table 8.1. The algorithm used is the tabu search algorithm and the measure used is $PSC\hat{Z}$. All the results are satisfactory as $\sigma_{\min} = \sigma' = 14dB$ and zero costs and 100% coverages are achieved.*

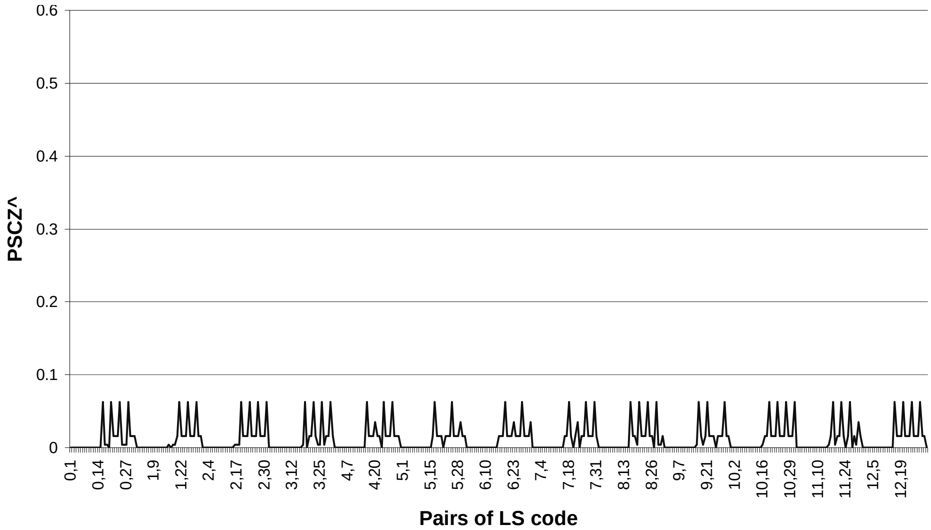


Figure 8.2: *This graph shows the $PSC\hat{Z}$ of $LS2 : (\mathrm{Tr}, \mathrm{NTr}) = (15, 0)$. It is easy to see that the $PSC\hat{Z}$ vary from pair to pair of LS codewords.*

to pair of the LS codewords. Tables 8.3 and 8.4 present the results of the random initial and optimised assignments respectively. It follows from these tables that the use of the tabu search algorithm resulted in improved system performances in this problem.

8.4 Assignment of a Kerdock derived code using $PSC\hat{Z}$

The same Kerdock code used for mean planning is used. The Kerdock code has 128 codewords each of length 254. The values of $PSC\hat{Z}$s between pairs of Kerdock codewords are presented in figure 8.3. The degree of variation in $PSC\hat{Z}$ of the Kerdock code is small compared to the LS code and the combination of two sets of LCS codewords. However, as demonstrated by the mean planning method

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
7	2.28588	99.7757	0	100	0
8	30.9257	98.8411	0	100	0
9	258.586	95.6636	0	100	0
10	1233.25	91.4393	0	100	0

Table 8.3: *This table shows the random initial assignments of an LS code to 485 transmitters using* $PSC\hat{Z}$*. All the assignments are unsatisfactory as the values of* $\sigma_{\min}$ *obtained are less than* $\sigma' = 14dB$*. Corresponding optimised results are presented in table 8.4.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
7	0	100	0	0	14
8	0	100	0	0	15
9	0	100	0	0	14
10	0	100	0	0	14

Table 8.4: *This table shows the corresponding optimised results to the initial assignments of LS code in table 8.3. The assignments achieved are satisfactory as the values of* $\sigma_{\min}$ *obtained are greater than or equal to* $\sigma' = 14dB$*.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
7	496.301	83.7757	0.765081	99.2349	0
8	2403.2	65.3084	1.31425	98.6857	0
9	9506.76	46.1682	0.850374	99.1496	0
10	29696.1	26.6916	0.594729	99.4053	0

Table 8.5: *This table shows random initial solutions obtained using the Kerdock derived code of length 254. The measures used are* $PSC\hat{Z}$ *and* $PSA\hat{Z}$*. All the solutions are unsatisfactory as the values of* $\sigma_{\min}$ *obtained are less than the value* σ' *required.*

(using $MSC\hat{Z}$) the optimisation algorithms adapted for code assignment can be used to mitigate the resulting interference due to small variation in $PSC\hat{Z}$. The simulated annealing algorithm is used to solve this assignment problem. Tables 8.5 and 8.6 present the random initial solutions and the corresponding optimised solutions respectively. The optimised assignments result in costs that are less than half of the corresponding initial costs. A large part of these costs result from high $PSC\hat{Z}$ (i.e. $AIC\%$). The algorithm is also able to achieve satisfactory codeword re-use. The simulated annealing algorithm is able to work on the small variation in $PSC\hat{Z}$ of the Kerdock code to obtain better system performance.

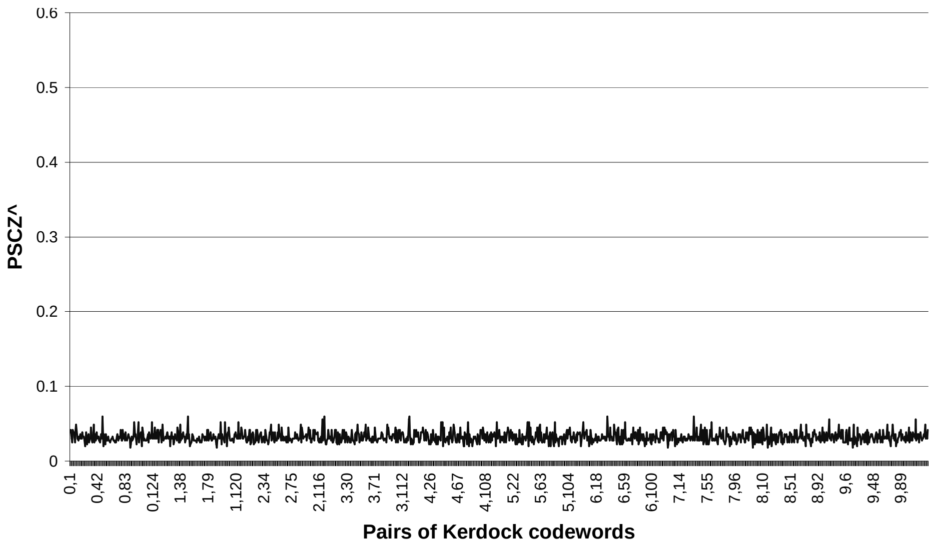


Figure 8.3: *This graph shows the* $PSC\hat{Z}$ *of the Kerdock derived code. Again the variations in* $PSC\hat{Z}$ *are not very significant.*

σ (dB)	true cost	coverage (%)	$ACC\%$	$AIC\%$	$\sigma_{\min}$ (dB)
7	192.079	90.6542	0.137038	99.863	14
8	1139.62	78.3551	0.012458	99.9875	18
9	5348.02	58.243	0.0117612	99.9882	23
10	18059.5	39.514	0.0525719	99.9474	14

Table 8.6: *This shows the optimised solutions of the Kerdock code assignment using* $PSC\hat{Z}$. *All the assignments achieved satisfactory codeword re-use.*

8.5 Assignment of a Large set of Kasami sequences using $PSC\hat{Z}$

The Large set of Kasami sequences used is as described in section 7.5. There are 4,111 codewords available but only the first 1,000 codewords are made available to the algorithm. Again, there are sufficient codewords and so the algorithm can not re-use a codeword. Figure 8.4 presents the $PSC\hat{Z}$ of a sample consisting of the first 10 codewords against the first 255 codewords. The degree of variation is small and so little work is required by the algorithm to reduce the resulting interference. The results of random initial assignments and the optimised assignments are presented in tables 8.7 and 8.8 respectively. The solution is similar to when a Kerdock code is used. The optimised costs obtained are just a little less than half of the initial costs. This again confirms the effectiveness of the algorithms in minimising the resulting interference due to small variations in $PSC\hat{Z}$ values.

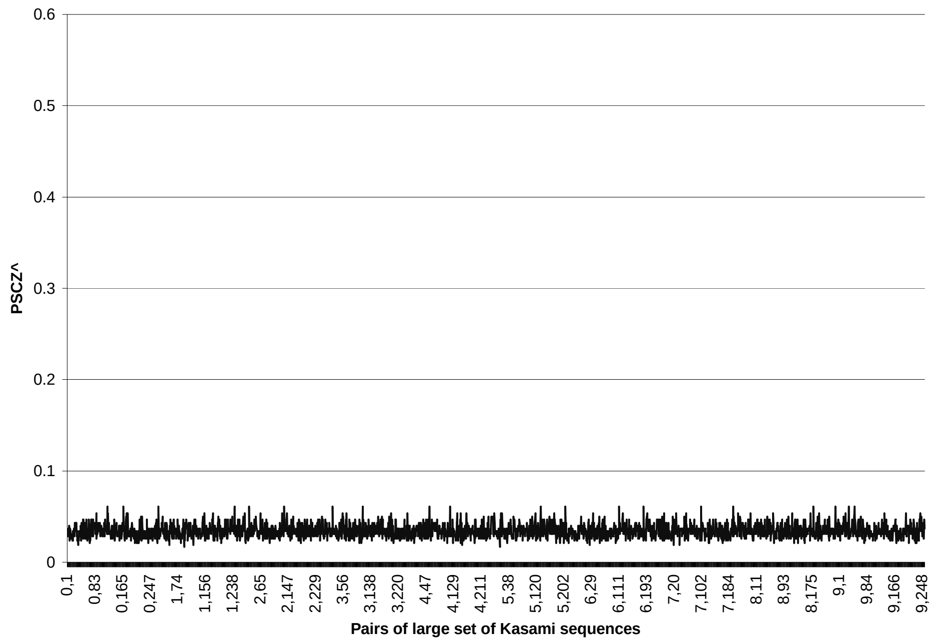


Figure 8.4: *This graph shows the $PSC\hat{Z}$ of the Large set of Kasami sequences using a sample of the first ten codewords against the rest of the first 255 codewords. It is easy to see that the degree of variation of $PSC\hat{Z}$ values is small.*

σ (dB)	true cost	coverage (%)	ACC%	AIC%	$\sigma_{\min}$ (dB)
7	797.27	77.2336	0	100	no codeword re-use
8	3557.52	58.5421	0	100	no codeword re-use
9	12432.2	37.8318	0	100	no codeword re-use
10	36287.9	20.4112	0	100	no codeword re-use

Table 8.7: *This table shows the results of the random initial assignments of the Large set of Kasami sequences using $PSC\hat{Z}$. The corresponding optimised results are presented in table 8.8.*

σ (dB)	true cost	coverage (%)	ACC%	AIC%	$\sigma_{\min}$ (dB)
7	207.297	88.3738	0	100	no codeword re-use
8	1249.12	75.1402	0	100	no codeword re-use
9	5412.89	57.1589	0	100	no codeword re-use
10	18765.5	38.5047	0	100	no codeword re-use

Table 8.8: *This table shows the optimised solutions of the assignments of the Large set of Kasami sequences using $PSC\hat{Z}$.*

8.6 Comparison of the optimised solutions achieved using $PSC\hat{Z}$

In this section the optimised assignments of the two sets of LCS codewords, the LS code, the Kerdock code and the Large set of Kasami sequences will

code class	$\max\{PSC\hat{Z}_{x,y}\}$	Average $PSC\hat{Z}$	$\min\{PSC\hat{Z}\}$
LCS	5.6103×10^{-1}	1.56×10^{-2}	3.84×10^{-4}
LS	6.25×10^{-2}	1.17×10^{-2}	0
Kerdock	7.596×10^{-2}	3.25×10^{-2}	1.395×10^{-2}
Large Kasami	7.3218×10^{-2}	3.53×10^{-2}	1.478×10^{-2}

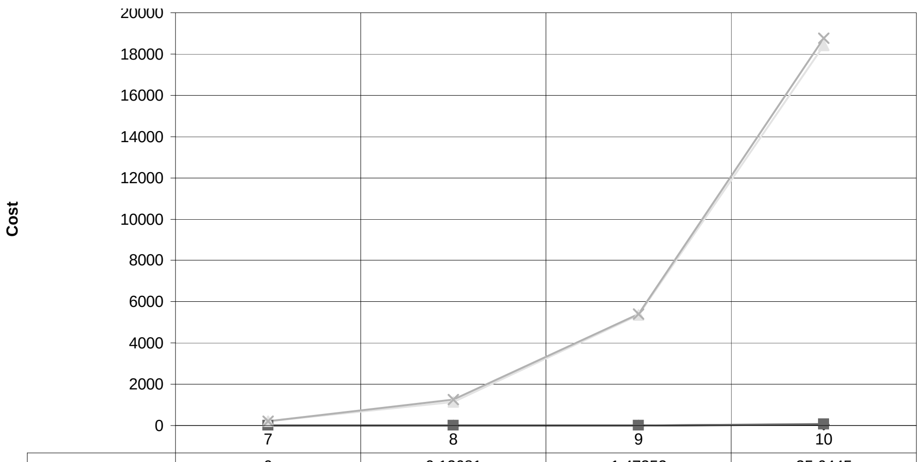
Table 8.9: *This table summarises the $PSC\hat{Z}$ of the LCS code, the LS code, the Kerdock code and the Large sets of Kasami sequences. It is easy to see that the LCS code has the worst peak $PSC\hat{Z}$ but its average and minimum $PSC\hat{Z}$ are better than that of the Kerdock code and the Large set of Kasami sequences.*

be compared. Consider the $PSC\hat{Z}$ graphs of these codes which are presented in figures 8.1, 8.2, 8.3 and 8.4 respectively. From these graphs and table 8.9 it is clear that the combination of two sets of LCS codewords resulted in the worst $PSC\hat{Z}$ value. However, these worst correlations only occur for very small numbers of LCS codeword pairs. In particular, from table 8.10 only 3.3% of the LCS correlations are higher than the $\max\{PSC\hat{Z}\}$ of the Kerdock code and the Large set of Kasami sequences. 78.89% and 87.33% of the LCS correlations are less than the $\min\{PSC\hat{Z}\}$ of the Kerdock code and the Large set of Kasami sequences respectively. As shown in figure 8.1, the optimisation algorithm adapted to spreading code assignment clearly mitigated the resulting interferences due to these worst correlations.

Figures 8.5 and 8.6 present the costs and coverages of all the optimised results achieved. It easy to see that the LCS code and the LS code (for quasi-synchronous operations) performed better that the Kerdock and the Large set of Kasami sequences (for asynchronous operations). The effectiveness of the algorithms and the approach for spreading code assignment is then apparent.

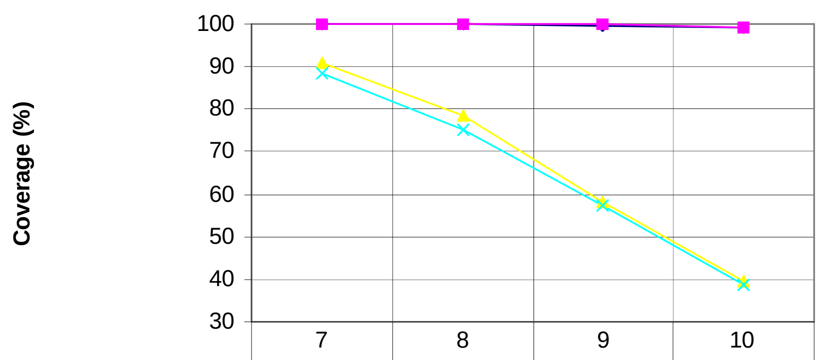
	Kerdock code	Large kasami sequences
% of $PSC\hat{Z}$ of the LCS code that are less than $\min\{PSC\hat{Z}\}$ for the other codes	78.89%	87.33 %
% of $PSC\hat{Z}$ of the LCS code that are greater than $\min\{PSC\hat{Z}\}$ but less than $\max\{PSC\hat{Z}\}$ for the other codes	17.88%	9.33%
% of $PSC\hat{Z}$ that are greater than $\max\{PSC\hat{Z}\}$ for the other codes	3.33 %	3.34%

Table 8.10: *This table compares the correlations of the LCS code with the Kerdock code and the Large set of Kasami sequences.*



	7	8	9	10
LCS	0	0.12681	1.47352	35.6445
LS	0	0	0.116272	61.266
kerdock	205.839	1139.62	5348.02	18413.9
Large kasami	207.297	1249.12	5412.89	18765.5

Figure 8.5: *True cost obtained using peak of $\hat{Z}$ planning with $\psi_{x,x} = 4$*



	7	8	9	10
LCS	100	99.9252	99.7757	99.2523
LS	100	100	99.8879	99.2897
kerdock	90.7664	78.3551	58.243	39.5888
Large kasami	88.3738	75.1402	57.1589	38.5047

SIR

Figure 8.6: *True coverage obtained using peak of* $\hat{Z}$ *planning* $\psi_{x,x} = 4$

Chapter 9

A 358 Transmitter Problem

9.1 Description of problem

This problem is an example of a cellular problem. The model represents a downlink problem of a cellular network with serving transmitters (base stations) at the cell centres. The hexagonal geometry is presented in figure 9.1. The vertices of the regular hexagons represent the reception points with serving transmitters at the cell centres. In this problem, there are 156 distinct receiver points. The coordinates of the cell centres (i.e. the serving transmitters) correspond to $x = 1000 \times (j-1) + 500 \times i,\ y = 500\sqrt{3} \times i$, with $i, j \in \{1, 2, \ldots, 9\}, 5 < i+j < 15$. Serving transmitters are, therefore, generated sequentially in order of the sequence given by $(1,5), (1,6), \ldots, (1,9), (2,4), (2,5), \ldots, (9,5)$. There are 61 cells with demands as follows:

$(6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 3; 3; 5; 5; 7; 6;$

$8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7)$.

Thus there are 2148 reception point instances. Again, a $d^{-3.5}$ law is used for propagation loss. When the system is operated quasi-synchronously and an LCS code or an LS code is used it is assumed that $-31 \leq \tau \leq 31$.

9.1.1 Analysis of the 358 transmitter problem

As with other problems considered in this work, the problem is modelled as a downlink problem. To allow comparison with the technique used in 3^{rd} generation mobile telephone systems, the value of τ is always taken to be zero within a cell. Interfering signals from other cells suffer different propagation time delays and are considered to be within the low correlation zone, $\{-\tau_{max}+1, \ldots, \tau_{max}-1\}$ (in the case of quasi-synchronous operation) or within any time delay $\mid \tau \mid \leq N$ (for asynchronous operation), where N is the code length.

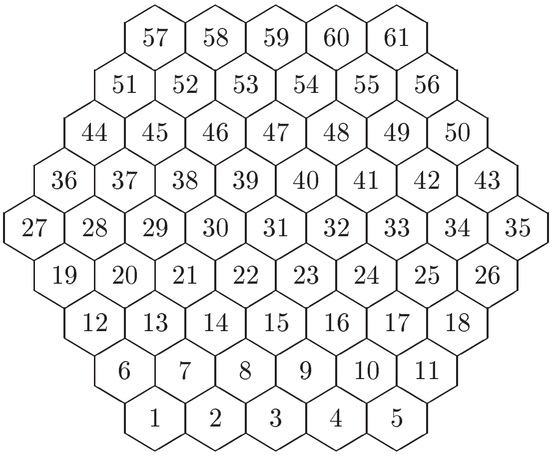


Figure 9.1: *The cellular geometry of HEX358, with 61 cells.*

An initial experiment as described in section 5.6 suggested that at least 60 codewords are required for satisfactory codeword re-use. This may be as a result of high demands within cells. The value of $IP_{\max}$ of equation (5.43) is 25.8166. From equation (5.45), for a satisfactory assignment at $\sigma = 14dB$, an average correlation (i.e $\psi_{x,x}$ and $\psi_{x,y}$) should be less than or equal to $\Phi_{x,y} = 1.54 \times 10^{-3}$. These two conditions will be considered in the selection of a spreading code class for this problem.

Next, the length of code required to achieve a correlation satisfying the bound $\Phi_{x,y} = 1.54 \times 10^{-3}$ is determined. A Large kasami set with codewords of length 255 is not suitable as the average of all $\psi_{x,y}$ is $3.92 \times 10^{-3} > \Phi_{x,y}$. The average of all $\psi_{x,y}$ of a Gold code of length 1023 with 1024 codewords is $9.77 \times 10^{-4} < \Phi_{x,y}$. For this reason a code length of about 1023 will be used.

Even though these three conditions are satisfied, assignment of the spreading code may still not be trivial, especially when codewords have to be re-used. Re-use of spreading codewords within a cell must be strictly avoided and the distance between transmitters assigned pairs of codewords with higher correlation than $\Phi_{x,y}$ must be maximised in order to reduce interference in the system.

9.1.2 Software specification

Unlike the 458 transmitter problem, two correlation files are made available to the software - the file containing $\psi_{x,y}$ at $\tau = 0$ (which is only used between codewords assigned to pairs of transmitters within the same cell) and the file containing $\psi_{x,y} = MSC\hat{Z}$ (which is used between codewords assigned to pairs of transmitters from different cells).

For consistency, the tabu search algorithm will be used to assign an LCS code or an LS code. Simulated annealing will be used when a Gold code is assigned. The tabu search algorithm has a neighbourhood size of $0.25 \times |T|$ where $|T|$ is the total number of transmitters, i.e $\lceil 0.25 \times 358 \rceil = 90$. A move to a new assignment is declared tabu if the move has been made over the last 143 i.e $\lfloor 0.4 \times |T| \rfloor$ moves. Tabu search is allowed to stop if no improved solution is obtained over 10^5 iterations. For the simulated annealing algorithm, the cooling schedule is $t := t \times 0.999999$ and a full neighbourhood is used.

Note that during the optimisation process, the algorithm uses $\psi_{x,x} = 1$ for any pair of transmitters assigned the same codeword for which σ' is not satisfied. $\psi_{x,x} = MSA\hat{Z}$ is used otherwise. The value of $\psi_{x,x} = MSA\hat{Z}$ is used during evaluation of an assignment. Reasons for this have been explained in sections 5.40 and 5.5. The assignment using $PSC\hat{Z}$ will not be attempted. This is because the performance of spreading codes using $MSC\hat{Z}$ in chapter 7 is similar to that using $PSC\hat{Z}$ in chapter 8. In the 458 transmitter problem it was clearly shown that the Kerdock derived code is similar to the Large set of Kasami sequences in performance; it will therefore not be assigned in this problem.

9.2 Assignment of LCS code using $MSC\hat{Z}$

In this section the 358 assignment problem will be solved using an LCS code. Recall that an LCS code is an example of a code for quasi-synchronous operation. The codewords are assigned to transmitters as spreading codewords.

9.2.1 Assignment of an LCS code of length 255 using $MSC\hat{Z}$

In section 9.1.1, it is stated that at least 60 codewords are required for a satisfactory code re-use but a set of LCS codewords of length 255 only contains 15 codewords, with $\tau_{\max} = 17$. Four sets of LCS codewords could be combined to give 60 codewords, but the average of all the correlations is $2.827 \times 10^{-3} > 1.54 \times 10^{-3}$, which does not satisfy the bound. By direct implication, an LCS code of length 255 does not have satisfactory correlations for this problem.

To show how effective the correlation bound is, 60 codewords of 4 sets of LCS codewords of length 255 are assigned at $14dB$ threshold. A random starting assignment obtained a true cost of 104220 at 44.3669% coverage. Average percentage of interferences due to codeword re-use and correlation properties are 6.79017% and 93.2098% respectively. The value of $\sigma_{\min}$ obtained is $0dB < \sigma'$ - an unsatisfactory codeword re-use. The algorithm then optimised this initial solution to achieve a true cost of 30660.4 at 53.7709% coverage. The minimum codeword re-use distance achieved is $15dB > \sigma'$. On average, only 0.0122605% of the true cost is the interference contribution due to code re-use. This further confirms that correlation values of 60 LCS codewords of length 255 are unsatisfactory for this problem. The low correlation zone with $\tau_{\max} = 17$ for LCS of length 255 is also too tight for this problem.

9.2.2 Assignment of an LCS code of length 1023 using $MSC\hat{Z}$

In this case, two sets of LCS codewords (with each set containing 31 codewords) of length 1023 are combined to give 62 codewords. The value of $\tau_{\max}$ is 33. The shift sequences used in constructing the two sets of LCS codewords were obtained using primitive polynomials $x^{10} + x^3 + 1$ and $x^{10} + x^8 + x^3 + x^2 + 1$ respectively. The number of codewords available then satisfies the requirement for satisfactory codeword re-use. The average of all the $MSC\hat{Z}$ correlations is $4.98 \times 10^{-4} < 1.54 \times 10^{-3}$ and so the LCS code satisfies the correlation bound at $14dB$.

As previously demonstrated in chapter 7, the $MSC\hat{Z}$ values of the combined sets of LCS codewords vary from pair to pair of codewords. Low $MSC\hat{Z}$ values are maintained for pairs of codewords from the same set but are much worse for pairs of codewords from different sets. Ideally, pairs of transmitters within the same cell should be assigned pairs of codewords from the same set. Pairs of transmitters from distant cells may be assigned pairs of codewords from different sets or re-use a codeword.

At $14dB$, a random starting assignment gave a true cost of 73792.2 at 99.0347% coverage. On average, 99.0347% of the total interference is due to codeword re-use. The $\sigma_{\min}$ obtained is $0dB < \sigma'$. This assignment is then unsatisfactory. Tabu search optimised this initial assignment to achieve a $\sigma_{\min}$ of $19dB$, a zero true cost and a 100% coverage. All the optimised solutions are satisfactory with zero cost, 100% coverage and $\sigma_{\min} > \sigma'$. Tables 9.1 and 9.2

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	73792.2	94.1341	99.0347	0.965281	0
15dB	175118	90.9683	94.93	5.07001	0
16dB	398897	85.987	86.7698	13.2302	0
17dB	557633	85.0559	70.2922	29.7078	0

Table 9.1: *A table to show initial random starting results using two sets of LCS codewords of length 1023. Minimum codeword re-use SIR, $\sigma_{\min}$, is unsatisfactory for all the assignments.*

threshold	true cost	coverage	co-code%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	19
15dB	0	100	0	0	19
16dB	0	100	0	0	15
17dB	0	100	0	0	19

Table 9.2: *A table to show the optimised results (with starting results shown in table 9.1) using two sets of LCS codewords of length 1023. The value of $\sigma_{\min} > \sigma'$ and so is satisfactory in all cases.*

present the starting and the optimised results for other thresholds. It is clear that the use of a tabu search algorithm is effective in this case.

9.3 Assignment of an LS code using $MSC\hat{Z}$

An LS code (being an example of a code for quasi-synchronous operation) will be assigned to transmitters as a spreading code. To achieve $\tau_{\max} = 32$, the construction by Stańczak et. al in [78] will not be appropriate for this problem. If an LS code of length 1088 is used only 32 codewords can be constructed which would give unsatisfactory codeword re-use. For this reason, the approach introduced in [35] is used. 64 LS codewords of length 1296 are constructed with all internal padding lengths assigned to transitions, i.e. $LS : (Tr, NTr) = (31, 0)$; $\pi = 010101\ldots010101$. A 32×32 Hadamard matrix is used with Golay pairs of length 16. The Golay pairs (of length 16) used are constructed from a Golay pair of length 8 as follows:
Recall the Golay pair:

$$C_{0_8}(Z) = 1 + z + z^2 - z^3 + z^4 + z^5 - z^6 + z^7,$$

$$S_{0_8}(Z) = 1 + z + z^2 - z^3 - z^4 - z^5 + z^6 - z^7.$$

A Golay pair of length 16 is then constructed as:

$$C_{0_{16}}(Z) = C_{0_8} + Z^{15} S_{0_8}$$

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	97772.8	92.1788	99.4582	0.541833	0
15dB	123835	93.8547	99.3611	0.638913	0
16dB	216967	93.2961	99.4315	0.568481	0
17dB	276133	94.5065	96.0815	3.91854	0

Table 9.3: *A table to show unsatisfactory initial random starting assignments using an LS code of length 1296.*

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	19
15dB	0	100	0	0	19
16dB	0	100	0	0	19
17dB	0	100	0	0	19

Table 9.4: *A table to show the satisfactory optimised results (with starting results of table 9.3) obtained by tabu search using an LS code of length 1296. All $\sigma_{\min}$ are greater than re-use distance threshold, σ'.*

$$S_{0_{16}}(Z) = C_{0_8} - Z^{15} S_{0_8},$$

with mate $C_{1_{16}}(Z), S_{1_{16}}(Z)$ constructed as:

$$C_{1_{16}}(Z) = Z^{15} S_{0_{16}}(Z^{-1})$$

$$S_{1_{16}}(Z) = Z^{15} C_{0_{16}}(Z^{-1}).$$

The average $\psi_{x,y}$ of the 64 codewords obtained using the construction above is 4.81×10^{-4} which is less than the 1.54×10^{-3} required.

At $14dB$ and other thresholds, random initial assignments started with unsatisfactory assignments (the minimum codeword re-use, $\sigma_{\min}$, obtained is $0dB < \sigma'$). These results are presented in table 9.3. Tabu search then effectively optimised these initial assignments to achieve a zero true cost at 100% coverage. All the optimised assignments presented in table 9.4 achieved codeword re-use, $\sigma_{\min} > \sigma'$. Again, tabu search is able to effectively achieve satisfactory spreading code assignment using an LS code of length 1296.

9.4 Assignment of the combination of a Walsh-Hadamard code and a Gold code using $MSC\hat{Z}$

In the downlink of a DS-CDMA system, a Walsh-Hadamard code and a Gold code are used as the spreading code and the scrambling code respectively [13]. The Walsh-Hadamard code is used to maintain orthorgonality between transmitters within the same cell while the Gold code is used to maintain low correlation between transmitters in different cells. A scrambling code is therefore cell specific. All transmitters within the same cell are assigned the same scrambling code (in this case, a Gold codeword), different from those assigned to other cells. A scrambling code may be re-used between cells that are farther apart. Interference at any reception point in a particular cell is only due to interfering transmitters from other cells and they correspond to correlations between pairs of scrambling codewords used.

Assignment of a spreading code (or a scrambling code) can be viewed as a cell assignment problem. In the case of the 358 transmitter problem considered here, a Gold code of length 1023 is used as the scrambling code. Since orthogonality is maintained between any pairs of Walsh-Hadamard codewords, no special code planning is required. Walsh-Hadamard codewords are assigned at random but not re-used within a cell.

Recall that this problem consists of 61 cells and so the assignment is just about selecting the best 61 Gold codewords out of the 1024 codewords available. A simulated annealing algorithm is used for this assignment. Interference from interfering transmitters within the same cell as the wanted transmitter is set to zero - the crosscorrelation properties between any pairs of Walsh-Hadamard codewords operating synchronously. Interference from interfering transmitters from cells different from that of the wanted transmitters use the correlation properties of the Gold code. Initial random assignments obtained are presented in table 9.5. All the random assignments (without any optimisation) obtained satisfactory assignment - zero true costs and 100% coverages.

9.5 Discussion of results

In this chapter a 358 transmitter problem has been solved using an LCS code of length 1023, an LS code of length 1296, and a combination of a Walsh-Hadamard (as a spreading code) and a Gold code (as a scrambling code). The construction of an LS code with internal padding is used as the construction by Stańczak et. al in [78] gives too few codewords for satisfactory codeword re-use. The optimised

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	no codeword re-use
15dB	0	100	0	0	no codeword re-use
16dB	0	100	0	0	no codeword re-use
17dB	0	100	0	0	no codeword re-use

Table 9.5: *A table to show satisfactory random initial assignments obtained using a Walsh-Hadamard code as a spreading code and Gold code of length 1023 as a scrambling code.*

results presented in tables 9.2, and 9.4 clearly show the efficiency of the tabu search algorithm when codes for quasi-synchronous operation are adopted as spreading codes. Careful assignment of the spreading codes to avoid interference from codeword re-use and $MSC\hat{Z}$ lead to better system performance. Table 9.5 justifies the random assignment approach adopted in WCDMA (or third generation mobile telephone system). All the results presented in tables 9.2, 9.4, and 9.5 achieved equal performances.

It appears that this problem is too easy to illustrate the advantages of quasi-synchronous operation. It will be necessary to note that this problem is assumed to operate in a perfect channel without noise and signal multi-path. If these channel conditions are considered, the advantage of spreading codes for quasi-synchronous operation over asynchronous operation (using a Gold code) will become apparent. The next chapter considers a similar problem to the one presented here but with more cells. Harder versions of the 358 transmitter problem will appear in chapter 11.

Chapter 10

A 1794 Transmitter Problem

10.1 Description of Problem

In this chapter a 1794 transmitter problem is considered. This problem was generated by applying the same method as that used for generating the 358 transmitter problem presented in chapter 9. The only difference is that more cells are introduced into the network without increasing the cell demands of the previous problem (which is a subset of the new problem considered here). The coordinates of the 310 cell centres are given by $x = 1000 \times (j-1) + 500 \times i$, $y = 500\sqrt{3} \times i$, with $i, j \in \{1, 2, \ldots, 20\}$, $10 < i + j < 32$. They are generated sequentially in the order given by the sequence $(1, 10), (1, 11), \ldots, (1, 20), (2, 9), \ldots, (20, 11)$ for (i, j). Again the set of receiver test points R corresponds to the vertices of the regular hexagons. Each vertex may correspond to one, two or three receiver test points with different serving transmitters at the centre of an adjacent cell. The total number of cells available in the network is now 310 with demands arranged as follows: (6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 3; 3; 5; 5; 7;

6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 3; 3; 5; 5; 8;

7; 7; 5; 4; 8; 8; 7; 5; 7; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5;

5; 6; 4; 4; 3; 3; 5; 5; 7; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5;

5; 6; 4; 4; 3; 3; 5; 5; 8; 7; 7; 5; 4; 8; 8; 7; 5; 7; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8;

7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 3; 3; 5; 5; 7; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 4; 6; 8;

7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 3; 3; 5; 5; 8; 7; 7; 5; 4; 8; 8; 7; 5; 7; 6; 8; 7; 7; 5; 4; 8; 8; 7;

5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4; 3; 3; 5; 5; 7; 6; 8; 7; 7; 5; 4; 8; 8; 7;

5; 5; 5; 6; 4; 4; 4; 6; 8; 7; 7; 5; 4; 8; 8; 7; 5; 5; 5; 6; 4; 4).

Extension of the 358 transmitter problem to the 1794 transmitter problem could occur when a mobile operator extends its coverage areas. One of the advantages of a DS-CDMA system is that when a new user is introduced, interference from the new user is shared by all the existing users. Thus, there may be no need for more frequency spectrum or a longer spreading code. When the system is operated quasi-synchronously and an LCS or an LS code is used it will be assumed that $-31 \leq \tau \leq 31$.

10.1.1 Analyses of 1794 transmitter problem

This problem is first modelled as a graph colouring problem as described in section 5.6. This approach suggested that the same minimum of 60 codewords is required for satisfactory codeword re-use. The value of $IP_{\max}$ described in equation (5.43) is 27.3601 and the correlation bound in equation (5.45) is 1.46×10^{-3}. This suggests that the same set of codes (and the implied length) used to solve the 358 transmitter problem could be used in solving this problem.

10.1.2 Software Specification

The same specifications stated in section 9.1.2 are used in the software used for this problem. As more interference is experienced from the introduction of more users than in the previous 358 transmitter problem, the algorithm is allowed to prioritise the optimisation process. The algorithm is allowed to first solve the problem to achieve satisfactory codeword re-use. A correlation switch is introduced into the system. Correlations are switched off until satisfactory codeword re-use is achieved. They are by default set to zero except when a codeword is unsatisfactorily re-used in which case it is set to one. Once the cost (graph colouring cost) is zero, the two correlation files containing correlations at $\tau = 0$ and $\tau \in \{-N+1, \ldots, N-1\}$ (for asynchronous operation) or $\tau \in \{-\tau_{max}+1, \ldots, \tau_{max}-1\}$ (for quasi-synchronous operation) are automatically switched on. At this stage, the algorithm begins to take the variations in $MSC\hat{Z}$ values into account.

10.2 Assignment of an LCS code of length 1023 using $MSC\hat{Z}$

In this section the same sets of LCS codewords used in section 9.2.2 are used. Two sets, each with 31 codewords, are combined to give a total of 62 codewords.

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	2.75103×10^8	92.4749	98.9401	1.05988	0
15dB	7.29193×10^8	92.094	95.5481	4.4519	0
16dB	2.10391×10^9	88.8424	76.0993	23.9007	0
17dB	5.0519×10^9	84.2252	50.3156	49.6844	0

Table 10.1: *A table to show initial random starting results using two sets of LCS codewords of length 1023. Minimum codeword re-use $\sigma_{\min} = 0$ is unsatisfactory.*

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	15
15dB	0	100	0	0	15
16dB	0	100	0	0	15
17dB	0	100	0	0	15

Table 10.2: *A table to show the optimised results (with starting results shown in table 10.1) using two sets of LCS codewords of length 1023. The values of $\sigma_{\min}$ achieved are satisfactory.*

Average $MSC\hat{Z}$ is 4.98×10^{-4} which is less than $\Phi_{x,y} = 1.46 \times 10^{-3}$, of equation (5.45).

At $14dB$, the random initial assignment obtained a true cost of 2.75103×10^8 with a 92.4749% coverage. On average, 98.9401% of the interference is due to unsatisfactory codeword re-use. $\sigma_{\min}$ obtained is $0dB < 14dB$ and so codewords are unsatisfactorily re-used. Tabu search optimised the initial random start solution to achieve a zero true cost at 100% coverage. $\sigma_{\min}$ achieved is $15dB > \sigma'$. The random starting and optimised assignments at other SIR thresholds are presented in tables 10.1 and 10.2. All the optimised assignments presented in table 10.2 gave zero true cost with 100% coverage and satisfactory codeword re-use. These results demonstrate the effectiveness of the tabu search algorithm used.

10.3 Assignment of an LS code of length 1296 using $MSC\hat{Z}$

The same set of LS codewords (with internal padding) of length 1296 constructed in section 9.3 for the 358 transmitter problem is used. Tables 10.3 and 10.4 present the random initial and optimised assignments respectively. Comparing these two tables, it is clear that the tabu search algorithm is effective in this optimisation problem. The need for a careful assignment to achieve a satisfactory solution such as the one presented in table 10.4 is also apparent.

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	97772.8	92.1788	99.4582	0.541833	0
15dB	123835	93.8547	99.3611	0.638913	0
16dB	216967	93.2961	99.4315	0.568481	0
17dB	276133	94.5065	96.0815	3.91854	0

Table 10.3: *A table to show unsatisfactory initial random starting assignments using an LS code of length 1296.*

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	19
15dB	0	100	0	0	19
16dB	0	100	0	0	19
17dB	0	100	0	0	19

Table 10.4: *A table to show the satisfactory optimised results (with starting results of table 9.3) obtained by tabu search using an LS code of length 1296. All* $\sigma_{\min}$ *are greater than the codeword re-use threshold,* σ'.

10.4 Assignment of the combination of a Walsh-Hadamard code and a Gold code using $MSC\hat{Z}$

The assignment presented in this section uses a Walsh-Hadamard code as spreading code and a Gold code as scrambling code as described in section 9.4. Table 10.5 presents random assignments obtained by the simulated annealing algorithm. All the assignments are satisfactory but simulated annealing is further used to optimise the cost at $17dB$ to achieve a zero cost at 100% coverage. It is then apparent that the use of an optimisation algorithm at higher thresholds is necessary to obtain a better solution than those achievable using just a random assignment method.

10.5 Discussion of Results

In this chapter, a 1794 transmitter problem (consisting of 310 cells) has been solved using the LCS code of length 1023, the LS code of length 1296, and the combination of a Walsh-Hadamard as spreading code and the Gold code of length 1023 as scrambling code. Even though there is more interference in the 1794 transmitter problem presented here than that presented in chapter 9, the same number and length of code (hence, same frequency spectrum) has been used to obtain similar results. This is due to the effectiveness of the algorithm used.

threshold	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	no codeword re-use
15dB	0	100	0	0	no codeword re-use
16dB	0	100	0	0	no codeword re-use
17dB	139.414	98.7087	0	100	no codeword re-use

Table 10.5: *A table to show random initial assignments obtained using a Walsh-Hadamard code as a spreading code and a Gold code of length 1023 as a scrambling code. The results are satisfactory. Tabu search is further used to optimise the cost at 17dB down to zero cost.*

Comparing the random start results of tables 10.1 and 10.3 with the optimised results of tables 10.2 and 10.4 it is clear that the need for a good optimisation algorithm like the tabu search algorithm is important when these types of LCS code and LS code are used as spreading codes. The optimised results obtained at $17dB$ in section 10.5 also demonstrates the need to optimise the small variations in Gold code correlation values in order to further reduce interference in the system. The problem is still a little too easy to demonstrate the advantages of quasi-synchronous operation. Harder versions of this problem will appear in chapter 12.

Chapter 11

The 358 Transmitter Problem with Overlapping Cells

11.1 Description of problem

This problem is formulated in the same way as the 358 transmitter problem in chapter 9 except that neighbouring cells are allowed to overlap. There are a total of 366 distinct receiver locations in this problem. Two modifications are considered. In the first, each receiver location is moved away from the serving transmitter by a factor of 1.1. In the second, each receiver location is moved away from the serving transmitter by a factor of 1.2. These problems could be viewed as cases when coverage areas of neighbouring cells are allowed to overlap each other. In practise, this helps to facilitate call handover. Thus, the problems are harder than the original problem. When the system is operated quasi-synchronously, a low correlation zone with $-31 \leq \tau \leq 31$ is desirable. A minimum of 60 codewords are required for satisfactory codeword re-use at $\sigma' = 14dB$.

11.2 Overlapping cells by a factor of 1.1

Here receiver points are moved away from the serving transmitters by a factor of 1.1. Due to some parts of a cell overlapping into a neighbouring cell, more interference is incurred in the system than in the non-overlapping cell problem in chapter 9. The value of $IP_{\max}$ becomes 37.3317 and so the bound of equation (5.45) at $14dB$ is 1.07×10^{-3}. Comparing this bound with average $MSC\hat{Z}$ of the two sets of LCS codewords (which is 4.98×10^{-4}), the LS code (which is 4.81×10^{-4}), and the Gold code (which is 9.77×10^{-4}) which are all presented in chapter 9, it is clear that the same lengths of these codes can be used for this problem.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	129731	89.5717	97.7255	2.27448	0
15dB	175272	90.3166	88.3134	11.6866	0
16dB	285396	85.1024	57.3291	42.6709	0
17dB	389999	73.2775	23.7796	76.2204	0

Table 11.1: *A table to show unsatisfactory random starting assignments of the two sets of LCS codewords for the first 358 transmitter overlapping cells problem. Much of the interference is due to codeword re-use (ACC%) at 14dB to 16dB.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	18
15dB	0	100	0	0	18
16dB	0	100	0	0	18
17dB	16.9581	98.848	0.00295348	99.997	18

Table 11.2: *A table to show the optimised results (with starting results shown in table 11.1) using the two sets of LCS codewords of length 1023. The values of $\sigma_{\min} > \sigma'$ achieved show that codewords are satisfactorily re-used.*

11.2.1 Software specification

The same specification used for the 1794 transmitter problem in chapter 10 is used. The problem is solved by initially switching off and later switching on the correlations. Correlations are by default switched off (i.e. set to zero) for the algorithm to only work out an assignment with satisfactory codeword re-use. Once satisfactory codeword re-use is achieved, correlations are then switched on to enable the algorithm to take account of the variations in the correlation in order to reduce interference.

11.2.2 Assignment of an LCS Code of length 1023 using $MSC\hat{Z}$

The same LCS code of length 1023 as in section 9.2.2 (formed by combining two sets, each of size 31) is used. The random initial and optimised results are presented in tables 11.1 and 11.2. Comparing the random start results with the optimised results, the need for an optimisation algorithm to assign an LCS code of this type is more apparent. The tabu search algorithm achieved satisfactory solutions with zero costs and 100% coverages (except at $17dB$ with 98.85% coverage). The minimum codeword re-use measure $\sigma_{\min}$ achieved is greater than the value $\sigma' = 14dB$ required.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	83832.3	93.2961	99.069	0.931005	0
15dB	135120	93.2961	99.0529	0.947063	0
16dB	271801	90.7356	89.5817	10.4183	0
17dB	394748	89.1993	69.1724	30.8276	0

Table 11.3: *A table to show unsatisfactory initial random assignments of an LS code of length 1296.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	18
15dB	0	100	0	0	18
16dB	0	100	0	0	18
17dB	0	100	0	0	18

Table 11.4: *A table to show the satisfactory optimised results (with starting results of table 11.3) obtained by tabu search using an LS code of length 1296 . All* $\sigma_{\min}$ *are greater than the re-use distance threshold,* σ'.

11.2.3 Assignment of an LS code of length 1296 using $MSC\hat{Z}$

The same LS code with internal padding presented in section 9.3 is used. The initial random and optimised results are presented in tables 11.3 and 11.4. Again, the algorithm used is able to successfully optimise the initial solutions to achieve satisfactory solutions with zero costs, 100% coverages and minimum codeword re-use $\sigma_{\min} > \sigma'$.

11.2.4 Assignment of the combination of a Walsh-Hadamard code and a Gold code using $MSC\hat{Z}$

The same method presented in section 9.4, where a Walsh-Hadamard code is used as a spreading code and a Gold code is used as a scrambling code, is used here. Again, no codeword re-use is necessary. The random initial results are presented in table 11.5. Random assignments at $14dB$ and $15dB$ are satisfactory but the results obtained at $16dB$ and $17dB$ are optimised (and presented in table 11.6) to achieve true costs of 1471.91 and 71645.7 at 85.149% and 45.0652% coverages respectively. Although, the algorithm worked on optimising the small variations in the $MSC\hat{Z}$ values of the Gold code, the results obtained are not good enough. It could be inferred from this that in harsh network conditions a Gold code fails.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	no codeword re-use
15dB	0	100	0	0	no codeword re-use
16dB	7249.58	75.5121	0	100	no codeword re-use
17dB	126195	38.0354	0	100	no codeword re-use

Table 11.5: *A table to show random initial assignments of a Walsh-Hadamard code as a spreading code and a Gold code of length 1023 as a scrambling code.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
16dB	1471.91	85.149	0	100	no codeword re-use
17dB	71645.7	45.0652	0	100	no codeword re-use

Table 11.6: *A table to show the optimised results obtained by the simulated annealing algorithm using a Walsh-Hadamard code as a spreading code and a Gold code of length 1023 as a scrambling code. Only very small improvements on the corresponding initial results of table 11.5 are achieved.*

11.2.5 Discussion of Results

Comparing the random starting solutions and the optimised solutions of codes for quasi-synchronous operation (i.e. the LCS code and the LS code), it is apparent that the use of the tabu search algorithm greatly improved system performance. With the combination of the Walsh-Hadamard and the Gold code, the simulated annealing algorithm could only make small improvements on the solutions as the variation in $MSC\hat{Z}$ values is small.

The optimised solutions achieved by the algorithms are presented in graphical form in figures 11.1 and 11.2. It is clear from these graphs that with careful spreading code assignment (such as that demonstrated here) the method of spreading signals using codes for quasi-synchronous operation outperforms the method of spreading and then scrambling the signal using the Walsh-Hadamard code and the Gold code. The LS code at $17dB$ also performed better than the LCS code.

11.3 Overlapping cells by a factor of 1.2

The problem described in section 11.2 can be made even harder by moving receiver points away from the serving transmitters by a factor of 1.2 instead of 1.1. The value of $IP_{\max}$ becomes 53.6406 and so the bound $\Phi_{x,y}$ of equation (5.45) at $14dB$ becomes 7.42×10^{-4}. Comparing this with average $MSC\hat{Z}$ of the same two sets of LCS codewords (which is 4.98×10^{-4}) and LS code (which is 4.81×10^{-4}), it is clear that good assignment could be obtained with the

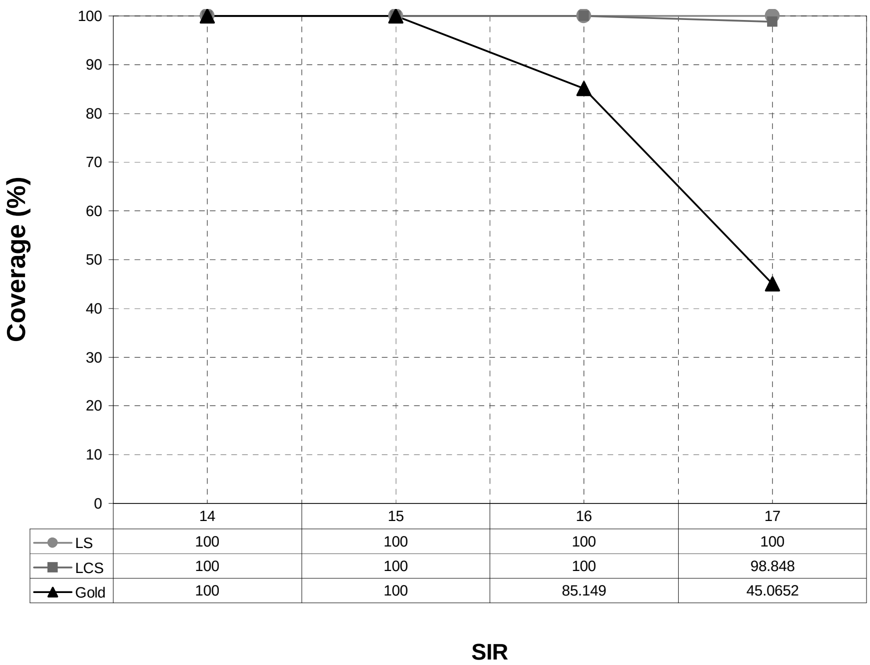


Figure 11.1: *This graph shows the optimised coverage results achieved for the first 358 transmitter overlapping cells problem using the LCS code of length 1023, the LS code of length 1296 and the combination of the Walsh-Hadamard and the Gold code of length 1023. The codes for quasi-synchronous operation (i.e. the LCS code and the LS code) clearly outperformed the asynchronous operation of the Gold code.*

use of an optimisation algorithm at $14dB$. The Gold code, on the other hand, with average $MSC\hat{Z}$ of 9.77×10^{-4}, does not satisfy the bound of equation (5.45). To make a Gold code work for this problem a greater length than 1023 should be used. But for performance comparison purposes (with the LS code of length 1296 and the LCS code of length 1023), the Gold code of length 1023 is used here. A preliminary solution to this problem using the graph described in section 5.6 indicates that a minimum of sixty codewords are required to achieve a satisfactory codeword re-use at $\sigma' = 14dB$.

For the two sets of LCS codewords, the random initial and optimised solutions are presented in tables 11.7 and 11.8. For the LS code, the initial random and optimised solutions are presented in tables 11.9 and 11.10. From these tables the effectiveness of the tabu search algorithm is apparent. However, poor coverages are achieved at an SIR threshold of $17dB$. For the combination of the Walsh-Hadamard code and the Gold code, the random initial solutions and the optimised solutions are presented in tables 11.11 and 11.12 respectively. All the random initial solutions presented are unsatisfactory. The optimised assignments are also unsatisfactory. The Gold code of length 1023 is indeed not

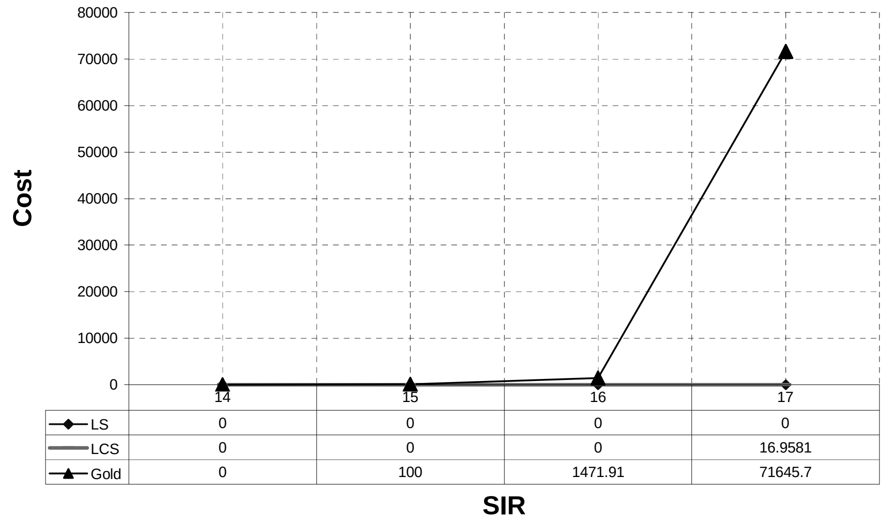


Figure 11.2: *This graph shows the optimised cost results achieved for the first 358 transmitter overlapping cells problem using the LCS code 1023, the LS code of length 1296 and the combination of the Walsh-Hadamard and the Gold code of length 1023. The codes for quasi-synchronous operation (i.e. the LCS code and the LS code) clearly outperformed the asynchronous operation of the Gold code.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	108817	90.7356	91.9403	8.05972	0
15dB	196490	80.3073	47.2589	52.7411	0
16dB	248656	68.0633	20.6637	79.3363	0
17dB	590615	46.5084	14.9407	85.0593	0

Table 11.7: *A table to show unsatisfactory random starting solutions using the LCS code of length 1023. At thresholds of 15dB to 17dB much of the interference is due to the* $MSC\hat{Z}$ *correlations (AIC%).*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	16
15dB	0	100	0	0	16
16dB	120.553	96.7877	0.00327114	99.9967	16
17dB	28424.2	66.3408	0.00360083	99.9964	16

Table 11.8: *A table to show the optimised solutions (with starting solutions shown in table 11.7) using the LCS codewords of length 1023. All the* $\sigma_{\min}$*s achieved are satisfactory.*

appropriate for this problem.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	147684	87.7095	94.1856	5.81445	0
15dB	147381	90.1304	72.6935	27.3065	0
16dB	321092	79.8883	46.5086	53.4914	0
17dB	600098	65.1304	28.3849	71.6151	0

Table 11.9: *A table to show unsatisfactory initial random assignments using an LS code of length 1296.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	16
15dB	0	100	0	0	16
16dB	0	100	0	0	16
17dB	2367.12	89.4786	0.00295348	99.997	16

Table 11.10: *A table to show the satisfactory optimised results (with starting results of table 11.9) obtained by tabu search using an LS code of length 1296. All* $\sigma_{\min}$ *are greater than codeword re-use threshold,* $\sigma' = 14dB$.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	1266.39	83.054	0	100	no codeword re-use
15dB	38593.7	43.0633	0	100	no codeword re-use
16dB	228093	25.4655	0	100	no codeword re-use
17dB	775994	15.689	0	100	no codeword re-use

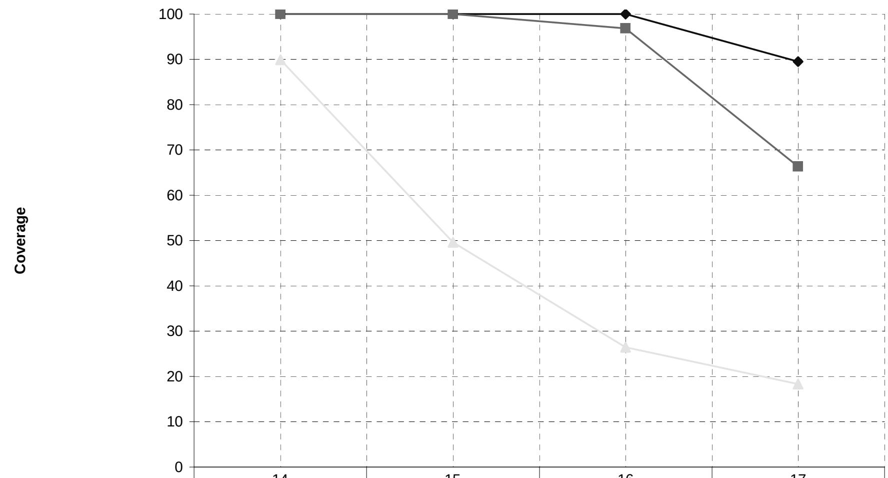
Table 11.11: *A table to show unsatisfactory random initial solutions obtained using the Walsh-Hadamard code as spreading code and the Gold code of length 1023 as the scrambling code.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	145.984	89.9441	0	100	no codeword re-use
15dB	20724.6	49.6276	0	100	no codeword re-use
16dB	172306	26.4432	0	100	no codeword re-use
17dB	648230	18.2961	0	100	no codeword re-use

Table 11.12: *A table to show optimised solutions obtained (with corresponding initial assignments of table 11.11) using the Walsh-Hadamard code as the spreading code and the Gold code of length 1023 as the scrambling code. All the assignments are unsatisfactory.*

11.3.1 Discussion of results

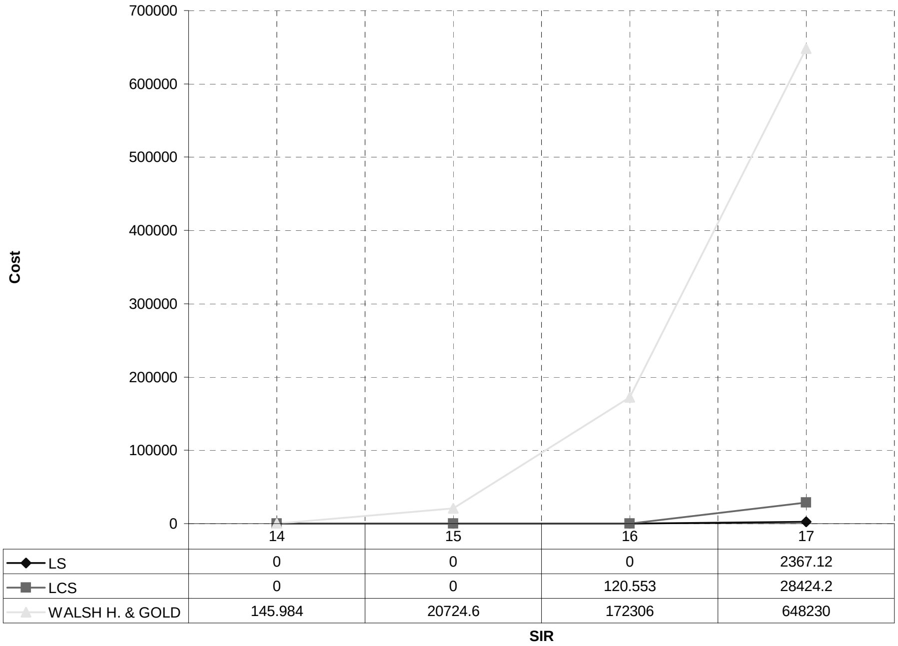
As with other problems previously presented it is apparent from figures 11.3 and 11.4 that the use of the tabu search algorithm results in better system perfor-



	14	15	16	17
LS	100	100	100	89.4786
LCS	100	100	96.7877	66.3408
WALSH H. & GOLD	89.9441	49.6276	26.4432	18.2961

Figure 11.3: *This graph shows the optimised coverage results achieved for the second 358 transmitter overlapping cells problem (overlapping by a factor of 1.2) using the LCS code of length 1023, with* $\tau_{\max} = 33$*, the LS code of length 1296, with* $\tau_{\max} = 32$*, and the combination of the Walsh-Hadamard and the Gold code of length 1023. The quasi-synchronous operation of the LCS code and the LS code clearly outperformed the asynchronous operation of the Gold code.*

mance (measured in SIR and area coverage) for quasi-synchronous operation using the two sets of LCS codewords and the LS code than asynchronous operation using the combination of the Walsh-Hadamard code and the Gold code. The LS code performed better than the LCS code and the LCS code performed much better than the Gold code.



SIR	14	15	16	17
LS	0	0	0	2367.12
LCS	0	0	120.553	28424.2
WALSH H. & GOLD	145.984	20724.6	172306	648230

Figure 11.4: *This graph shows the corresponding cost results to the coverage results presented in figure 11.3 for the second 358 transmitter overlapping cells problem (overlapping by a factor of 1.2).*

Chapter 12

The 1794 Transmitter Problem with Overlapping Cells

12.1 Description of problem

The examples considered here are similar to the 1794 transmitter problem presented in chapter 10. The only difference is that the problems solved here allow the 310 cells to overlap in a similar way to the overlapping cells problems presented in chapter 11. The distance of each reception point from its serving transmitter is increased by a factor of 1.1 or 1.2. The problems are then harder than the original problem presented in chapter 10. The same demands presented in section 10.1 are maintained. There are 1794 transmitters, 1860 distinct receiver locations and a total of 10764 receiver instances. When the system is operated quasi-synchronously, a low (or zero) correlation zone with $-31 \leq \tau \leq 31$ is required. A minimum of 60 codewords is required for satisfactory codeword re-use at $\sigma_{\min} = 14dB$. For both examples, the same system specification used for the problems presented in chapters 10 and 11 is employed. The tabu search algorithm is used to assign the LCS code and the LS code. The simulated annealing algorithm is used to assign the combination of the Walsh-Hadamard code and the Gold code. The mean planning method is used.

12.2 Overlapping cells by a factor of 1.1

Here reception points are moved away from the serving transmitters by a factor of 1.1. This makes the problem harder than that presented in chapter 10. The way that the cells overlap creates more interference at any given reception point than the non-overlapping cells. The value of $IP_{\max}$ in equation (5.43) becomes 39.4966 and so average $MSC\hat{Z}$ of any code to be used on this network must be less than or equal to $\Phi_{x,y} = 1.01 \times 10^{-3}$. The same set of codes described in chapter 9 will be used.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	448285	92.7072	96.211	3.78898	0
15dB	792280	90.4775	80.6066	19.3934	0
16dB	1.3036×10^{6}	83.2033	45.7922	54.2078	0
17dB	2.15561×10^{6}	66.8803	21.2208	78.7792	0

Table 12.1: *A table showing unsatisfactory random starting results for the first 1794 transmitter overlapping cells problem using the two sets of LCS codewords of length 1023. Much of the interference is due to codeword re-use (ACC%) at 14dB to 15dB.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	18
15dB	93.4202	99.8792	0	0	18
16dB	4112.08	97.8168	0.00397445	99.996	18
17dB	74984.1	89.7622	0.00471108	99.9953	18

Table 12.2: *A table showing the optimised results (with starting results shown in table 12.1) using the two sets of LCS codewords of length 1023. In all cases* $\sigma_{\min} > \sigma'$.

12.2.1 Assignment of an LCS code of length 1023 using $MSC\hat{Z}$

From section 9.2.2 average $MSC\hat{Z}$ of the combination of the two sets of LCS codewords is 4.98×10^{-4} which is less than $\Phi_{x,y} = 1.01 \times 10^{-3}$. The random initial solutions and optimised solutions are presented in tables 12.1 and 12.2. Examining the solutions in table 12.2, the solutions achieved at $14dB$ to $16dB$ are satisfactory but the solution at $17dB$ threshold is not very good. Much of this cost is due to high $MSC\hat{Z}$ ($AIC\%$) between pairs of the LCS codewords from different sets. Average $MSC\hat{Z}$ between any pair of codewords from different sets is 9.64×10^{-4} which is greater than $\Phi_{x,y} = 5.05\times10^{-4}$ at $17dB$. This demonstrates that there is a limit to how much an algorithm can optimise the assignment of a code with unsatisfactory correlation. All the assignments achieved satisfactory codeword re-use.

12.2.2 Assignment of an LS code of length 1296 using $MSC\hat{Z}$

Recall from section 9.3 that the average $MSC\hat{Z}$ for the 64 LS codewords of length 1296 is 4.81×10^{-4} which is less than $\Phi_{x,y}$ at $14dB$. The LS code of length 1296 has a satisfactory number of codewords and satisfactory $MSC\hat{Z}$ values. The random initial solutions and optimised solutions are presented in tables 12.3 and

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	454166	92.7536	98.8712	1.12884	0
15dB	760714	92.382	97.7693	2.23068	0
16dB	1.07959×10^6	92.317	85.4062	14.5938	0
17dB	2.04445×10^6	86.715	57.65	42.35	0

Table 12.3: *A table to show unsatisfactory initial random solutions using the LS code of length 1296.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	18
15dB	0	100	0	0	18
16dB	0	100	0	0	18
17dB	0	100	0	0	18

Table 12.4: *A table to show the satisfactory optimised solutions (with starting solutions of table 12.3) obtained by the tabu search algorithm using the LS code of length 1296. All $\sigma_{\min}$ achieved are greater than the codeword re-use threshold, σ'.*

12.4. The random initial solutions obtained in table 12.3 all resulted in unsatisfactory codeword re-use with poor coverages. The optimised solutions presented in table 12.4 achieved satisfactory solutions with excellent coverages (i.e. 100% coverages). The most significant improvement on the random initial solutions is achieved by the tabu search algorithm at $17dB$ threshold. The algorithm started with a cost of 2.04445×10^6 at 86.715% coverage to achieve a zero true cost at 100% coverage.

12.2.3 Assignment of the combination of a Walsh-Hadamard code and a Gold code using $MSC\hat{Z}$

The same method presented in section 9.4 is used. The Walsh-Hadamard code is used as the spreading code and the Gold code is used as the scrambling code. The random initial solutions and the optimised solutions are presented in figures 12.5 and 12.6. As expected, a random starting assignment at $14dB$ obtained a zero cost at 100% coverage. However, at other thresholds, the algorithm could not make any significant improvement on the random initial assignment. The solutions obtained at $16dB$ and $17dB$ demonstrate that the $MSC\hat{Z}$ of the Gold code is not satisfactory at these thresholds.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	no codeword re-use
15dB	74.885	98.3556	0	100	no codeword re-use
16dB	85017.7	62.8019	0	100	no codeword re-use
17dB	990014	23.9502	0	100	no codeword re-use

Table 12.5: *A table to show unsatisfactory (except at 14dB and 15dB) random initial assignments obtained using a Walsh-Hadamard code as spreading code and a Gold code of length 1023 as scrambling code.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
15dB	0.0274526	99.777	0	100	no codeword re-use
16dB	31309	74.0988	0	100	no codeword re-use
17dB	663941	32.2278	0	100	no codeword re-use

Table 12.6: *A table to show optimised solutions achieved using the Walsh-Hadamard code as spreading code and the Gold code of length 1023 as scrambling code. The algorithm achieved very little improvement in the corresponding random initial solutions presented in table 12.5.*

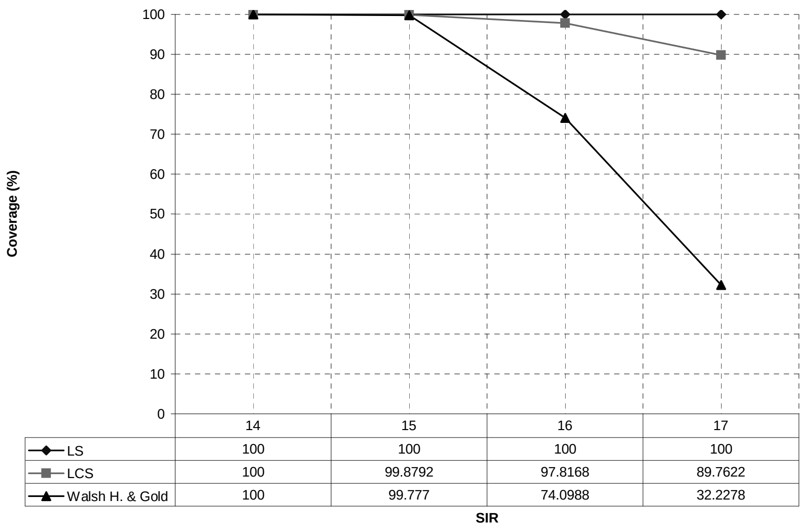
12.2.4 Discussion of results

In this example, a 1794 transmitter overlapping cells problem has been solved. The same sets of spreading code classes used for other cellular problems presented in the previous chapters are used.

In section 12.2.1, the results achieved using the LCS code are presented. The optimised results achieved (compared with the random initial results) justified the use of the tabu search algorithm to minimise interference in the system. The result achieved at $17dB$ indicates a limit on the performance of the LCS code in this problem. Small interference contributions become significant at this stage.

The solutions achieved using the LS code are described in section 12.2.2. The assignments achieved zero costs at 100% coverage even at $17dB$. Although the average of all non-zero correlations of the LS code is 9.61×10^{-4} which is greater than $\Phi_{x,y}$ at $17dB$ the assignment avoids the larger correlations. The tabu search algorithm is able to successfully minimise the resulting interference due to large correlations to achieve satisfactory solutions. Solutions achieved using the Walsh-Hadamard code and the Gold code are described in section 12.2.3. The algorithm is only able to make an improvement of up to $1dB$ on the random results.

The optimised solutions achieved for all the three codes are presented in figures 12.1 and 12.2. Again, it is clear that with careful code assignment, the



	14	15	16	17
LS	100	100	100	100
LCS	100	99.8792	97.8168	89.7622
Walsh H. & Gold	100	99.777	74.0988	32.2278

Figure 12.1: *This graph shows the optimised coverage results achieved for the first example of the 1794 transmitter overlapping cells problems. The LCS code of length 1023, the LS code of length 1296 and the combination of the Walsh-Hadamard code and the Gold code of length 1023 are used. The codes for quasi-synchronous operation (i.e LCS code and LS code) clearly outperformed the Gold code.*

use of the LCS code and the LS code for quasi-synchronous operation achieved better results than the asynchronous operation using the combination of the Walsh-Hadamard code and the Gold code. The LS code achieved better results than the LCS code.

12.3 Overlapping cells by a factor of 1.2

In this example, the 1794 transmitter problem described in chapter 10 is made harder by moving receiver points away from its serving transmitter by a factor of 1.2 instead of 1.1. The value of $IP_{\max}$ for this problem becomes 56.6225 and so the value of $\Phi_{x,y}$ of equation (5.45) becomes 7.03×10^{-4} at $\sigma = 14dB$. Average $MSC\hat{Z}$ of the combined sets of LCS codewords (which is 4.98×10^{-4}) and the LS code (which is 4.81×10^{-4}) thus have satisfactory correlations at an SIR threshold of $14dB$. Average $MSC\hat{Z}$ of the combination of the Walsh-Hadamard and the Gold code of length 1023 (which is 9.77×10^{-4}) does not have satisfactory correlations. However, for performance comparison with the LCS code and the LS code, the Gold code of length 1023 will be assigned. The same minimum of 60 codewords are required for satisfactory codeword re-use with $\sigma_{\min} = 14dB$. The same software requirements used for previous problems are used for this problem. $MSC\hat{Z}$ correlations are initially switched off (i.e. set to zero) until

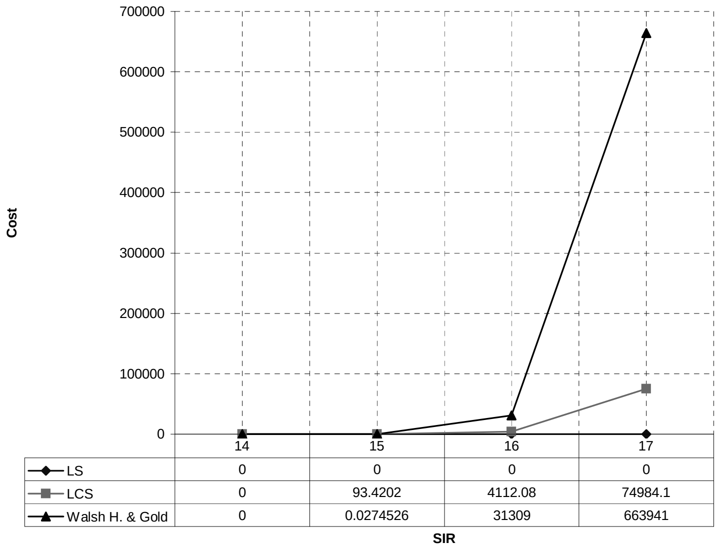


Figure 12.2: *This graph shows the corresponding optimised costs achieved for the first of the 1794 transmitter overlapping cells problems. Codes for quasi-synchronous operation (i.e LCS code and LS code) clearly outperformed the Gold which is for asynchronous operation.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	1.18744×10^6	49.8235	88.1203	11.8797	0
15dB	2.33671×10^6	49.3125	87.232	12.768	0
16dB	4.44149×10^6	47.6124	84.4012	15.5988	0
17dB	8.17426×10^6	43.5712	78.357	21.643	0

Table 12.7: *A table to show unsatisfactory random starting solutions using the LCS code of length 1023.*

the algorithm achieves a satisfactory codeword re-use $\sigma_{\min}$. Correlations are then switched on for the algorithm to take account of the $MSC\hat{Z}$ during the optimisation process.

For the two sets of LCS codewords, the random initial and optimised solutions are presented in tables 12.7 and 12.8. For the LS code, the initial random and optimised solutions are presented in tables 12.9 and 12.10. From these tables the effectiveness of the tabu search algorithm is apparent. For the combination of the Walsh-Hadamard code and the Gold code, the random initial solutions and the optimised solutions are presented in table 12.11 and 12.12 respectively. All the random initial solutions presented are unsatisfactory. The optimised assignments are also unsatisfactory. The Gold code of length 1023 is therefore not appropriate for this problem.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	34.6944	99.8699	0.00302265	99.997	18
15dB	3832.76	96.5905	0.0036764	99.9963	18
16dB	64286.9	83.0639	0.00410132	99.9959	18
17dB	541333	52.016	0.00500579	99.995	18

Table 12.8: *A table to show the optimised solutions (with starting solutions shown in table 12.7) using the LCS codewords of length 1023. The values of* $\sigma_{\min}$ *achieved are satisfactory.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	2.75103×10^{8}	92.4749	98.9401	1.05988	0
15dB	7.29193×10^{8}	92.094	95.5481	4.4519	0
16dB	2.10391×10^{9}	88.8424	76.0993	23.9007	0
17dB	5.0519×10^{9}	84.2252	50.3156	49.6844	0

Table 12.9: *A table to show unsatisfactory initial random assignments using an LS code of length 1296.*

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	0	100	0	0	16
15dB	418.195	99.1081	0	100	16
16dB	12503.7	94.0543	0	100	16
17dB	161125	77.0253	0	100	16

Table 12.10: *A table to show the satisfactory optimised results (with starting results of table 12.9) obtained by the tabu search at 14dB, 15dB and 16dB using an LS code of length 1296. The result achieved at 17dB is unsatisfactory. All* $\sigma_{\min}$ *are greater than codeword re-use threshold,* $\sigma' = 14dB$.

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	21075.4	72.6031	0	0	no codeword re-use
15dB	302527	31.3452	0	0	no codeword re-use
16dB	1.57166×10^{6}	12.7183	0	0	no codeword re-use
17dB	4.94388×10^{6}	7.20922	0	100	no codeword re-use

Table 12.11: *A table to show unsatisfactory random initial solutions obtained using the Walsh-Hadamard code as spreading code and the Gold code of length 1023 as the scrambling code.*

12.3.1 Discussion of results

The optimised solutions are summarised in figures 12.3 and 12.4. It is easy to see that the codes suitable for quasi-synchronous operation (i.e the LCS code and the

σ	true cost	coverage	ACC%	AIC%	$\sigma_{\min}(dB)$
14dB	5498.05	81.1873	0	0	no codeword re-use
15dB	190844	39.9944	0	0	no codeword re-use
16dB	1.21547×10^6	14.3255	0	0	no codeword re-use
17dB	4.24202×10^6	8.54701	0	100	no codeword re-use

Table 12.12: *A table to show optimised solutions obtained (with corresponding initial assignments of table 12.11) using the Walsh-Hadamard code as the spreading code and the Gold code of length 1023 as the scrambling code. All the assignments are unsatisfactory.*

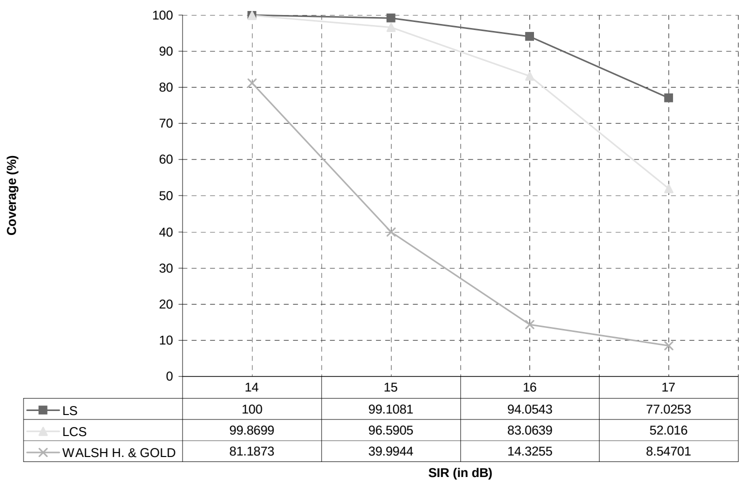
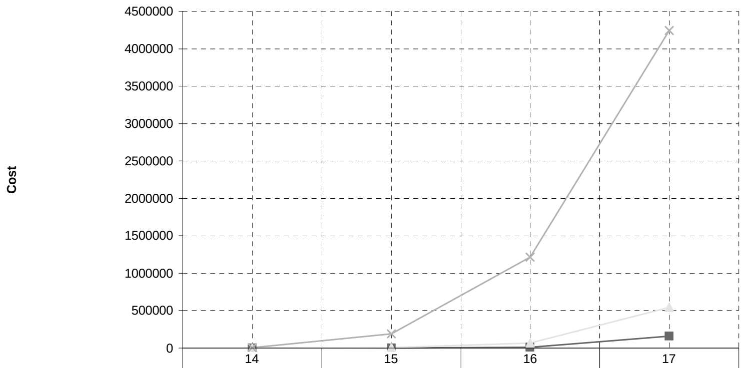


Figure 12.3: *This graph shows the optimised coverage results achieved for the second of the 1794 transmitter overlapping cells problem (overlapping by a factor of 1.2). The codes used are the LCS code of length 1023 with* $\tau_{\max} = 33$*, the LS code of length 1296 with* $\tau_{\max} = 32$*, and the combination of the Walsh-Hadamard and the Gold code of length 1023. The LCS code and the LS code clearly outperformed the combination of the Walsh-Hadamard code and the Gold code.*

LS code) outperformed the combination of the Walsh-Hadamard code and the Gold code (suitable for asynchronous operation). The LS code performed better than the LCS code and the LCS code performed better than the combination of the Walsh-Hadamard code and the Gold code. The effectiveness of the algorithm used in these assignment problems is then apparent.



	14	15	16	17
LS	0	418.195	12503.7	161125
LCS	34.6944	3832.76	64286.9	541333
WALSH H. & GOLD	5498.05	190844	1.22E+06	4.24E+06

SIR (in dB)

Figure 12.4: *This graph shows the corresponding cost results to the coverage results presented in figure 12.3 for the 310 overlapping cells problem (overlapping by a factor of 1.2).*

Chapter 13

Conclusion

This research work has dealt with interference management in DS-CDMA radio systems through careful spreading code assignment. The work incorporated spreading code measures into standard system measures (in particular the Signal-to-Interference Ratio SIR). These spreading code measures are expressed in terms of the voltage output $\hat{Z}$ of a receiver resulting from the interference. Specifically, the measures are the Mean Square of $\hat{Z}$ ($MSC\hat{Z}$ and $MSA\hat{Z}$) and the Peak Square of $\hat{Z}$ ($PSC\hat{Z}$ and $PSA\hat{Z}$). A novel method to determine the average value (of these interference measures) required for a problem is formulated. Also, a method of graph colouring for determining the minimum number of codewords required for a problem has been devised. These techniques can be used for initial code selection. Thus, both the number of codewords and the correlation properties must be taken into account in selecting a code for assignment. Both the simulated annealing and the tabu search algorithms have proved effective in assigning the codes selected.

The work has been able to successfully justify the modifications to increase the number of codewords of the Ling-Chang Simplex (LCS) code and the Loosely Synchronous (LS) code . This is despite the fact that these modifications result in worse correlations between some pairs of codewords. As a result of careful code assignment, the work has shown that the modified LCS and LS codes performed better than the combination of the Walsh-Hadamard code and the Gold code currently used in the 3$^{\text{rd}}$ generation mobile telephone system. It should be noted that the assumptions made for the Walsh-Hadamard and Gold combination are conservative. It has been assumed that the synchronisation within a cell is perfect and thus all intra-cell interference is assumed to be actually zero. In practice this assumption may be optimistic and the actual advantages of the modified LCS and LS codes may be somewhat greater than is presented here.

The modified LCS and LS codes also performed better than the modified

Kerdock code and the large set of Kasami sequences. In very hard problems, the LS code performed better than the LCS code. The LCS code and the LS code are examples of codes suitable for quasi-synchronous operations while the Gold code is suitable for asynchronous operation. Thus the advantages resulting from the use of the modified LCS and LS codes can be balanced against the extra complexity required to implement a quasi-synchronous system. One of the complexities is the use of a GPS as an external timing source. As discussed in section 1.6, an alternative method of self-synchronisation of base stations can be used. The possible uptake of software defined radios is also relevant to the practicability of the systems studied here. However, if the increased complexity is acceptable, it will be possible to use the techniques developed here to create radio systems with much less interference for the same number of users or more users for a given level of interference.

The C++ code used to generate the results in this thesis, together with random number seeds used for the results, is available from the author.

Bibliography

[1] Aardal K.I., Van Hoesel C.P.M., Koster A.M.C.A., Mannino C. and Sassano A., "Models and Solution Techniques for the Frequency Assignment Problem", 4OR 1(4), pp. 261-317, 2003.

[2] Adachi F., Sawahashi M. and Okawa K., "Tree-Structured Generation of Orthogonal Spreading Codes with Different Lengths for Forward Link of DS-CDMA Mobile Radio", *Electronics Letters*, vol. 33, no. 1, pp. 27-28, 1997.

[3] Adachi F., Sawahashi M. and Suda H., "Wideband DS-CDMA for Next-Generation Mobile Communications Systems", *IEEE Communications Magazine*, vol. 36, pp. 56-69, September 1998.

[4] Bertossi A.A. and Bonuccelli M.A., "Code Assignment for Hidden Terminal Interference Avoidance in Multihop Packet Radio Networks", *IEEE/ACM Transactions on Networking*, vol. 3, no. 4, pp. 441-449, 1995.

[5] Bostaz S., Hammons R. and Kumar P.V., "4-Phase Sequences with Near-Optimum Correlation Properties", *IEEE Trans. Inform. Theory*, vol. 38, pp. 1101-1113, May 1992.

[6] Capone A. and Trubian M., "Channel Assignment Problem in Cellular Systems: a New Model and a Tabu Search Algorithm" *IEEE Transactions on Vehicular Technology*, VT-48(4), pp. 1252-1260, July 1999.

[7] Chia, S.T.S. and Lee, W.C.Y., "A Synchronized Radio System Without Stable Clock Sources", *IEEE Personal Communication Magazine*, vol. 8, no. 2, pp. 45-50, April 2001.

[8] Choi B.-J. and Hanzo L., "On the Design of LAS Spreading Codes", *in Proc. 56th IEEE Vehicular Technology Conference*, vol. 4, pp. 2172-2176, September 2002.

[9] Chuck W., "Self-synchronizing a CDMA Cellular Network", *Microwave Journal*, vol. 42, pp. 320-328, May 1999.

[10] Colbourn C.J. and Dinitz J.H., “The CRC Handbook of Combinatorial Designs”, CRC Press, Boca Raton, Florida, 1996.

[11] Cooper G.R. and McGillem C.D., “Modern Communications and Spread Spectrum”, *McGraw-Hill*, USA 1986.

[12] Dahlman E., Gudmundson B., Nilsson M. and Skold, “UMTS/IMT-2000 Based on Wideband CDMA”, *IEEE Communications Magazine*, vol. 36, pp. 70-80, September 1998.

[13] Dinan E.H. and Jabbari B., “Spreading Codes for Direct Sequence CDMA and Wideband CDMA Cellular Networks”, *IEEE Communications Magazine*, pp. 48-54, September 1998.

[14] Dixon R.C., “Spread Spectrum Systems with Commercial Applications”, 3rd edition, John Wiley & Sons, Inc. USA, 1994.

[15] Donelan H.M. and O’Farrell T., “Large Families of Ternary Sequencies with Aperiodic Zero Correlation Zones for a MC-DS-CDMA System”, *in Proc. 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 5, pp. 2322-2326, September 2002.

[16] Eliahou S., Kervaire M. and Saffari B., “A New Restriction on the Lengths of Golay Complementary Sequences”, *J. Combinatorial Theory (A)*, vol. 55, pp. 49-59, 1990.

[17] Fan P.Z., “Spreading Sequence Design and Theoretical Limits for Quasisynchronous CDMA Systems”, *EURASIP Journal on Wireless Communications and Networking*, vol. 2004, no.1, pp. 19-31, 2004.

[18] Fan P.Z. and Hao L., “Generalised Orthogonal Sequences and their Applications in Synchronous CDMA Systems”, *IEICE Trans. Fundamentals*, vol. E83-A, no 11, pp. 1-16, 2000.

[19] Fong M.-H., Bhargava V.K. and Wang Q., “Concatenated Orthogonal/PN Spreading Sequences and their Applications to Cellular DS-CDMA Systems with Integrated Traffic”, *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 3, 1996.

[20] Fong M.-H., Wang Q. and Bhargava V.K., “Concatenated Orthogonal PN codes for DS-CDMA Systems in a Multi-User and Multipath Fading Environment”, *Proc. IEEE Global Telecommunications Conference*, San Francisco, USA, pp. 1642-1646, 1994.

[21] Fukasawa A., Sato T., Takizawa Y., Kato T., Kawabe M. and Fisher R.E., “Wideband CDMA System for Personal Radio Communications”, *IEEE Communications Magazine*, vol. 34, no. 10, pp. 116-123, October 1996.

[22] Gilhousen K.S., Jacobs I.M., Padovani R., Weaver L.A., Wheatley C.E. and Viterbi A.J., "System and Method for Generating Signal Waveforms in a CDMA Cellular Telephone System", US patent, no. 5,103,459, 1992.

[23] Glover F.W. and Kochenberger G.A., "Handbook of Metaheuristics", Kluwer Academic Publishers, Boston, USA, 2002.

[24] Golay M.J.E., "Complementary Series" *IRE Trans. Infor. Theory* vol. 7, pp. 82-87, 1961.

[25] Gold R., "Maximal Recursive Sequences with 3-Valued Cross-Correlation Function", *IEEE Trans. Inform. Theory*, vol. 14, pp. 154-156, 1968.

[26] Golomb S.W., "Correlation Properties of Periodic and Aperiodic Sequences and Applications to Multi-User Systems. *Proc. NATO Advanced Study Institute on Multi-User Communications*, Norwich, UK, pp. 161-197, 1980.

[27] Golomb S.W., "Shift Register Sequences", Revised Edition, Aegan Park Laguna Hills, San Francisco, USA, 1982.

[28] Gong G., "New Designs for Signal Sets with Low Cross Correlation, Balance Property and Large Linear Span: GF(p) Case", *IEEE Trans. Inform. Theory*, vol. 48, no. 11, pp. 2847-2867, November 2002.

[29] Hammons Jr. A.R. and Kumar P.V., "On a Recent 4-Phase Sequences Design for CDMA", *IEICE Trans. Communications*, vol. E7B-b, no. 8, pp. 804-813, August 1993.

[30] Heleseth T., "Some Results about the Cross-Correlation Function between Two Maximal Linear Sequences", *Discrete Math.*, vol. 16, pp. 209-232, 1976.

[31] Hoffman D.G., Leonard D.A., Lindner C.C., Phelps K.T., Rodger C.A. and Wall J.R., "Coding Theory; The Essentials", Marcel Dekker, New York, 1992.

[32] Hung K.W. and Yum T.S., "Efficient Spreading Code Assignment Algorithm for Packet Radio Networks", *Electronics Letters*, pp. 2193-2195, November 1992.

[33] Hurley S., Whitaker R.M. and Smith D.H., "Channel Assignment in Cellular Networks without Channel Separation Constraints", *Proceedings of IEEE Vehicular Technology Conference*, pp. 1714-1718, Fall 2000.

[34] John B.G. and Lawrence E.L., "CDMA Mobile Radio Design", *Artech House Mobile Communication Series*, 2000.

[35] Jones R.A., Perkins S., Rutherford C.G. and Smith D.H., "Quasisynchronous and Asynchronous CDMA: Code Construction and Assignment", University of Glamorgan, Pontypridd, South Wales, UK, 2004.

[36] Jones R.A., Smith D.H. and Perkins S., "Assignment of Spreading Codes in DS-CDMA UWB Systems, *Proceedings of IEEE Conference on Ultra Wide-band Systems and Technologies*, Reston, Virginia, C4, November 2003.

[37] Jung Y-H. and Lee Y.H., "Scrambling Code Planning for 3GPP W-CDMA Systems", in Proceedings of *IEEE Vehicular Technology Conference* vol. 4, pp. 2431-2434, Spring 2001.

[38] Kasami T., "Weight Distribution Formula for Some Class of Cyclic Codes", *Coordinated Science Lab.*, University of Illinois, Urbana, IL, Tech. Report no. R-285, 1966.

[39] Karkkainen K.H., "Code Families and their Performance Measures for CDMA and Military Spread-Spectrum Systems", PhD. Thesis, Oulu University, Oulu, Finland, 1996.

[40] Karkkainen K.H., "Mean-Square Correlation as a Performance Measure for Spreading Code Families", *IEEE* 2^{nd} *Int. Symp. on Spread Spectrum Techniques and Applications (ISSSTA' 92)*, pp. 147-150, November 1992.

[41] Karkkainen K.H., "Meaning of Maximum and Mean-Square Cross-Correlation as a Performance Measure for CDMA Code Families and their Influence on System Capacity", *IEICE Transactions on Communications,* vol. E-76-B, no. 8, August 1993.

[42] Kettunen K., "Licentiate Course on Signal Processing in Communication", *VTT, Information Technology* November 1997.

[43] Li D., "A High Spectrum Efficient Multiple Access Code", *Chinese Journal of Electronics*, vol. 8, pp. 221-226, July 1999.

[44] Li D., "Scheme for Spread Spectrum Multiple Access Coding", US Patent, US 6,331,997 B1, December 2001.

[45] Lin X.D. and Chang K.H., "Optimal PN Sequence Design for Quasi-Synchronous CDMA Communication Systems", *IEEE Trans. Communications*, vol. 45, no.2, pp. 221-226, February 1997.

[46] MacWilliams F.J. and Sloane N.J.A., "The Theory of Error-Correcting Codes", ninth edition, Elsevier, Amsterdam, 1996.

[47] Massey J.L. and Uhran J.J., “Sub-Baud Coding”, *Proceedings of the Thirteenth Annual Allerton Conference on Circuit and System Theory*, pp. 539-547, October 1975.

[48] Matsufuji S. and Suehiro N., “Complex Hadamard Matrices Related to Bent Sequences”, *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 637, 1996.

[49] Milstein L.B., “Wideband Code Division Multiple Access” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 8, August 2000.

[50] Montemanni R., Moon J.N.J. and Smith D.H., “An Improved Tabu Search Algorithm for the Fixed Spectrum Frequency Assignment Problem”, *IEEE Transactions on Vehicular Technology*, VT-52(4), pp. 891-901, May 2003.

[51] Nechaev A.A., “Kerdock Code in a Cyclic Form”, *Discrete Math. Appl.*, vol. 1, no. 4, pp. 365-384, 1991, (translated from the Russian: Diskretnaya Matematika, vol. 1, no. 4, pp. 123-139, 1989.

[52] Ojanperä T. and Prasad R., “An Overview of Air Interface Multiple Access for IMT-2000/UMTS”, *IEEE Communications Magazine*, vol. 36, pp.82-95, September 1998.

[53] Oppermann I., Rapajic P. and Vucetic B.S., “Complex Valued Spread Spectrum Sequences with —Good Cross-Correlation Properties”, *in Symp. on Spread Spectrum Techniques and Applications (ISSTA '94)*, pp. 500-504, July 1994.

[54] Oppermann I. and Vucentic B.S., “Complex Spreading Sequences with a Wide Range of Correlation Properties”, *in Proc. GLOBECOM '94*, pp. 1738-1742, Nov. 1994.

[55] Oppermann I. and Vucentic B.S., “Complex Spreading Sequences with a Wide Range of Correlation Properties”, *IEEE Transactions on Communications,* vol. 45, no. 3, pp. 365-375, 1997.

[56] Pahlavan K. and Levesque A.H., “Wireless Information Networks”, *Wiley and Sons*, March, 1995.

[57] Park S.R., Song L., Yoon S. and Lee J., “A New Polyphase Sequence with Perfect Even and Good Cross-Correlation Function for DS/CDMA Systems”, *IEEE Trans. on Vehicular Technology*, vol. 51, no.5, pp. 855-865, 2002.

[58] Peterson R.L., Ziemer R.E. and Borth D.E., “Introduction to Spread Spectrum Communications”, Prentice-Hall, Inc. Inglewood Cliffs, NJ, USA, 1995.

[59] Pettit R.H., "Pulse Sequences with Good Auto-correlation Properties", *Microwave Journal* , no. 2, pp. 63-67, 1967.

[60] Pickholtz R. L., Schilling D. L. and Milstein L. B., "Theory of Spread-Spectrum Communications - A Tutorial" *IEEE Transanctions on Communications*, vol. COM 30, no. 5, pp. 855-884, May 1982.

[61] Popovic B.M., Suehiro N. and Fan P.Z., "Orthogonal Sets of Quadriphase Sequences with good Correlation Properties", *IEEE Trans. Inform. Theory*, vol. 48, no. 4, pp. 956-959, 2002.

[62] Proakis J.G., "Digital communications", McGraw Hill, New York, USA, 1995.

[63] Pursley M.B., "Performance Evaluation for Phase-Coded Spread-Spectrum Multiple-Access Communication- Part I: System Analysis", *IEEE Transactions on Communications,* vol. COM-25, no. 8, pp. 795-799, August 1977.

[64] Pursley M.B. and Sarwate D.V., "Performance Evaluation for Phase-Coded Spread-Spectrum Multiple-Access Communication- Part I: Code Sequence Analysis", *IEEE Transactions on Communications,* vol. COM-25, no. 8, pp. 800-803, August 1977.

[65] Sanusi S., Jones R.A., Perkins S. and Smith D.H., "The Application of Frequency Assignment Techniques in Spreading Code Assignment" Preprint.

[66] Sarwate D.V., "Bounds on Crosscorrelation and Autocorrelation of Sequences", *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 720-724, 1979.

[67] Sarwate D.V., "Mean-Square Correlation of Shift-Register Sequences", *Proceedings of the Institute of Electrical Engineering,* pp. 795-799, 1984.

[68] Sarwate D.V. and Pursley, M.B., "Crosscorrelation Properties of Pseudorandom and Related Sequences", *Proceeding of the IEEE*, vol. 68, pp. 593-619, 1980.

[69] Scholtz R.A., "The Origins of Spread Spectrum Communications", *IEEE Transactions on Communications*, vol. 30, pp. 882-854, May 1982.

[70] Selmer E.S., "Linear Recursive Relations over Finite Fields", Dept. Math., University of Bergen, Bergen, Norway, 1996.

[71] Sidelnikov V.M., "On Mutual Correlation of Sequences", *Soviet Math Dokl.,* vol. 12, pp. 197-201, 1971.

[72] Simon M.K., Omura J.k., Scholtz R.A. and Levitt B.K., "Spread Spectrum Communications Handbook", Revised edition, McGraw-Hill, Inc. USA, 1994.

[73] Smith D.H., Allen S., Dunkin N. and Hurley S., "Frequency Assignment Problems: Benchmarks and Lower Bounds", Division of Mathematics and Computing, University of Glamorgan, Pontypridd, Technical report, UG-M-98-1, 1998.

[74] Smith D.H., Allen S.M., Hurley S. and Watkins W.J., "Frequency Assignment: Methods and Algorithms, *Proceedings of NATO IST Symposium on Frequency Assignment, Sharing and Conservation in Systems (Aerospace)*, Aalborg, Denmark, 5-7 October 1998, NATO RTO-MP-13, ppK1-K18, January 1999.

[75] Smith D.H., Hurley S. and Thiel S.U., " FASoft: A system for Discrete Channel Frequency Assignment", *Radio Science*, 32, 5, pp. 1921-1939, 1997.

[76] Smith D.H. and Perkins S., "Applications of Coding Theory in Mobile Radio Communications", *Mathematics Today,* vol. 37, pp. 84-87, June 2001.

[77] Smith D.H., Taplin R. K. and Hurley S., "Frequency Assignment with Complex Co-Site Constraints" *IEEE Transactions on Electromagnetic Compatibility*, vol. 43, no. 2, pp. 210-218, May 2001.

[78] Stańczak S., Boche H. and Haardt M., "Are LAS Codes a Miracle?" *IEEE Global Communications Conference (GLOBECOM)*, BWS05-4, pp. 589-593, Nov. 2001.

[79] Tang X.H. and Fan P.Z., "Bounds on Aperiodic and Odd Correlations of Spreading Sequences with Low and Zero Correlation Zone" *Electronic Letters,* vol. 37, no.19, pp. 1201-1203, Sep. 2001.

[80] Tang X.H., Fan P.Z. and Matsufuji S., "Lower Bounds on Correlation of Spreading Sequence Set with Low or Zero Correlation Zone", *Electronic Letters,* vol. 36, no. 6, pp.551-552, 2000.

[81] Viterbi A.J., "CDMA: Principles of Spread Spectrum Communication", Addison-Wesley Longman Inc. 1995.

[82] Viterbi A.M. and Viterbi A.J., "Erlang Capacity of a Power Controlled CDMA System", *IEEE J. Select. Areas Commun.*, vol. 11, no. 6, pp. 892-900, 1993.

[83] Welch L.R., "Lower Bounds on the Maximum Crosscorrelation of Signals", *IEEE Trans. Commun.* vol. Com-22, pp.997-1004, 1974.

[84] Wu Y., Zhang Q., Zhu W. and Kung S.-Y., "Spreading code assignment in an Ad-hoc DS-CDMA Wireless Network", *International Conference on Communications (ICC)*, New York, NY, Apr. 2002.

[85] Xiaoxu G., Jie C. and Naoki S., " Code Aquisition of ZCZ-CDMA Systems based on Complete Complementary Codes" *Wireless Commun. Mobile Comp.* vol. 3, pp. 585-595, 2003.

[86] Yao K., "Error Probability of Asynchronous Spread Spectrum Multiple Access Communication Systems", *IEEE Trans. on Communications*, vol. COM-25. no. 8, August 1977.

[87] Zeng M., Annamalai A. and Bhargava V.K., "Recent Advances in Cellular Wireless Communications", *IEEE Communications Magazine*, vol. 9, pp. 128-138, September 1999.

[88] Zierler N., "Linear Recurring Sequences", *J. Soc. Industrial and Applied Mathematics*, vol. 7, pp. 31-48, 1959.

[89] Zierler N. and Mills W.H., "Products of Linear Recurring Sequences", *J. Algebra*, vol. 27, pp. 147-157, 1973.