A Chip-Level IS95-Compliant Cellular CDMA Simulator: Design, Implementation, and Analysis*

A. Mark Earnshaw & Steven D. Blostein

Image Processing and Communications Lab
Department of Electrical and Computer Engineering
Queen's University
Kingston, Ontario, Canada K7L 3N6

earnshaw@ee.queensu.ca
sdb@ee.queensu.ca

http://ipcl.ee.queensu.ca

Phone: (613)-545-6561
Fax: (613)-545-6615

Abstract

Cell capacity in a CDMA communication system can be increased through the use of base station antenna arrays and digital beamforming. We have therefore developed a comprehensive chip-level simulator of the reverse link component of an IS95-compliant cellular system. This level of detail is necessary to validate our simplifying models for imperfect beamforming and imperfect power control. This paper explains in detail the implementation of the entire reverse link simulation process. The various techniques available for power control and beamforming are then described, together with methods for estimating maximum cell capacity. An error analysis which was used to validate the accuracy of the simulator is included. Finally, sample simulation results are presented, our ongoing research is discussed, and some concluding remarks are made.

* This research has been supported by the Canadian Institute for Telecommunications Research under the NCE program of the Government of Canada.

Contents

List of Figures

List of Tables

List of Symbols

α_i	relative phase angle of i^{th} element of array response vector element
β_k	signal envelope for k^{th} mobile at the receiver
γ_i	physical angular location of i^{th} element in a circular antenna array
ϵ	path loss exponent
η_k	signal attenuation over the transmission channel for mobile k
Θ_k	phase shift of signal for mobile k after beamforming with uncorrected weights has taken place
θ	direction of propagation of plane wave
θ_{I_k}	in-phase angle used for generating a ±1 component in the signal
θ_{Q_k}	quadrature angle used for generating a ±1 component in the signal
λ_c	wavelength of the carrier frequency
μ_c	mean PN chip correlation value
μ_F	mean Walsh function correlation value when signal is correlated with the correct Walsh function
$\mu_{\overline{F}}$	mean Walsh function correlation value when signal is correlated with the incorrect Walsh function
μ_{nc}	mean of thermal noise component in correlation value over one PN chip
μ_R	mean of a standard Rayleigh distribution
μ_w	mean Walsh chip correlation value
ν	factor by which transmitted power is reduced when speech is not active
ρ_k	path loss attenuation factor for mobile k
σ_c^2	variance of the PN chip correlation value
σ_F^2	variance of the Walsh function correlation value
σ_n^2	variance of the background thermal noise
σ_{nc}^2	variance of the thermal noise component in the PN chip correlation value
σ_S^2	variance of the shadowing random variable
σ_w^2	variance of the Walsh chip correlation value
τ_{kp}	time delay of k^{th} mobile's signal relative to p^{th} mobile's signal
Υ	sum of squared sin and cos terms when beamforming is used
Φ_p	phase shift which must be applied to the Wiener beamforming weights of the p^{th} mobile in order to obtain the desired solution and achieve phase synchronization

ϕ_i	relative phase angle of i^{th} entry of beamforming weight vector
φ	angle formed within a circular antenna array by two adjacent elements
ψ	mean percentage of time that a user is actually talking (voice activity factor)
$\mathbf{a}_k$	array response vector for k^{th} mobile
B	bandwidth of spread signal
B_D	number of data bits per frame
B_T	total number of bits per frame
B_W	number of encoded bits per Walsh function
b_i	number of known bits at i^{th} Walsh function position in interleaved Viterbi decoding
$C_{C_p}[r]$	PN chip correlation for mobile p
$C_{F_p}[r]$	Walsh function correlation for mobile p
$C_{I_p}[r]$	in-phase signal correlated over one PN chip for p^{th} mobile
$C_{Q_p}[r]$	quadrature signal correlated over one PN chip for p^{th} mobile
$C_{W_p}[r]$	Walsh chip correlation for mobile p
$\mathcal{C}_F$	correlation value for a Walsh function position within a surviving path
$c_{I_k}[r]$	in-phase PN code sequence for k^{th} mobile
$c_{Q_k}[r]$	quadrature PN code sequence for k^{th} mobile
D_i	distance along direction of wave propagation between center of array and i^{th} antenna element
D_A	spacing between adjacent elements in an antenna array
D_{I_k}	demodulated in-phase signal for k^{th} mobile
D_{Q_k}	demodulated quadrature signal for k^{th} mobile
d_k	distance from k^{th} mobile to base station
E_b/N_0	ratio of bit energy to noise and interference density
f_c	carrier frequency
F	number of frames per second
F_F	number of Walsh functions per frame
F_W	number of Walsh chips per frame
G	spread spectrum gain factor
$\mathbf{g}$	generator vector for convolutional encoding
H	matrix used to recursively build up the matrix of Walsh function chip values
I	identity matrix

I_p	random variable specifying the interference to mobile p's signal due to other mobiles
L_{PC}	length of one power control period in terms of PN chips
$\mathrm{M_I}$	shift register XOR mask for generating in-phase short code PN sequence
$\mathrm{M_L}$	shift register XOR mask for generating long code PN sequence
$\mathrm{M_O}$	shift register output mask for long code PN sequence
$\mathcal{M}_P$	surviving path metric for Viterbi decoding
$\mathrm{M_Q}$	shift register XOR mask for generating quadrature short code PN sequence
$\mathcal{M}_W$	metric value for a Walsh function in Viterbi decoding
N_A	number of antenna elements
N_B	number of bits per Walsh function
N_F	number of Walsh chips in one Walsh function sequence
N_M	number of mobiles
N_{PC}	number of power control periods in one frame
N_W	number of PN chips in one Walsh chip
$\mathbf{n}$	thermal background noise vector
n	noise
$\mathcal{P}$	surviving path
P	power
$P_{I_{kp}}$	received power of in-phase signal for mobile k when synchronized with mobile p
$P_{Q_{kp}}$	received power of quadrature signal for mobile k when synchronized with mobile p
P_{I_k}	interference and noise power for mobile k
P_{M_k}	received power from mobile k
P_{T_k}	signal power at transmitter for k^{th} mobile
Q_p	noise and interference covariance matrix used for estimating Wiener beamforming weights for the p^{th} mobile
R_k	Rayleigh fading multiplicative attenuation for mobile k
R_A	radius of the antenna array
R_B	data bit rate (number of data bits per second)
R_C	radius of cell
$\mathrm{R}_{\mathbf{n}}$	autocorrelation of background noise
$\mathrm{R}_{\mathbf{xx}}$	correlation/covariance matrix of received signal
$\mathrm{R}_{\mathbf{zz}}$	correlation/covariance matrix of correlated signal
r	number of bits in shift register used for generating PN sequences

S_k	shadowing multiplicative attenuation for k^{th} mobile
T_c	period of one PN chip
t	current time
$\mathbf{v}$	unit vector specifying direction of wave propagation
$\mathcal{W}_p$	Wiener beamforming weights for the p^{th} mobile
$W_n[\cdot]$	sequence of Walsh chips corresponding to the n^{th} Walsh function
$\mathbf{w}_p$	vector of beamforming weights for mobile p
w_i	i^{th} entry of beamforming weight vector
$w_k(t)$	current Walsh chip value for the k^{th} mobile
X_{I_k}	binary random variable (± 1) representing in-phase multiplied PN code values
X_{Q_k}	binary random variable (± 1) representing quadrature multiplied PN code values
$\mathbf{x}$	received signal vector
x_{I_k}	in-phase component of signal for k^{th} mobile
x_{Q_k}	quadrature component of signal for k^{th} mobile
Y	uniformly distributed random variable representing a phase angle in the interval $[0, 2\pi)$

1 Introduction

CDMA (Code Division Multiple Access) is one technology which has been proposed for the next generation of digital cellular phones in North America. In particular, Qualcomm has prepared the IS95 specification [22] which details the implementation of a CDMA cellular communication system. CDMA promises to improve cell capacity by increasing the number of users within a given bandwidth as compared with other multiple access techniques, and various studies have been performed to estimate the capacity of such a system [4] [5] [7] [23]. However, it is possible to further increase cell capacity through the use of digital beamforming and base station antenna arrays [10] [12] [17] [18] [25] [26].

We have developed a complete IS95 compliant chip-level simulator for the reverse link of a CDMA cellular communication system. This level of detail permits more realistic system emulations to be performed, unlike previous simulations which required a number of rough approximations. We are thus able to estimate capacity figures via detailed simulations and then compare these results with predicted values. In addition, the chip-level signal values are used to generate data necessary for the beamforming process, so that the effects of beamforming on cell capacity can be investigated. We intend to streamline the capacity estimation process by developing our own simplifying models which can be verified against the chip-level simulations. It should also be noted that the simulation program is sufficiently flexible to be configured for modeling services other than voice such as video and computer data. Thus, we are not just limited to investigating the IS95 specification.

This paper focuses primarily on the simulation methodology and describes in detail how each stage of the reverse link is simulated. Special consideration is given to the topics of power control and digital beamforming which are the two areas that this simulator was designed to investigate. Two methods for estimating cell capacity are discussed: 1) a more detailed chip-level approach, and 2) a more efficient simplified technique based on the power control information. An error analysis is used to validate the accuracy of the simulation process. We then present some preliminary simulation results and note the effects of various factors such as imperfect power control, encoded bit interleaving, CDMA processing gain, and the number of base station antennas on the total cell capacity. Finally, we describe our ongoing research in this area and summarize the work presented here.

2 Background & Motivation

In this section, we briefly review other research which has been conducted to analyze system performance in CDMA systems. In addition, we justify our creation of a chip-level simulation program for achieving the same purpose.

Gilhousen *et al* [4] performed a theoretical analysis of a CDMA system and derived the expected cell capacities for both the forward and reverse links. They considered a simple sectorization approach using directional antennas in order to increase the number of users, but did not investigate true beamforming. In addition, they assumed perfect power control within each cell. This paper is mainly a theoretical analysis and makes a number of simplifying assumptions which may affect the final results.

Jalali and Mermelstein [5] investigated the effects of imperfect power control and multi-path diversity on CDMA capacity. Their data coding technique followed the IS95 standard which we have also investigated. Their simulations did not consider the entire coding process, but only went as far as the Walsh function level (see Section 3.2.1 for more information on the complete coding process). In addition, it seems as though their simulator assumed perfect synchronization between all of the mobile signals received at the base station, which may or may not be an acceptable simplification.

Stüber and Kchao investigated the single cell [7] and multi-cell [23] situations for a standard CDMA system with sectoring (three sectors per cell). They analyzed the uplink and downlink error probabilities and considered the effects of factors such as different numbers of propagation paths, fading, and power control. All of their results were derived from equations and did not involve any simulation work.

Our beamforming research is based primarily on work performed by Naguib and Paulraj. However, most of their work focuses on performing probabilistic analyses and using the resulting equations to estimate system performance for different numbers of antenna elements. In [12], they considered a standard CDMA system (no IS95) with a Rake receiver. A 2D Rake receiver was considered in [11] with the assumption of perfect power control.

Only the forward link situation was investigated in [18] where it was necessary to assume that feedback information from the mobiles was available in order to perform beamforming at the base station transmitter. We are currently only studying the reverse link, although it is our intention to investigate the forward link as well. Equations were derived to estimate the outage probability for a specified number of sensors.

The reverse link with a Rake receiver and M-ary orthogonal modulation was investigated in [14]. This encoding scheme is consistent with IS95. However, perfect power control was assumed. Capacity estimations consisted of a probabilistic error analysis and simulations were only used to estimate probability distributions. In [13], Naguib and Paulraj investigated improvements in terms of capacity, range, and coverage area due to beamforming. Sectorized cells were considered, and the cases of beamforming and no beamforming were compared.

A standard CDMA spreading technique that does not match the IS95 encoding scheme was considered in [17]. This investigated the same beamforming method discussed in the other papers for the uplink and downlink and performed a theoretical error analysis of the system. Outage probabilities were calculated via equations.

In [16], the performance of an IS95 system with a sectorized uniform linear antenna array and beam-steered RAKE architecture was analyzed by extending the results in [6]. Diversity combining was used to exploit delay spread, and the estimation of the array response of multiple paths per mobile was required. Due to the delay spread arising from statistical variation of unresolvable paths, the multiple access interference was approximated by white noise. Errors in closed-loop power control were modeled by an equivalent Doppler shift. Our paper differs from [16] in that we are presenting a complete chip-level simulation using a much lower complexity single-path beamforming architecture. Due to the low long-code processing gain, we recognize that array response estimation for even a single path is nontrivial in an IS95-compatible system. However, the promising results in [16] indicate that a further capacity gain may be realized through diversity combining.

In all of Naguib and Paulraj's work, they focused on analyzing system performance via probabilistic equations that were derived using simplifying assumptions. Simulations appear to have been used only to estimate probability distributions for random variables that could not be derived analytically. Our goal in developing a detailed chip-level simulation of a CDMA system was to determine cell capacity in various situations using a highly realistic emulation of a physical system. This would allow us to justify any simplifying assumptions made in our own performance analyses by comparing predicted results with those observed from a system simulation. In addition, the chip-level data is required for testing out our own approaches to the efficient estimation of beamforming weights.

3 Simulation Methodology

Currently, only the reverse link (mobile to base) of the communication system has been studied. It is not realistically practical to implement beamforming at the mobiles due to the necessary antenna arrays and processing requirements. Therefore, beamforming is used only at the base station, and the reverse link situation is more suited to this task since the received signal data can be processed to obtain the required beamforming information. In addition, cell capacity is usually limited by the reverse link. However, beamforming at the base transmitters can also be implemented on the forward link using feedback information from the mobiles [3].

We now present a detailed description of the simulation process for the reverse link which considers a number of modeling issues. The items which are covered in this section include the cell layout, encoding of data for transmission, transmission channel parameters, decoding of received data, and call initialization.

3.1 Cell Layout

Cell shapes in the simulator can be square, hexagonal, or circular. The first two shapes are suitable for multicell simulations since it is possible to tile the plane with either squares or hexagons. However, circular cells are best if restricted to a single cell environment since it is not possible to provide complete coverage over a specified area without having the cells overlap.

The cell radius R_C is specified as a parameter and is defined in Figure 1 for both hexagons and squares. A typical value for R_C would be 500 m. All of the cells are assumed to be of the same size.

If a multicell environment is being simulated, the number of tiers may also be specified. This indicates the number of cell layers which surround the central cell. Figure 2 illustrates two-tier designs for both hexagonal and square cells. A multi-tier simulation is more realistic since it also considers intercell interference. At the present time, however, our simulations are usually limited to just a single cell, due to the computational expense and memory requirements.

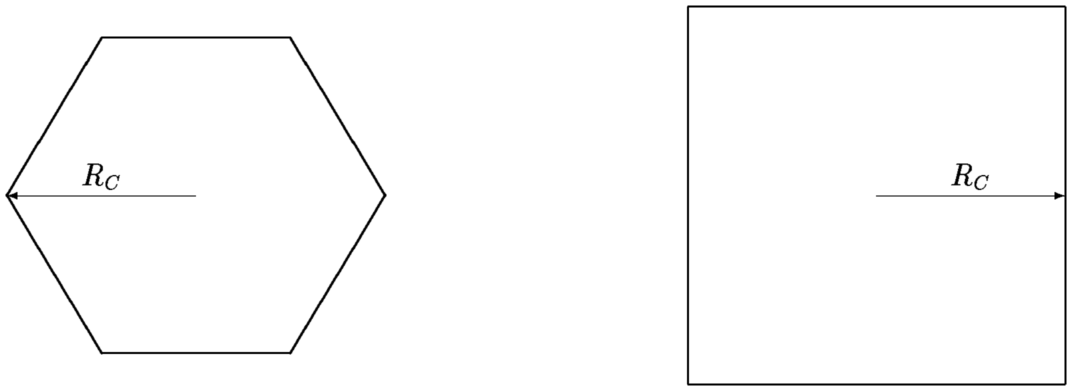

Figure 1: Cell radii for hexagonal and square cells

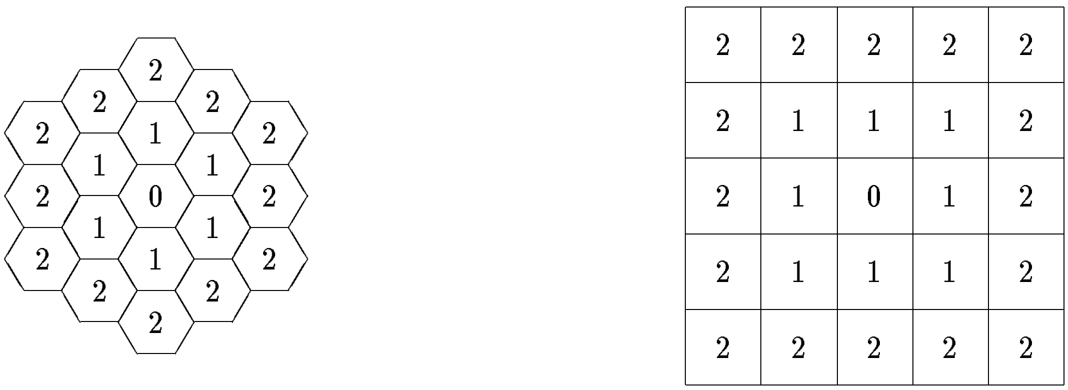

Figure 2: Sample cell layouts with two-tier designs

3.2 Encoding Reverse Link Data

The source data is assumed to consist of sequences of random bits representing compressed voice signals which are generated as required. In order to provide protection against errors, the data bits are sent through a multi-level encoding process which is described in the following section.

3.2.1 Encoding Data Bits

It is assumed that acceptable voice quality data can be transmitted at a bit rate R_B (9600 bps) which includes overhead. These bits are divided into frames with F (50) frames per second and B_T (192) total bits per frame. The first B_D (184) bits represent actual data, and the remainder are tail bits which are all zero.

The frame bits are first passed through a rate 1/3 convolutional encoder with a constraint length of 9. This produces 576 encoded bits per frame. The generator vectors for the IS95 encoder are:

$$\mathbf{g}_0 = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (1)$$

$$\mathbf{g}_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (2)$$

$$\mathbf{g}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

A diagram of the corresponding encoder is shown in Figure 3. New data bits are shifted in from the left. The 1's in the generator $\mathbf{g}_i$ indicate which of the shift register bits are XORed together to produce the corresponding output bit c_i.

The next stage involves block interleaving the encoded bits using a 32×18 interleaver matrix. Bits are written into successive columns of this array and then read out row by row. Although the interleaving step is part of the IS95 specification, it is an optional process in the simulator so that its effects on system performance can be studied.

The resulting bits are then grouped into sets of N_B (6) to form binary numbers which select one of 2^{N_B} (64) different Walsh functions. This yields a total of F_F (96) Walsh functions per frame.

Each Walsh function has a sequence of 64 chips associated with it which results in a total of F_W (6144) Walsh chips per frame. These sequences are constructed so that a given sequence is orthogonal to each of the other 63 Walsh chip sequences. The Walsh function

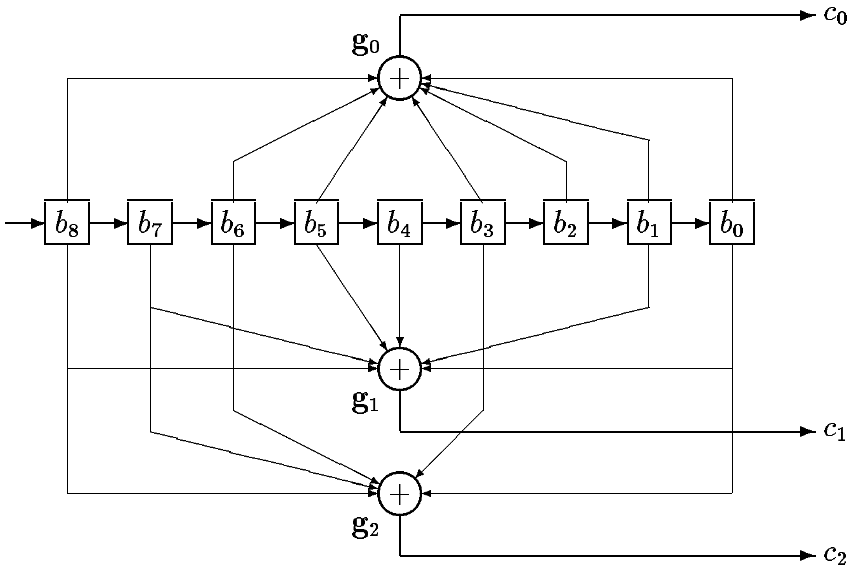

Figure 3: IS95 convolutional encoder

matrix is constructed by the following recursive procedure.

$$\mathrm{H}_1 = \begin{bmatrix} 0 \end{bmatrix} \tag{4}$$

$$\mathrm{H}_{2N} = \begin{bmatrix} \mathrm{H}_N & \mathrm{H}_N \\ \mathrm{H}_N & \overline{\mathrm{H}}_N \end{bmatrix} \tag{5}$$

$\overline{\mathrm{H}}_N$ represents the complement of the matrix H_N. Recursion is continued until a 64×64 matrix is obtained. The i^{th} row of this matrix gives the sequence of Walsh chips for the i^{th} Walsh function.

Finally, pseudo-noise (PN) sequences are used to spread the signal by a factor of 4. A long code with a period of $2^{42} - 1$ is first used to spread the Walsh chips. The resulting signal is then split into in-phase and quadrature components with each component being further encoded by a different short code sequence, each with a period of 2^{15} chips. The long and short code chips are of the same length. Further information on the generation of the PN sequences is contained in the following section.

The in-phase and quadrature components are modulated using BPSK (Binary Phase Shift Keying) with a carrier frequency of f_c Hz. A typical value for f_c would be 2 GHz.

3.2.2 Generating PN Code Sequences

As mentioned earlier, there are three pseudo-noise (PN) sequences used for encoding data. The long code has a period of $2^{42}-1$ and is used to separate signals originating from different mobiles. Two short code sequences with periods of 2^{15} chips are used to encode the in-phase and quadrature components of the same signal. The short codes are intended to enhance the separation of the in-phase and quadrature components.

General information on producing PN sequences can be found in [28] and is summarized below for both the short and long code situations.

Short Code PN Sequences

The generator polynomials for the in-phase and quadrature short code sequences are:

$$P_I(x) = x^{15} + x^{13} + x^9 + x^8 + x^7 + x^5 + 1 \qquad (6)$$

$$P_Q(x) = x^{15} + x^{12} + x^{11} + x^{10} + x^6 + x^5 + x^4 + x^3 + 1 \qquad (7)$$

These can be translated into the following recursive relationships as illustrated in Figure 4.

$$i_n = i_{n-15} \oplus i_{n-10} \oplus i_{n-8} \oplus i_{n-7} \oplus i_{n-6} \oplus i_{n-2} \qquad (8)$$

$$q_n = q_{n-15} \oplus q_{n-12} \oplus q_{n-11} \oplus q_{n-10} \oplus q_{n-9} \oplus q_{n-5} \oplus q_{n-4} \oplus q_{n-3} \qquad (9)$$

The corresponding in-phase and quadrature short code XOR masks are:

$$\mathrm{M_I} = (g_{15} \cdots g_1) = (100\ 0010\ 1110\ 0010)_2 = (42\mathrm{E}2)_{16} \qquad (10)$$

$$\mathrm{M_Q} = (g_{15} \cdots g_1) = (100\ 1111\ 0001\ 1100)_2 = (4\mathrm{F}1\mathrm{C})_{16} \qquad (11)$$

A PN sequence with a shift register of length r bits will normally have a period of $2^r - 1$ different register states since the all-zero state is not achievable. However, the short code period in this case is defined to be 2^r where $r = 15$. The extra state is obtained by converting the unique sequence of 14 consecutive zeros in each bit cycle to a sequence of 15 zeros. It is necessary to recognize this particular shift register state as the short code bits are generated and to insert the extra zero bit at the appropriate location.

Since the size of the short code shift register is relatively small, a new bit (b_n) is generated from the current shift register value by ANDing the contents with the appropriate short code mask ($\mathrm{M_I}$ or $\mathrm{M_Q}$), and then using the resulting value to index into a precalculated array which specifies whether the new bit should be a 0 or a 1. This eliminates the need to XOR individual bits together and thereby increases the processing speed.

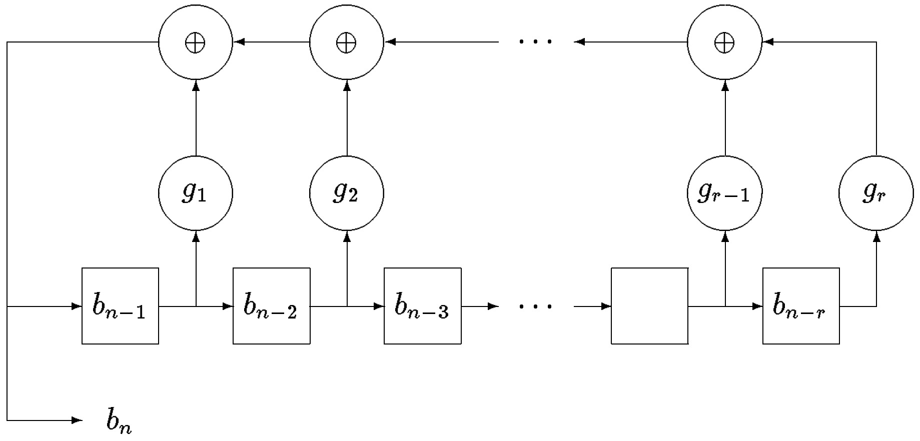

Figure 4: PN code shift register design (short code)

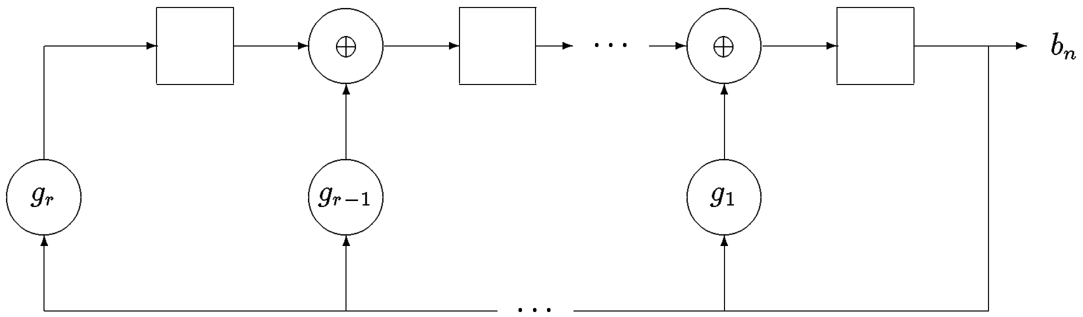

Figure 5: PN code shift register design (long code)

If $R_I(n)$ represents the in-phase shift register contents at time n, then the new register contents can be calculated as:

$$R_I(n+1) = (R_I(n) \ll 1) + \mathrm{XOR}[R_I(n) \cdot \mathrm{M_I}] \tag{12}$$

where $\ll$ indicates that the value should be shifted to the left by the specified number of bits, and XOR[$\cdot$] specifies that all of the bits in the given value should be XORed together to produce a single bit value.

Since these PN sequences have a relatively short period ($2^{15} = 32768$ chips), the entire sequence is precalculated at the beginning of a simulation and then the current position within the sequence is maintained for each mobile. This eliminates the need to compute the short code chip streams repeatedly which would be much more computationally expensive.

Long Code PN Sequence

The characteristic polynomial for the long code is:

$$\begin{aligned} P(x) = {} & x^{42} + x^{35} + x^{33} + x^{31} + x^{27} + x^{26} + x^{25} + x^{22} + x^{21} + x^{19} \\ & + x^{18} + x^{17} + x^{16} + x^{10} + x^{7} + x^{6} + x^{5} + x^{3} + x^{2} + x^{1} + 1 \end{aligned} \tag{13}$$

The corresponding XOR mask value is:

$$\mathrm{M_L} = (g_1 \ldots g_{42}) \tag{14}$$

$$= (00\ 0000\ 0101\ 0100\ 0111\ 0011\ 0111\ 1000\ 0010\ 0111\ 0111)_2 \tag{15}$$

$$= (547378277)_{16} \tag{16}$$

Due to the larger size of the required shift register, this code sequence is more easily implemented using the design shown in Figure 5. If the feedback bit, b_n, is zero, then the register can simply be shifted by one bit and no other action is required. Conversely, if the feedback bit is one, the long code mask ($\mathrm{M_L}$) is XORed with the current register contents before shifting takes place.

Bits are not taken from the shift register for the long code in the same manner that they are for the short code. That is, b_n is not used as the output bit in the PN sequence. Instead, the current contents of the long code shift register are ANDed with an output mask value $\mathrm{M_O}$ and then the bits of the resulting binary value are XORed together to produce either a 0 or 1. This value is taken to be the current chip in the long code sequence.

The upper 10 bits of the output mask are defined to have the value $(318)_{16}$. The lower 32 bits are formed by the unique ESN (Electronic Serial Number) of each mobile. For simulation purposes, ESN values are simply assigned at random. Technically, the bits corresponding to the ESN should undergo a prespecified permutation when creating M_O since this compensates for mobiles with consecutive serial numbers operating in the same area. However, this step is ignored for simulations since the ESN values are completely random in this situation.

Rather than actually XORing the masked bits together to produce the output bit, the binary value resulting from the masking operation is broken down into three parts: the uppermost 10 bits, the middle 16 bits, and the lowest 16 bits. In a manner similar to that used for the short code, these binary numbers are used to index into a precalculated array which specifies the number of ones contained in the binary representation of each index. The three resulting values are then summed and evaluated mod 2 to yield the required output chip. This approach is much more computationally efficient than XORing individual bits together and produces the same result.

The long code shift register for each mobile is initialized with an arbitrary random value at the beginning of each simulation. This essentially guarantees that all of the mobiles will have different phases in the basic long code PN sequence, although the output masking should generate different chip sequences for different mobiles, even if their shift registers begin with the same initial value.

The period of the long code is $2^{42} - 1$ chips. At a spreading frequency of 1.2288 MHz, one cycle of the long code will require approximately 41.43 days to complete.

3.3 Transmission Channel Parameters

The following section details how the reverse link transmission channel between the mobiles and base station is modeled. This mainly involves quantities specifying signal attenuation over the transmission channel such as path loss, shadowing, and Rayleigh fading. In addition, the array response vector for a circular antenna array is derived.

3.3.1 Path Loss

The amount of power in a signal falls off as a function of the distance from the transmitter. A generally accepted model of this path loss is:

$$\rho_k = d_k^{-\epsilon} \tag{17}$$

where d_k is the distance between the k^{th} transmitter and the receiver, and ϵ is the path loss exponent. ϵ can range between 2 and 5 depending upon the local environment, and a commonly used value for testing purposes is 4.

Note that ρ_k dictates how the power of the signal falls off as a function of distance. Therefore, the amplitude of the signal envelope should be multiplied by $\sqrt{\rho_k}$.

The path loss quantity remains constant for the duration of a call since the mobiles are currently assumed to be stationary. When the simulation program is enhanced to incorporate moving mobiles, it will be necessary to update ρ_k dynamically.

3.3.2 Shadowing

Shadowing is caused by various obstacles such as trees and buildings in the transmission path of a mobile. As a mobile moves around, the degree of shadowing will vary. However, in our simulations this quantity currently remains constant for the duration of a call since the mobiles are assumed to be stationary. A commonly accepted model is to use a lognormal random variable such as:

$$S_k = 10^{\xi/10} \tag{18}$$

where ξ has a $N(0, \sigma_S^2)$ distribution. The value typically used for σ_S is 8 dB.

A new value of S_k is chosen for each mobile at the start of each call by that mobile, and S_k currently remains constant for the duration of each call.

3.3.3 Rayleigh Fading

Since the mobiles in the simulation program are currently assumed to be stationary, there should theoretically be no source of dynamic fading due to time-varying multipath propagation. However, Rayleigh amplitude fading is modeled to represent the worst-case scenario. This multiplicative quantity is calculated as:

$$R_k = \sqrt{(G_1)^2 + (G_2)^2} \tag{19}$$

where G_1 and G_2 are $N(0, 1)$ Gaussian random variables.

A new Rayleigh fading parameter is calculated for each mobile every 48 PN chips. This roughly corresponds to the time required to transmit the equivalent of one data bit (46 PN chips), but was rounded up since 48 divides evenly into the power control period of 1536 PN chips, thereby simplifying the simulation process.

3.3.4 Overall Channel Attenuation

The transmission power and, hence, the signal amplitude or envelope originating from each mobile is known. As the signal travels to the base station, its strength is attenuated by the previously discussed factors. Thus, the envelope is multiplied by the following factors to determine its magnitude as observed at the base station.

$$\eta_k = \sqrt{\rho_k S_k} R_k \tag{20}$$

The path loss and shadowing terms affect the power of a signal, so it is necessary to take the square root of these terms when considering the signal amplitude. Currently, R_k is the only one of these parameters which varies during the duration of a call.

3.3.5 Array Response Vector Calculation

If a multi-element antenna array is used at the base station, it is necessary to determine the array response vector $\mathbf{a}$ for a signal arriving from a given direction. This vector specifies the relative phases of the signal at each of the antenna elements. Each entry in $\mathbf{a}$ is a unit magnitude complex number whose phase specifies the relative phase of the corresponding antenna element.

The antenna array is assumed to be in a circular form with N_A representing an arbitrary number of elements which are omnidirectional and have equal gains. Adjacent elements are assumed to have one-half wavelength ($\lambda_c/2$) spacing between them. This analysis can be easily adapted for a linear array, although circular arrays were chosen since they are less directional in nature. In terms of antenna arrangements other than circular and linear designs, Liang and Paulraj [8] investigated several different array topologies and the resulting improvements in coverage.

Refer to Figure 6 for the definition of quantities used in this derivation. If the inter-element spacing is $D_A = \lambda_c/2$, then the corresponding radius of the circular antenna array

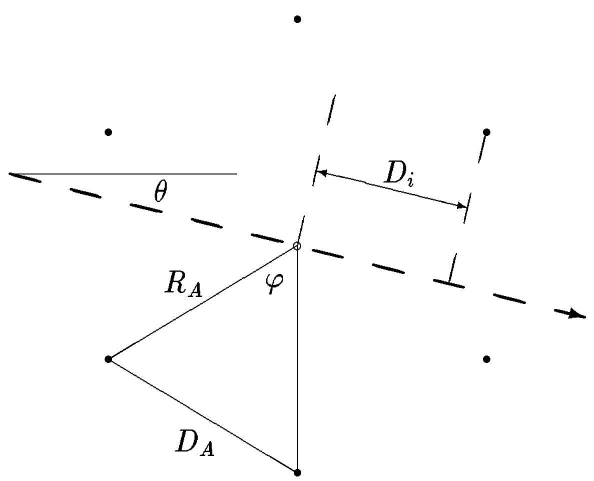

Figure 6: Array response vector calculation for a multi-element antenna array

can be calculated as follows [9, pp. 65].

$$\varphi = \frac{2\pi}{N_A} \tag{21}$$

$$\sin(\varphi/2) = \frac{D_A/2}{R_A} \tag{22}$$

$$R_A = \frac{D_A/2}{\sin(\varphi/2)} = \frac{D_A}{2\sin(\pi/N_A)} = \frac{\lambda_c}{4\sin(\pi/N_A)} \tag{23}$$

Let γ_i represent the angle at which the i^{th} $(0 \le i \le N_A - 1)$ element is located on the circle. Without loss of generality, it can be assumed that $\gamma_0 = 0$. Then:

$$\gamma_i = i\left(\frac{2\pi}{N_A}\right) \tag{24}$$

The position of the i^{th} antenna element in Cartesian coordinates can thus be written as:

$$x_i = R_A \cos(\gamma_i) \tag{25}$$

$$y_i = R_A \sin(\gamma_i) \tag{26}$$

Now assume that a plane wave with frequency f_c and propagating in the direction θ is impinging upon the antenna array as shown by the dashed arrow in Figure 6. A unit vector in the direction of wave propagation can be written as:

$$\mathbf{v} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \end{bmatrix} \tag{27}$$

We now simply need to take the dot product of this vector with the Cartesian coordinates of each antenna element in order to find D_i. This represents the distance along the direction of wave propagation between the center of the array and the i^{th} antenna element.

$$\begin{aligned} D_i &= x_i \cos(\theta) + y_i \sin(\theta) && (28) \\ &= R_A \cos(\gamma_i)\cos(\theta) + R_A \sin(\gamma_i)\sin(\theta) && (29) \\ &= R_A \cos(\theta - \gamma_i) && (30) \end{aligned}$$

Let α_i represent the relative phase delay for the i^{th} element. The center of the circle is taken to be the reference point with a phase delay of zero. Consequently, the phase at the i^{th} element can be calculated as:

$$\begin{aligned} \alpha_i &= 2\pi \left(\frac{D_i}{\lambda_c}\right) && (31) \\ &= \frac{2\pi R_A \cos(\theta - \gamma_i)}{\lambda_c} && (32) \\ &= \frac{2\pi \cos(\theta - \gamma_i)}{\lambda_c}\left(\frac{\lambda_c}{4\sin(\pi/N_A)}\right) && (33) \\ &= \frac{\pi \cos(\theta - \gamma_i)}{2\sin(\pi/N_A)} && (34) \end{aligned}$$

It is interesting to note that the relative phase is independent of the carrier frequency since the inter-element spacing is a function of this frequency value.

3.4 Decoding Reverse Link Data

There are a number of processing steps to be simulated in the base station receiver in order to recover the original data bits. These include beamforming, demodulation, correlation, and Viterbi decoding.

3.4.1 Beamforming

In a simulation, the true array response vector $\mathbf{a}_k$ can be determined for each of the mobiles in the system as shown in Section 3.3.5. For a multi-element base station array, the received signal vector $\mathbf{x}(t)$ can be calculated as:

$$\mathbf{x}(t) = \sum_{k=1}^{N_M} \beta_k(t)\mathbf{a}_k + \mathbf{n}(t) \qquad (35)$$

where $\beta_k(t)$ represents the signal strength from the k^{th} mobile and $\mathbf{n}(t)$ represents wide-sense stationary zero-mean Gaussian white noise with autocorrelation:

$$\mathrm{R}_{\mathbf{n}}(\tau) = \begin{cases} \sigma_n^2 \mathrm{I} & \tau = 0 \\ 0 & \tau \neq 0 \end{cases} \tag{36}$$

It is necessary to reduce the complex vector $\mathbf{x}$ to a scalar quantity that can be used in the remainder of the decoding process. This can be accomplished by taking a weighted sum of the elements of $\mathbf{x}$.

$$x_p = \mathbf{w}_p^* \mathbf{x} \tag{37}$$

Each entry in the beamforming weight vector $\mathbf{w}_p$ induces a phase shift in the corresponding entry of the signal vector $\mathbf{x}$ and may also scale that quantity. If $\mathbf{w}_p = \mathbf{a}_p$, then the phase shifts introduced by the antenna array and the beamforming process will cancel each other out for mobile p. However, if the signal from another mobile with a different array response vector is considered, the entries from each antenna element will not be in phase with each other when they are summed in equation (37). Thus, the interference from other mobiles will be greatly reduced by the beamforming process.

If the interference is assumed to be white, which will be the case for a large number of mobiles, then the optimal choice of beamforming weights for the p^{th} mobile is to let $\mathbf{w}_p$ equal the true array response vector $\mathbf{a}_p$. However, in reality $\mathbf{a}_p$ is not known exactly so it is necessary to estimate suitable values for $\mathbf{w}_p$. Section 5 describes the selection of beamforming weights in more detail.

3.4.2 Demodulation

After taking the inner product of the signal vector from the antenna array with the vector of beamforming weights, the result is a scalar complex quantity x_p as shown in (37) which can then be demodulated. It is simpler to consider the demodulation of the signal from a single antenna element first and then to generalize this result to a multi-element antenna array.

Single Element Demodulation

Consider the BPSK modulated in-phase and quadrature signal components originating from the k^{th} mobile:

$$x_{I_k}(t) = \cos(2\pi f_c t + \theta_{I_k}(t)) \tag{38}$$

$$x_{Q_k}(t) = \sin(2\pi f_c t + \theta_{Q_k}(t)) \tag{39}$$

where $\theta_{I_k}(t)$ and $\theta_{Q_k}(t)$ are the phase angles used to modulate the carrier signal and are limited to the following values:

$$\theta_{I_k}, \theta_{Q_k} \in \{0, \pi\} \tag{40}$$

Note that there should also be a scaling factor representing the signal amplitude, but this quantity can be ignored for the purposes of this analysis.

The combined signal can be written as:

$$\begin{aligned} X_k(t) &= x_{I_k}(t) + x_{Q_k}(t) && (41) \\ &= \cos(2\pi f_c t + \theta_{I_k}(t)) + \sin(2\pi f_c t + \theta_{Q_k}(t)) && (42) \end{aligned}$$

Now introduce a random time delay τ_k which will cause a phase shift. This quantity is due to the lack of time synchronization between the mobiles on the reverse link and is assumed to be uniformly distributed over the interval $[0, T_c)$.

$$X_k(t) = \cos(2\pi f_c(t - \tau_k) + \theta_{I_k}(t - \tau_k)) + \sin(2\pi f_c(t - \tau_k) + \theta_{Q_k}(t - \tau_k)) \tag{43}$$

θ_{I_k} and θ_{Q_k} have a much lower frequency than f_c, so their dependence on time and the effects of the delay τ_k on these terms can be ignored during this analysis.

$$\begin{aligned} X_k(t) &= \cos(2\pi f_c(t - \tau_k) + \theta_{I_k}) + \sin(2\pi f_c(t - \tau_k) + \theta_{Q_k}) && (44) \\ &= \{\cos(\theta_{I_k})\cos(2\pi f_c(t - \tau_k)) - \sin(\theta_{I_k})\sin(2\pi f_c(t - \tau_k))\} && (45) \\ &\qquad + \{\sin(\theta_{Q_k})\cos(2\pi f_c(t - \tau_k)) + \cos(\theta_{Q_k})\sin(2\pi f_c(t - \tau_k))\} \\ &= \{\cos(\theta_{I_k}) + \sin(\theta_{Q_k})\}\cos(2\pi f_c(t - \tau_k)) && (46) \\ &\qquad + \{-\sin(\theta_{I_k}) + \cos(\theta_{Q_k})\}\sin(2\pi f_c(t - \tau_k)) \end{aligned}$$

Equation (46) can also be written in phasor format to simplify the remainder of the analysis.

$$\begin{aligned} X_k(t) &= \mathrm{Re}\Big\{\Big[\{\cos(\theta_{I_k}) + \sin(\theta_{Q_k})\} - j\{-\sin(\theta_{I_k}) + \cos(\theta_{Q_k})\}\Big] && (47) \\ &\qquad \exp(j2\pi f_c(t - \tau_k))\Big\} \\ &= \mathrm{Re}\Big\{[\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})]\exp(j2\pi f_c(t - \tau_k))\Big\} && (48) \\ &= \mathrm{Re}\Big\{[\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})]\exp(-j2\pi f_c\tau_k)\exp(j2\pi f_c t)\Big\} && (49) \end{aligned}$$

Note that the sin terms in (47) will evaluate as zero due to the constraints given in (40) and can therefore be dropped from the equation. The cos terms will evaluate as ± 1.

To demodulate the in-phase component, multiply equation (49) by $\cos(2\pi f_c t)$ and low-pass filter.

$$\cos(2\pi f_c t) = \frac{1}{2}\Big\{\exp(j2\pi f_c t) + \exp(-j2\pi f_c t)\Big\} \tag{50}$$

This yields the final in-phase demodulated result:

$$\begin{aligned} D_{I_k}(t) &= \mathrm{Re}\Big\{[\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})]\exp(-j2\pi f_c \tau_k)\exp(j2\pi f_c t) \quad (51)\\ &\qquad \left[\frac{1}{2}\{\exp(j2\pi f_c t) + \exp(-j2\pi f_c t)\}\right]\Big\} \\ &\overset{\text{LPF}}{=} \mathrm{Re}\left\{\frac{1}{2}[\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})]\exp(-j2\pi f_c \tau_k)\right\} \quad (52)\\ &= \frac{1}{2}\cos(\theta_{I_k})\cos(2\pi f_c \tau_k) - \frac{1}{2}\cos(\theta_{Q_k})\sin(2\pi f_c \tau_k) \quad (53) \end{aligned}$$

If the time delay τ_k is such that the effective phase due to the time delay is a multiple of 2π, it will be possible to recover the in-phase signal exactly. Otherwise, the quadrature signal will contribute interference.

To demodulate the quadrature component, multiply equation (49) by $\sin(2\pi f_c t)$ and low-pass filter.

$$\sin(2\pi f_c t) = -j\frac{1}{2}\Big\{\exp(j2\pi f_c t) - \exp(-j2\pi f_c t)\Big\} \tag{54}$$

This yields the demodulated quadrature signal:

$$\begin{aligned} C_{Q_k}(t) &= \mathrm{Re}\Big\{[\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})]\exp(-j2\pi f_c \tau_k)\exp(j2\pi f_c t) \quad (55)\\ &\qquad \left[-j\frac{1}{2}\{\exp(j2\pi f_c t) - \exp(-j2\pi f_c t)\}\right]\Big\} \\ &\overset{\text{LPF}}{=} \mathrm{Re}\left\{\frac{1}{2}[j\cos(\theta_{I_k}) + \cos(\theta_{Q_k})]\exp(-j2\pi f_c \tau_k)\right\} \quad (56)\\ &= \frac{1}{2}\cos(\theta_{I_k})\sin(2\pi f_c \tau_k) + \frac{1}{2}\cos(\theta_{Q_k})\cos(2\pi f_c \tau_k) \quad (57) \end{aligned}$$

Similarly, there will be interference from the in-phase component unless the phase shift due to the time delay is an integral multiple of 2π.

We note that equations (53) and (57) only represent the demodulation of a signal from a single mobile. Consequently, it is necessary to sum the demodulation results for each mobile in the system in order to obtain the total demodulated signal. If mobile p is the user of interest, then $\tau_p = 0$ but τ_k will be a uniformly distributed random variable for $k \neq p$.

Multiple Element Demodulation

Consider a received signal from mobile k in the form of (49).

$$X_k(t) = \mathrm{Re}\left\{ [\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})] \exp(-j2\pi f_c \tau_k) \exp(j2\pi f_c t) \right\} \tag{58}$$

For the i^{th} antenna element, there is a known phase offset α_{ki} relative to the center of the array. This phase offset depends on the direction of arrival of the signal and the geometry of the antenna array, and the value of this quantity comes directly from the array response vector which was derived in Section 3.3.5. Thus, with this additional phase offset the received signal at the i^{th} array element can be written as:

$$X_{ki}(t) = \mathrm{Re}\left\{ [\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})] \exp(-j\alpha_{ki}) \exp(-j2\pi f_c \tau_k) \exp(j2\pi f_c t) \right\} \tag{59}$$

Let w_{pi} be the i^{th} entry in the beamforming weight vector for the p^{th} mobile which is the mobile of interest. This represents a phase shift of the corresponding element of the signal vector, so w_{pi} can be written as:

$$w_{pi} = \exp(-j\phi_{pi}) \tag{60}$$

$$w_{pi}^* = \exp(j\phi_{pi}) \tag{61}$$

Thus, by combining equations (37), (59) and (61) the final complex quantity that needs to be demodulated can be obtained as:

$$\begin{aligned}
X_p(t) &= \sum_{i=1}^{N_A} w_{pi}^* \sum_{k=1}^{N_M} X_{ki}(t) && (62)\\
&= \sum_{i=1}^{N_A} \sum_{k=1}^{N_M} \mathrm{Re}\Big\{ [\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})] && (63)\\
&\qquad \exp(j\phi_{pi}) \exp(-j\alpha_{ki}) \exp(-j2\pi f_c \tau_{kp}) \exp(j2\pi f_c t) \Big\} \\
&= \sum_{i=1}^{N_A} \sum_{k=1}^{N_M} \mathrm{Re}\left\{ [\cos(\theta_{I_k}) - j\cos(\theta_{Q_k})] \exp(j(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp})) \exp(j2\pi f_c t) \right\} && (64)
\end{aligned}$$

X_p represents the received signal vector containing signals from all of the mobiles after having undergone beamforming with the weights for the p^{th} mobile. τ_{kp} represents the relative time delay of the k^{th} mobile's signal when τ_{pp} is assumed to be zero.

Referring to equations (53) and (57), it is easy to see that the demodulated in-phase and quadrature components should be:

$$\begin{aligned}
D_{I_p}(t) &= \frac{1}{2}\sum_{i=1}^{N_A}\sum_{k=1}^{N_M}\beta_k w_k(t)\Big\{\cos(\theta_{I_k})\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp}) && (65)\\
&\qquad + \cos(\theta_{Q_k})\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\Big\}\\
&= \frac{1}{2}\sum_{i=1}^{N_A}\sum_{k=1}^{N_M}\beta_k w_k(t)\Big\{c_{I_k}\cos(\alpha_{ip}-\phi_{ki}-2\pi f_c\tau_{kp}) && (66)\\
&\qquad + c_{Q_k}\sin(\alpha_{ip}-\phi_{ki}-2\pi f_c\tau_{kp})\Big\}\\
D_{Q_p}(t) &= \frac{1}{2}\sum_{i=1}^{N_A}\sum_{k=1}^{N_M}\beta_k w_k(t)\Big\{-\cos(\theta_{I_k})\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp}) && (67)\\
&\qquad + \cos(\theta_{Q_k})\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\Big\}\\
&= \frac{1}{2}\sum_{i=1}^{N_A}\sum_{k=1}^{N_M}\beta_k w_k(t)\Big\{-c_{I_k}\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp}) && (68)\\
&\qquad + c_{Q_k}\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\Big\}
\end{aligned}$$

where β_k is the received signal strength as included in equation (35), $w_i(t)$ is the current Walsh chip value for the k^{th} mobile, and c_{I_k} and c_{Q_k} represent the current in-phase and quadrature PN chip values which are ± 1.

3.4.3 Correlation

The basic unit of time correlation is one PN chip (T_c). It is necessary to integrate equations (66) and (68) over this time period, multiply by the current known PN chip value for the p^{th} mobile, and add the results together in order to obtain a PN chip correlation value.

For the in-phase and quadrature signal components, the correlation values over one PN chip can be evaluated as:

$$\begin{aligned}
C_{I_p}[r] &= \frac{1}{2}c_{I_p}[r]\sum_{i=1}^{N_A}\sum_{k=1}^{N_M}\beta_k w_k(t)\Big\{c_{I_k}[r]\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})(T_c-\tau_{kp})\\
&\qquad + c_{I_k}[r-1]\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\tau_{kp} && (69)\\
&\qquad + c_{Q_k}[r]\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})(T_c-\tau_{kp})\\
&\qquad + c_{Q_k}[r-1]\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\tau_{kp}\Big\}\\
C_{Q_p}[r] &= \frac{1}{2}c_{Q_p}[r]\sum_{i=1}^{N_A}\sum_{k=1}^{N_M}\beta_k w_k(t)\Big\{c_{I_k}[r]\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})(T_c-\tau_{kp})\\
&\qquad + c_{I_k}[r-1]\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\tau_{kp} && (70)
\end{aligned}$$

$$+ c_{Q_k}[r]\cos(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp})(T_c - \tau_{kp}) + c_{Q_k}[r-1]\cos(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp})\tau_{kp}\Big\}$$

Adding (69) and (70) yields the correlation value for the r^{th} PN chip of mobile p.

$$C_{C_p}[r] = C_{I_p}[r] + C_{Q_p}[r] \tag{71}$$

In the IS95 standard, the quadrature chip values are actually delayed by half a chip period. This feature is incorporated into the simulation program but is not included here since it does not affect the basic results of the analysis.

The correlation value for each Walsh chip is found simply by summing the N_W (4) correlation values for the PN chips corresponding to the current Walsh chip.

$$C_{W_p}[r] = \sum_{m=0}^{N_W-1} C_{C_p}[rN_W + m] \tag{72}$$

If $W_n[\cdot]$ represents the chip sequence of the n^{th} Walsh function, then the correlation value of the r^{th} Walsh function position with the n^{th} Walsh function is:

$$C_{F_p}[r,n] = \sum_{m=0}^{N_F-1} C_{W_p}[rN_F + m]W_n[m] \tag{73}$$

Background Noise Term

Equations (69) and (70) do not include the background noise component. The integrated thermal noise will have a zero-mean Gaussian distribution. The variance of this distribution is relatively easy to find [19, pp. 209].

$$E\left[n^2[r]\right] = \int_0^{T_c}\int_0^{T_c} E\left[n(t_1)n(t_2)\right]dt_1dt_2 \tag{74}$$

$$= \int_0^{T_c}\int_0^{T_c} \mathrm{R}(t_1,t_2)dt_1dt_2 \tag{75}$$

where the autocorrelation of n in this situation is:

$$\mathrm{R}(t_1,t_2) = \begin{cases} \sigma_n^2 & t_1 = t_2 \\ 0 & t_1 \neq t_2 \end{cases} \tag{76}$$

Substituting yields:

$$E\left[n^2[r]\right] = \int_0^{T_c} \sigma_n^2 dt = T_c\sigma_n^2 \tag{77}$$

This variance must be scaled by 1/2 to account for the effects of the demodulation process. As a result, it is only necessary to add a $N(0, T_c\sigma_n^2/2)$ random variable to both (69) and (70) to complete the simulation process.

3.4.4 Viterbi Decoding

In order to reverse the convolutional encoding process described in Section 3.2.1 and recover the original data bits, it is necessary to use a Viterbi decoder. The actual implementation of this algorithm differs slightly, depending upon whether or not bit interleaving has been enabled.

Viterbi Decoding for Noninterleaved Data

Although interleaving the encoded data bits is part of the IS95 standard, it is simpler to consider Viterbi decoding for noninterleaved data first. Certain modifications must be introduced to the basic Viterbi algorithm in order to obtain a decoder that functions correctly on the interleaved information. For noninterleaved data, however, a standard Viterbi decoder such as that described in [21, pp. 454] is appropriate with a slight alteration.

The metric values which are used to evaluate the surviving paths in the decoder are the Walsh function correlation values as calculated in equation (73). Since each Walsh function represents six encoded bits which in turn correspond to two data bits, each state transition in the Viterbi trellis decoder outputs two data bits instead of the usual one. This implies that there are four possible output paths originating from each node in the trellis rather than the standard two. Note also that the known zero value tail bits at the end of each frame reduce the number of surviving paths which must be compared at the end of the decoding process in order to find the one with the lowest overall metric.

Viterbi Decoding for Interleaved Data

The interleaving technique used in IS95 complicates the decoding process since data is interleaved after the convolutional encoding rather than before. The only available metrics are the Walsh function correlation values from (73). However, when interleaving is used, the bits that make up each Walsh function value are not adjacent to each other in the trellis diagram. For example, after interleaving, the bits originating from the convolutional encoder will be in the following order:

$$b_0 b_{32} b_{64} \ldots b_1 b_{33} b_{65} \ldots b_{543} b_{575} \tag{78}$$

Clearly, the value of the first Walsh function will depend on bits $\{b_0, b_{32}, b_{64}, b_{96}, b_{128}, b_{160}\}$. Similarly, when using a Viterbi decoder to recover the original data bits, the six convolutionally encoded bits corresponding to each two data bits will be distributed among six different

Walsh functions. Thus, it is not possible to update the cost of a surviving path simply by adding the metric value for the current Walsh function to it.

The approach used to overcome this problem involves reevaluating the total metric for each potential surviving path at each stage of the decoding process, based on what is known about the path contents so far. This may initially sound to be computationally expensive, but the calculations can be structured in such a way that the overall computational requirements are not much greater than for the noninterleaved situation.

Consider a specific Walsh function position within a frame. Since each Walsh function represents 6 bits, there are $2^6 = 64$ possible values for this position, each of which has a correlation value associated with it from equation (73). Now assume that only the first k bits of this Walsh function are known for a partially completed surviving path. Then there are 2^{6-k} different Walsh function values that could match the known information since the final $6-k$ bits are arbitrary. A metric can be defined for this partially known value by taking the maximum correlation value from among the 2^{6-k} matching Walsh function values. For example, if the first two bits are 10, then Walsh functions 32 through 47 inclusive are potential matches.

As the decoding process progresses, more and more encoded bits are known for the surviving paths. As this information becomes available, the metric for each overall surviving path can be reevaluated. Note that it is possible for a path metric value to decrease in this instance, whereas this will not be the case in the noninterleaved situation. This implies that it may be possible to accidentally delete the overall optimal path at an earlier stage where it may appear to be suboptimal. Therefore, the capability of retaining more than one surviving path at each state node has been incorporated into the simulator design to guard against this possibility.

The metric value for a surviving path $\mathcal{P}$ can be calculated as:

$$\mathcal{M}_P(\mathcal{P}) \;=\; \sum_{i=1}^{F_F} \mathcal{M}_W(\mathcal{P}, i, b_i) \tag{79}$$

where $\mathcal{M}_W$ is the metric value for each Walsh function within the frame and is defined as:

$$\mathcal{M}_W(\mathcal{P}, i, b_i) \;\equiv\; \begin{cases} \max\left[\mathcal{C}_F(\mathcal{P}, i, b_i)\right] & b_i > 0 \\ 0 & b_i = 0 \end{cases} \tag{80}$$

where b_i is the number of bits that are known for the i^{th} Walsh function position in path

$\mathcal{P}$. $\mathcal{C}_F(\mathcal{P}, i, b_i)$ is any correlation value whose corresponding Walsh function value written in binary form matches the first b_i bits of the Walsh function in position i of path $\mathcal{P}$.

While performing Viterbi decoding, only B_W (6) Walsh functions in each path are modified at each stage. Thus, rather than reevaluating the entire path metric, it is only necessary to modify the B_W $\mathcal{M}_W(\cdot)$ values which have changed.

3.5 Initializing a Mobile Call

There are a number of tasks which must be executed whenever a mobile begins a new call.

The first step is to assign random coordinates to the mobile and ensure that this location falls within a valid cell. The nearest base station and its immediately adjacent neighbors (in the event of a multi-cell simulation) are identified and considered as candidates for the base station with which the mobile will communicate.

The length of the call in frames is taken from an exponential probability distribution with a specified mean.

Next, the initial path loss and shadowing values are randomly determined for each of the candidate base stations. The path loss depends upon the distances between the mobile and the base stations and is calculated as shown in equation (17). The shadowing value is computed from equation (18). Based on the path loss and shadowing values, the base station with the least attenuation in the transmission channel is selected as the one with which the mobile will communicate. This may not necessarily be the nearest one to the mobile due to the variability introduced by the shadowing factor.

The initial transmission power for the mobile is computed. This is accomplished by estimating the total interference and noise power as seen by the base station when synchronized with that particular mobile's signal. This information is then used together with the channel parameters to determine the initial transmission power required to obtain the desired E_b/N_0 ratio. If this power level is above the maximum possible output from the mobile, the call is denied and another attempt must be made to reinitialize it. A count of the number of denied calls is maintained for system performance evaluation purposes.

If P_{M_k} is the power received from the k^{th} mobile and P_{I_k} is the interference power, then the E_b/N_0 ratio is defined as:

$$\frac{E_b}{N_0} = \frac{P_{M_k}/R_B}{P_{I_k}/B + \sigma_n^2} = \frac{P_{M_k}B}{R_B(P_{I_k} + B\sigma_n^2)} \tag{81}$$

where B is the spread spectrum bandwidth. Solve (81) for P_{M_k}.

$$P_{M_k} = \frac{R_B(P_{I_k} + B\sigma_n^2)}{B} \times \frac{E_b}{N_0} \tag{82}$$

The average value of the power received from the mobile can be written as:

$$P_{M_k} = E\left[R^2\right] \rho_k S_k P_{T_k} \tag{83}$$

where P_{T_k} is the power of the signal at the transmitter (mobile) and is the quantity to be determined. Note that we need to consider the expected value of the square of the Rayleigh fading since it is the only quantity which is a random variable during the actual duration of a call.

Equate (82) and (83).

$$\frac{R_B(P_{I_k} + B\sigma_n^2)}{B} \times \frac{E_b}{N_0} = E\left[R^2\right] \rho_k S_k P_{T_k} \tag{84}$$

Solve (84) for P_{T_k}.

$$P_{T_k} = \frac{1}{E\left[R^2\right] \rho_k S_k} \left(\frac{R_B(P_{I_k} + B\sigma_n^2)}{B} \times \frac{E_b}{N_0} \right) \tag{85}$$

In the simulation, all of the terms on the right side of (85) are known, so the initial signal power can be calculated. The amplitude of the signal envelope at the mobile is the square root of P_{T_k}.

Relative time delays between the signal from the new mobile and signals from existing mobiles are then calculated. This is accomplished by assigning a random absolute time delay for each mobile's signal at each base station whenever a new call is initialized. This time delay is a uniformly distributed random variable in the interval $[0, T_c)$. By comparing the absolute time delay values for the mobiles, it is simple to compute the relative time delays with respect to each mobile.

4 Power Control

Power control is an important component of a CDMA communication system. If the ratio of the received power from a given mobile to the corresponding interference power is too low, an unacceptably large number of errors in the data flow will occur. Conversely, if this ratio is too high, the mobile will generate excessive interference for other users. Therefore, it is necessary to measure the received power and interference power for each user, and then to control the transmission power level for each mobile so that a state of relative equilibrium can be maintained.

In the simulator, two types of power control can be used: perfect and imperfect. These power control methods are discussed in more detail below. However, it is first necessary to explain how the power levels are actually measured.

4.1 Power Estimation

In order to perform power control, it is necessary to estimate or measure the signal power originating from each mobile. In addition, the interference power from all of the other mobiles in the system must be estimated. An antenna array with N_A elements is assumed here since the derivation results can be reduced to the single element case simply by setting $N_A = 1$.

From equations (66) and (68), the received power values for the in-phase and quadrature components of the k^{th} mobile when the beamforming coefficients for the p^{th} mobile are used can be derived to be:

$$P_{I_{pk}} = \frac{1}{4}\beta_k^2 \left\{ \left[\sum_{i=1}^{N_A} \cos(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) \right]^2 + \left[\sum_{i=1}^{N_A} \sin(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) \right]^2 \right\} \quad (86)$$

$$P_{Q_{pk}} = \frac{1}{4}\beta_k^2 \left\{ \left[\sum_{i=1}^{N_A} \sin(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) \right]^2 + \left[\sum_{i=1}^{N_A} \cos(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) \right]^2 \right\} \quad (87)$$

where β_k^2 is the received signal strength. For notational simplicity, we define:

$$\Upsilon_{pk} = \frac{1}{2} \left\{ \left[\sum_{i=1}^{N_A} \sin(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) \right]^2 + \left[\sum_{i=1}^{N_A} \cos(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) \right]^2 \right\} \quad (88)$$

Note that $P_{I_{pk}}$ and $P_{Q_{pk}}$ are equal. Since the in-phase and quadrature signal components are independent due to the differing short code PN sequences, the received power for mobile p will be:

$$P_{M_p} = P_{I_{pp}} + P_{Q_{pp}} = \Upsilon_{pp}\beta_p^2 \quad (89)$$

The corresponding interference power for the p^{th} mobile will be:

$$P_{I_p} = \sum_{k \neq p} \left(P_{I_{pk}} + P_{Q_{pk}} \right) = \sum_{k \neq p} \Upsilon_{pk}\beta_k^2 \quad (90)$$

β_k^2 is the only quantity which varies with time during a frame since the Rayleigh fading value is changed regularly. The beamforming weights also change dynamically, but are only updated on a frame by frame basis. Consequently, the remaining terms in the above expressions need only be precalculated once per frame to reduce the computational expense required.

4.2 Perfect Power Control

In perfect power control, each mobile's transmission power is set so that its corresponding E_b/N_0 is exactly the desired value. In actuality, since dynamic Rayleigh fading is used, the transmission power levels are chosen such that the observed E_b/N_0 averaged over time will equal the desired value.

Since the E_b/N_0 ratio depends on both the power received from the desired mobile and the interference power from other mobiles, it is necessary to solve a simultaneous system of N_M linear equations with N_M unknowns in order to determine the appropriate transmission levels for each mobile. For the p^{th} mobile, E_b/N_0 can be calculated as:

$$\left(\frac{E_b}{N_0}\right)_p = \frac{P_{M_p}/R_B}{P_{I_p}/B + \sigma_n^2} \tag{91}$$

where the mobile and interference powers are defined in (89) and (90) respectively.

Each $(E_b/N_0)_p$ in (91) can be set to the exact desired value. By rearranging this equation, the following expression can be obtained.

$$\left(\frac{E_b}{N_0}\right)_p \left(\frac{P_{I_p}}{B} + \sigma_n^2\right) = \frac{P_{M_p}}{R_B} \tag{92}$$

Substituting (89) and (90) into (92) yields:

$$\frac{1}{R_B\,(E_b/N_0)_p}\Upsilon_{pp}\beta_p^2 - \frac{1}{B}\sum_{k\neq p}\Upsilon_{pk}\beta_k^2 = \sigma_n^2 \tag{93}$$

If P_{T_p} is the transmission power of the p^{th} mobile, then the expected received signal strength from this mobile will be:

$$E\left[\beta_p^2\right] = \rho_p S_p E\left[R^2\right] P_{T_p} \tag{94}$$

where the effects of the channel parameters (path loss, shadowing, and Rayleigh fading) on the signal power have been taken into account. P_{T_p} represents the unknown quantity which we want to determine. Substitute (94) into (93) to obtain:

$$\frac{1}{R_B\,(E_b/N_0)_p}\Upsilon_{pp}\rho_p S_p E\left[R^2\right]P_{T_p} - \frac{1}{B}\sum_{k\neq p}\Upsilon_{pk}\rho_k S_k E\left[R^2\right]P_{T_k} \;=\; \sigma_n^2 \qquad (95)$$

There are N_M unknowns $(P_{T_p},\ P_{T_k})$ in each equation and p can take on N_M different values. The result is a system of N_M equations with N_M unknowns which can be easily solved using Gaussian elimination.

Of course, perfect power control does not reflect reality. However, this facility does provide a useful testing scenario for verifying error analyses that assume perfect power control.

4.3 Imperfect Power Control

A more realistic power control approach which is imperfect in nature involves estimating the power received from each mobile over a set period of time, comparing this to the noise and interference power, and instructing each mobile to either raise or lower its transmitted signal strength by some set amount based on whether the observed E_b/N_0 ratio is below or above a specified threshold respectively. If the number of mobiles does not exceed the capacity of the cell, a situation of relative equilibrium in the power levels can be attained.

This algorithm is imperfect in two senses. Firstly, the average E_b/N_0 ratios will not be exactly the desired value due to the quantization of the power adjustment step size. Secondly, there is a time delay involved in the control process. The received power levels must initially be estimated over a period of time. Some additional time is then required to process this data and transmit the appropriate control directions back to each mobile. Thus, there is a definite time lag before a mobile will actually react to its power levels being too low or too high. In addition, there is a nonzero probability that a power control bit (which specifies whether the transmission power should be raised or lowered) will have its value flipped due to transmission noise.

In the simulation, the received power levels are measured over the length of one power control period. Since there are $N_{PC} = 16$ of these periods in each frame for IS95, each period has a length of $L_{PC} = 1536$ PN chips. A new power control bit is generated and transmitted at the end of each of these periods.

4.4 Voice/Data Transmission Activity

In voice applications, users typically spend less than 50% of their time actually talking. The actual percentage of time when speech is active is defined as the voice activity factor (ψ) and is generally assumed to have a value between 0.35 and 0.45. For data transmission applications, we set $\psi = 1.0$ to simulate full loading

During periods of voice inactivity, there is less data to transmit since there is no voice signal to encode and send. Thus, the transmission power level can be reduced by a factor of ν ($\nu = 8$ for IS95) when speech is inactive, while still maintaining the communication link. This feature has the effect of reducing the average interference power to other mobiles. As a result, the net capacity of a cell will increase.

In our simulation program, periods of speech activity and inactivity are assumed to have exponentially distributed lengths with mean values specified for each of these two quantities. Usually, the same mean is used for both lengths so that a voice activity factor of 0.5 results.

5 Beamforming

Digital beamforming can be utilized to increase the capacity of a cell [10] [12] [17] [18]. By using a multi-element antenna array at the base station and a suitable set of beamforming coefficients, it is possible to take a weighted sum of the antenna element outputs which maximizes the ratio of the desired mobile's signal power to the noise and interference power. This is essentially a form of spatial filtering that reduces interference from other mobiles and thus boosts overall cell capacity. Applebaum [1], Swales *et al* [25], and Van Veen and Buckley [26] provide good introductions to beamforming and its effects on increasing the number of users in a communication system.

For the type of beamforming being considered in the current simulator, it is desirable to estimate the array response vector of the mobile being considered. Two types of beamforming are currently available for simulation purposes. Perfect beamforming assumes exact knowledge of the required array response vector, whereas imperfect beamforming attempts to estimate a suitable set of beamforming weights from the received signal data.

5.1 Perfect Beamforming

As mentioned above, perfect beamforming assumes that the array response vector for the mobile under consideration is known exactly. The beamforming weights for the p^{th} mobile can then be set equal to the entries of the array response vector since this set of weights will be optimal when the interference is assumed to be white, which will be the case for a large number of active mobiles.

$$\mathbf{w}_p = \mathbf{a}_p \tag{96}$$

Although in real life the true array response vector would not be known, this case is useful for debugging purposes and for verifying error analyses that were performed with simplifying assumptions.

5.2 Imperfect Beamforming

Imperfect beamforming represents a more realistic scenario. In this approach, the observed signal data is processed to produce an estimate for the beamforming weights.

The array response vector $\mathbf{a}_p$ specifies the relative phases of the p^{th} mobile's signal as received at each of the N_A antenna elements. This vector is a function of the signal's direction of arrival and the geometry of the antenna array. It is necessary to estimate $\mathbf{a}_p$ in order to determine suitable beamforming coefficients which are a linear function of this vector.

Let $\mathbf{x}_p[r]$ be the received signal vector correlated with the r^{th} PN chip of the p^{th} mobile's chip sequence, and let $\mathbf{z}_p[n]$ be $\mathbf{x}_p[r]$ correlated over the time period of the n^{th} Walsh chip. Then, $\mathbf{a}_p$ can be estimated as the principal eigenvector of the matrix $\mathrm{R}_{\mathbf{zz}} - \mathrm{R}_{\mathbf{xx}}$ where $\mathrm{R}_{\mathbf{zz}}$ and $\mathrm{R}_{\mathbf{xx}}$ are the correlation matrices of $\mathbf{x}_p$ and $\mathbf{z}_p$ respectively (note that $\mathbf{x}_p$ and $\mathbf{z}_p$ are both zero mean) [12] [24]. Since these matrices must be estimated from the observed data, the accuracy of these estimates affects the system performance.

We note that the asymptotic values of $\mathrm{R}_{\mathbf{xx}}$ and $\mathrm{R}_{\mathbf{zz}}$ can be written as:

$$\mathrm{R}_{\mathbf{xx}} = P_p \mathbf{a}_p \mathbf{a}_p^* + \sum_{k \neq p} P_k \mathbf{a}_k \mathbf{a}_k^* + \sigma_n^2 \mathrm{I} \tag{97}$$

$$\mathrm{R}_{\mathbf{zz}} = G P_p \mathbf{a}_p \mathbf{a}_p^* + \sum_{k \neq p} P_k \mathbf{a}_k \mathbf{a}_k^* + \sigma_n^2 \mathrm{I} \tag{98}$$

where G is the gain or spreading factor, P_p represents the received signal power, and σ_n^2 is the thermal noise power.

The covariance of the noise and interference for the p^{th} user can be estimated as [15]:

$$\mathrm{Q}_p = \frac{1}{G-1}(G\mathrm{R}_{\mathbf{xx}} - \mathrm{R}_{\mathbf{zz}}) \tag{99}$$

The optimum set of beamforming weights is given by the Wiener solution [15] [20].

$$\mathcal{W}_p = \frac{\mathrm{Q}_p^{-1}\mathbf{a}_p}{\mathbf{a}_p^*\mathrm{Q}_p^{-1}\mathbf{a}_p} \tag{100}$$

It should be noted that equation (100) will yield a vector of beamforming weights which have an essentially random unknown absolute phase shift relative to the desired values. That is, all of the entries of $\mathcal{W}_p$ will need to be phase-shifted by the same amount Φ_p in order to obtain suitable beamforming weights.

Consider equation (37) again with the uncorrected Wiener beamforming weights.

$$x_p = \mathcal{W}_p^*\mathbf{x} \tag{101}$$

Here, we are essentially adding a set of phase-shifted and potentially amplitude-scaled sinusoids, all of which share the same frequency. The result will be another sinusoidal signal with the same frequency and a phase shift of Θ_p. This quantity can be calculated, so it is simply necessary to apply the following phase shift to the original beamforming weights.

$$\Phi_p = -\Theta_p \tag{102}$$

This will ensure that the signal will have no phase offset after beamforming has taken place.

5.2.1 Complex Eigenvalues and Eigenvectors

As mentioned in Section 5.2, it is necessary to calculate the eigenvalues and eigenvectors of a complex Hermitian matrix. The simplest way to accomplish this is to reformulate the problem as a real-valued matrix [27, pp. 174].

Consider the following complex Hermitian matrix.

$$\mathrm{A} = \mathrm{B} + j\mathrm{C} \tag{103}$$

where B and C are real and satisfy the following properties.

$$\mathrm{B} = \mathrm{B}^T \tag{104}$$

$$\mathrm{C} = -\mathrm{C}^T \tag{105}$$

Let λ_i be an eigenvalue of A (necessarily real) and let $\mathbf{u}_i + j\mathbf{v}_i$ be the corresponding complex eigenvector. Thus:

$$\mathrm{A}(\mathbf{u}_i + j\mathbf{v}_i) = \lambda_i(\mathbf{u}_i + j\mathbf{v}_i) \tag{106}$$

Substituting (103) into (106) produces:

$$(\mathrm{B} + j\mathrm{C})(\mathbf{u}_i + j\mathbf{v}_i) = \lambda_i(\mathbf{u}_i + j\mathbf{v}_i) \tag{107}$$

Expanding equation (107) and separating the real and imaginary terms implies that:

$$\mathrm{B}\mathbf{u}_i - \mathrm{C}\mathbf{v}_i = \lambda_i\mathbf{u}_i \tag{108}$$

$$\mathrm{C}\mathbf{u}_i + \mathrm{B}\mathbf{v}_i = \lambda_i\mathbf{v}_i \tag{109}$$

Writing (108) and (109) in matrix form produces:

$$\begin{bmatrix} \mathrm{B} & -\mathrm{C} \\ \mathrm{C} & \mathrm{B} \end{bmatrix} \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} = \lambda_i \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} \tag{110}$$

Noting that both sides of (106) can be multiplied by $-j$ similarly results in the following matrix equation.

$$\begin{bmatrix} \mathrm{B} & -\mathrm{C} \\ \mathrm{C} & \mathrm{B} \end{bmatrix} \begin{bmatrix} \mathbf{v}_i \\ -\mathbf{u}_i \end{bmatrix} = \lambda_i \begin{bmatrix} \mathbf{v}_i \\ -\mathbf{u}_i \end{bmatrix} \tag{111}$$

The new real matrix on the left side of (110) and (111) has λ_i as a double eigenvalue. Therefore, it is simply necessary to calculate the eigenvalues and eigenvectors of this matrix D where:

$$\mathrm{D} = \begin{bmatrix} \mathrm{B} & -\mathrm{C} \\ \mathrm{C} & \mathrm{B} \end{bmatrix} \tag{112}$$

The Jacobi method is used to simultaneously solve for all of the eigenvalues and eigenvectors of the matrix [27, pp. 266-267]. Ideally, only the principal eigenvector is required so the power method [27, pp. 571-572] could be used instead, but is is useful to have all of the terms available for error analysis purposes.

6 Cell Capacity Estimation

As discussed back in the introduction, our research goal is to investigate the effects of digital beamforming and power control on cell capacity. This includes both perfect and imperfect beamforming and power control.

Cell capacity is defined as the maximum number of users per cell that can be supported with a given level of performance. With respect to voice communications, this is commonly taken to mean an observed BER (Bit Error Rate) of 10^{-3} or less. Consequently, a simple way of measuring cell capacity is to gradually add more users to a system until the observed BER exceeds the desired value.

The simulation program is currently capable of estimating cell capacity using two different approachs: a chip-level simulation and a power control method. This are discussed in more detail in the following sections.

6.1 Chip-Level Capacity Estimation

Chip-level capacity simulations are currently used to investigate situations involving imperfect power control and/or imperfect beamforming. These simulations are computationally expensive, so must currently be limited to a small number of antenna elements (*e.g.* $N_A \leq 3$). However, the ability to obtain data at the chip level is extremely important since it serves to validate simplifying assumptions which can allow faster simulations to be performed.

Since data is actually being transmitted through the system, it is quite easy to produce an estimate of the bit error rate for a given number of mobiles and antenna array elements simply by comparing the transmitted bits with the data recovered at the receiver. Cell capacity can be estimated by gradually increasing the number of mobiles until the observed BER exceeds the desired limit.

One problem with this approach is ensuring that a large enough number of frames are processed in each simulation to obtain an accurate estimate of the true BER for the situation under consideration.

6.2 Power Control Capacity Estimation

As has been mentioned, performing a chip-level simulation of a CDMA communication system is computationally expensive, especially for a large number of mobiles. As a result, the simulations which can be run are limited both in size and length.

It is generally accepted that a specific E_b/N_0 ratio corresponds to a given BER. Consequently, if perfect power control is used so that this E_b/N_0 value is maintained for all of the mobiles, then it is reasonably safe to assume that the observed BER will not be greater than the desired limit. This eliminates the need to actually encode and decode data bits via the chip-level technique. Perfect power control is only performed at the start of each frame, so it is much less computationally expensive than would be a chip-level simulation. Consequently, it is possible to generate capacity estimates over a longer time period and for a greater number of antenna elements (since this will require simulating more mobiles in order to saturate the cell).

The above assumption of perfect power control is, of course, unrealistic. However, it does allow comparisons between the capacity estimates for different numbers of antenna elements to be made efficiently.

To obtain a capacity estimate for a given number of antenna elements, power control simulations are performed over a large number of frames for successively increasing numbers of mobiles. For each simulation, the percentage of frames where the power control failed (*i.e.* it was physically impossible to achieve the desired E_b/N_0 ratio for all of the mobiles) is recorded. Once this percentage exceeds a certain amount (*e.g.* 1%), the capacity of the cell is deemed to have been reached.

The cell capacity can actually be analytically calculated for the case of one antenna element since the received mobile power is the same for each user. However, for a multiple element antenna array, each mobile will have a slightly different received power level since its interference power will vary based on the spatial distribution of the other users. Consequently, it is necessary to perform simulations over a large number of frames while continually moving the mobiles around at random in order to obtain accurate estimates of the cell capacity in this situation.

7 Error Analysis & System Validation

An error analysis of the simulated system has been performed. This allows the statistical properties of certain quantities to be predicted and compared with the corresponding values from simulations. A reasonable agreement between the predictions and observations tends to support the acceptability of the simulation results and thus the validity of the simulator.

7.1 Statistical Parameters of Correlation Quantities

Equation (69) contains the correlated in-phase signal component for the p^{th} mobile. Neglecting the thermal noise term for the moment, we have:

$$\begin{aligned} C_{I_p}[r] &= \frac{1}{2}c_{I_p}[r]\sum_{k=1}^{N_M}\beta_k w_k(t)\sum_{i=1}^{N_A}\Big\{c_{I_k}[r]\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})(T_c-\tau_{kp}) \\ &\quad + c_{I_k}[r-1]\cos(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\tau_{kp} \\ &\quad + c_{Q_k}[r]\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})(T_c-\tau_{kp}) \\ &\quad + c_{Q_k}[r-1]\sin(\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp})\tau_{kp}\Big\} \end{aligned} \tag{113}$$

Assume that the τ_{kp} time delay values are uniformly distributed between 0 and T_c and that $\tau_{pp}=0$. Therefore, the quadrature components of mobile p will disappear since the sin terms will evaluate as zero. Multiplying the other $w_k c_{I_k}$ and $w_k c_{Q_k}$ terms by $c_{I_p}[r]$ will result in binary random variables with the values ± 1 (X_{I_k} and X_{Q_k}). Also, $Y_{pki}=\phi_{pi}-\alpha_{ki}-2\pi f_c\tau_{kp}$ can be assumed to be a uniformly distributed random variable in $[0,2\pi)$. If the in-phase PN chip correlation value is treated as a random variable, it can be written as:

$$\begin{aligned} C_{I_1}[r] &= \frac{1}{2}N_A\beta_p T_c w_p(t)+\frac{1}{2}\sum_{k\neq p}\beta_k\sum_{i=1}^{N_A} \\ &\quad \Big\{X_{I_k}[r]\cos(Y_{pki})(T_c-\tau_{kp})+X_{I_k}[r-1]\cos(Y_{pki})\tau_{kp} \\ &\quad + X_{Q_k}[r]\sin(Y_{pki})(T_c-\tau_{kp})+X_{Q_k}[r-1]\sin(Y_{pki})\tau_{kp}\Big\} \end{aligned} \tag{114}$$

Let I_p be a random variable equal to the summation term in (114) which represents interference from other mobiles and which can be modeled as being Gaussian via the Central Limit Theorem. We now wish to determine the mean and variance of this distribution.

We assume X_{I_k}, X_{Q_k}, Y_{pki}, and τ_{pk} to all be independent. Although Y_{pki} is a function of τ_{pk}, the fact that its value also depends on ϕ_{pi} and α_{ki} lessens the dependence on τ_{pk}. Thus, when taking the expected value of I_1, the individual terms in the summation can be factored into the products of the expectations. X_{I_k} and X_{Q_k} are both zero-mean, so the mean of I_p will also be zero.

$$E\left[I_p\right] = 0 \tag{115}$$

The expected value of I_p^2 (which will equal the variance since I_p is zero-mean) is:

$$
\begin{aligned}
E\left[I_p^2\right] &= \frac{1}{4}E\Bigg[\Bigg(\sum_{k\neq p}\beta_k\sum_{i=1}^{N_A}\Big\{X_{I_k}[r]\cos(Y_{pki})(T_c-\tau_{kp})+X_{I_k}[r-1]\cos(Y_{pki})\tau_{kp} \\
&\quad + X_{Q_k}[r]\sin(Y_{pki})(T_c-\tau_{kp})+X_{Q_k}[r-1]\sin(Y_{pki})\tau_{kp}\Big\}\Bigg) \qquad (116)\\
&\quad \Bigg(\sum_{m\neq p}\beta_m\sum_{n=1}^{N_A}\Big\{X_{I_m}[r]\cos(Y_{pmn})(T_c-\tau_{mp})+X_{I_m}[r-1]\cos(Y_{pmn})\tau_{mp} \\
&\quad + X_{Q_m}[r]\sin(Y_{pmn})(T_c-\tau_{mp})+X_{Q_m}[r-1]\sin(Y_{pmn})\tau_{mp}\Big\}\Bigg)\Bigg] \\
&= \frac{1}{4}E\Bigg[\sum_{k\neq p}\beta_k\sum_{m\neq p}\beta_m\sum_{i=1}^{N_A}\sum_{n=1}^{N_A}\Big\{X_{I_k}[r]\cos(Y_{pki})(T_c-\tau_{kp})+X_{I_k}[r-1]\cos(Y_{pki})\tau_{kp} \\
&\quad + X_{Q_k}[r]\sin(Y_{pki})(T_c-\tau_{kp})+X_{Q_k}[r-1]\sin(Y_{pki})\tau_{kp}\Big\} \qquad (117)\\
&\quad \Big\{X_{I_m}[r]\cos(Y_{pmn})(T_c-\tau_{mp})+X_{I_m}[r-1]\cos(Y_{pmn})\tau_{mp} \\
&\quad + X_{Q_m}[r]\sin(Y_{pmn})(T_c-\tau_{mp})+X_{Q_m}[r-1]\sin(Y_{pmn})\tau_{mp}\Big\}\Bigg]
\end{aligned}
$$

The cos and sin terms are all zero-mean. Since Y_{pki} and Y_{pmn} are assumed to be independent unless $k=m$ and $i=n$, all of the expectations in (117) with $k\neq m$ and/or $i\neq n$ can be factored and will therefore evaluate as zero. Thus, (117) can be immediately simplified to:

$$
\begin{aligned}
E\left[I_p^2\right] &= \frac{1}{4}E\Bigg[\sum_{k\neq p}\beta_k^2\sum_{i=1}^{N_A}\Big\{X_{I_k}[r]\cos(Y_{pki})(T_c-\tau_{kp})+X_{I_k}[r-1]\cos(Y_{pki})\tau_{kp} \qquad (118)\\
&\quad + X_{Q_k}[r]\sin(Y_{pki})(T_c-\tau_{kp})+X_{Q_k}[r-1]\sin(Y_{pki})\tau_{kp}\Big\}^2\Bigg]
\end{aligned}
$$

Similarly, $X_{I_k}[r]$, $X_{I_k}[r-1]$, $X_{Q_k}[r]$, and $X_{Q_k}[r-1]$ are all independent zero-mean random variables, so product terms in (118) which do not include a squared random variable can be factored and will have an expected value of zero. Therefore, (118) can be further reduced to:

$$
\begin{aligned}
E\left[I_p^2\right] &= \frac{1}{4}E\Bigg[\sum_{k\neq p}\beta_k^2\sum_{i=1}^{N_A}\Big(\{X_{I_k}[r]\cos(Y_{pki})(T_c-\tau_{kp})\}^2+\{X_{I_k}[r-1]\cos(Y_{pki})\tau_{kp}\}^2 \qquad (119)\\
&\quad + \{X_{Q_k}[r]\sin(Y_{pki})(T_c-\tau_{kp})\}^2+\{X_{Q_k}[r-1]\sin(Y_{pki})\tau_{kp}\}^2\Big)\Bigg]
\end{aligned}
$$

The expected values of $X_{I_k}^2[r]$, $X_{I_k}^2[r-1]$, $X_{Q_k}^2[r]$, and $X_{Q_k}^2[r-1]$ will all be $+1$. Thus:

$$
E\left[I_p^2\right] = \frac{1}{4}E\Bigg[\sum_{k\neq p}\beta_k^2\sum_{i=1}^{N_A}\Big(\{\cos(Y_{pki})(T_c-\tau_{kp})\}^2+\{\cos(Y_{pki})\tau_{kp}\}^2 \qquad (120)
$$

$$+\left\{\sin(Y_{pki})(T_c-\tau_{kp})\right\}^2+\left\{\sin(Y_{pki})\tau_{kp}\right\}^2\Big)\Big]$$

$$= \frac{1}{4}N_A E\left[\sum_{k\neq p}\beta_k^2\left\{(T_c-\tau_{kp})^2+\tau_{kp}^2\right\}\right] \tag{121}$$

β_k and τ_{kp} are the only two remaining random variables. Since they are independent, the expected value can be factored and evaluated as follows.

$$E\left[I_p^2\right] = \frac{1}{4}N_A\sum_{k\neq p}E\left[\beta_k^2\right]\left\{T_c^2-2T_cE\left[\tau_{kp}\right]+2E\left[\tau_{kp}^2\right]\right\} \tag{122}$$

τ_{kp} has a uniform distribution between 0 and T_c. Therefore, its first two moments are:

$$E\left[\tau_{kp}\right] = \frac{1}{2}T_c \tag{123}$$

$$E\left[\tau_{kp}^2\right] = \frac{1}{3}T_c^2 \tag{124}$$

Substituting (123) and (124) into (122) yields:

$$E\left[I_p^2\right] = \frac{1}{6}N_AT_c^2\sum_{k\neq p}E\left[\beta_k^2\right] \tag{125}$$

β_k^2 represents the power received from the k^{th} mobile (a scaling factor has been dropped for notational simplicity). Let P represent the power received from a mobile with active speech. Assuming perfect power control and the voice activity situation described in Section 4.4, the expected value of β_k^2 will be:

$$E\left[\beta_k^2\right] = E\left[\psi P+(1-\psi)\frac{P}{\nu}\right] = \left(\frac{(\nu-1)\psi+1}{\nu}\right)E\left[P\right] \tag{126}$$

Substituting (126) into (125) and eliminating the summation produces:

$$E\left[I_p^2\right] = \frac{N_A(N_M-1)T_c^2\{(\nu-1)\psi+1\}E\left[P\right]}{6\nu} \tag{127}$$

The calculation of $E\left[P\right]$ is shown in Sections 7.2 and 7.3.

Therefore, $C_{I_p}[r]$ will have a Gaussian distribution with the following parameters.

$$\mu_c = \frac{1}{2}N_Aw_k(t)T_cE[R]\sqrt{\frac{E[P]}{E[R^2]}} = \frac{\sqrt{\pi}}{4}N_Aw_k(t)T_c\sqrt{E[P]} \tag{128}$$

$$\sigma_c^2 = \frac{N_A(N_M-1)T_c^2\{(\nu-1)\psi+1\}E[P]}{6\nu}+\frac{1}{2}N_AT_c\sigma_n^2 \tag{129}$$

$E[R]$ and $E[R^2]$ are included in (128) to account for Rayleigh fading, and the second term in (129) is due to thermal noise. Similarly, $C_{Q_p}[r]$ should have the same Gaussian distribution.

The sum of the N_W in-phase and quadrature PN chip correlations over one Walsh chip will be a Gaussian random variable with the following parameters.

$$\mu_w = 2N_W\mu_c \tag{130}$$

$$\sigma_w^2 = 2N_W\sigma_c^2 \tag{131}$$

Correlating with the Walsh functions over N_F Walsh chips will also produce a Gaussian random variable. If the signal is correlated with the correct Walsh function chip sequence (*i.e.* the one which was actually transmitted), the mean should be:

$$\mu_F = N_F\mu_w = 2N_FN_W\mu_c \tag{132}$$

If the signal is correlated with any of the other Walsh function chip sequences which do not match the transmitted sequence and are therefore incorrect, the result will be zero-mean since the Walsh function chip sequences are all mutually orthogonal.

$$\mu_{\overline{F}} = 0 \tag{133}$$

In both of these cases, the variance of the Walsh function correlation values will be the same.

$$\sigma_F^2 = N_F\sigma_w^2 = 2N_FN_W\sigma_c^2 \tag{134}$$

7.2 Single Element Power Value Calculation

If $E[P]$ is the expected power level received from a mobile with active speech and perfect power control is being used, then the mean interference power seen by that mobile will be:

$$E\left[I\right] = E\left[(N_M-1)\left\{\psi P + (1-\psi)\frac{P}{\nu}\right\}\right] = (N_M-1)\left(\psi + \frac{1-\psi}{\nu}\right)E\left[P\right] \tag{135}$$

Using these expected values, the E_b/N_0 ratio is defined as:

$$\frac{E_b}{N_0} = \frac{E[P]/R_B}{E\left[I\right]/B + \sigma_n^2} \tag{136}$$

Substituting (135) into (136) and solving for $E[P]$ yields:

$$E[P] = \left(\frac{E_b}{N_0}\right)\sigma_n^2\left\{\frac{1}{R_B} - \frac{N_M-1}{B}\left(\frac{E_b}{N_0}\right)\left(\psi + \frac{1-\psi}{\nu}\right)\right\}^{-1} \tag{137}$$

Equation (137) can now be used in (128) and (129) to determine μ_c and σ_c^2.

7.3 Multi-Element Power Value Calculation

Consider the sums of sin and cos from (88).

$$\mathcal{S}_{pk} = \sum_{i=1}^{N_A} \sin(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) = \sum_{i=1}^{N_A} \sin(Y_{pki}) \tag{138}$$

$$\mathcal{C}_{pk} = \sum_{i=1}^{N_A} \cos(\phi_{pi} - \alpha_{ki} - 2\pi f_c \tau_{kp}) = \sum_{i=1}^{N_A} \cos(Y_{pki}) \tag{139}$$

where the phase angles of the sin and cos expressions are assumed to be uniformly distributed random variables in the interval $[0, 2\pi)$. By the Central Limit Theorem, both $\mathcal{S}_{pk}$ and $\mathcal{C}_{pk}$ can be approximated by a Gaussian distribution. From simulations, we have been able to determine that a good approximation is obtained even when N_A is as small as 4 or 5.

Since sin and cos are zero-mean, $\mathcal{S}_{pk}$ and $\mathcal{C}_{pk}$ will also both be zero-mean.

$$\mu_{\mathcal{S}} = \mu_{\mathcal{C}} = 0 \tag{140}$$

The variances of $\sin(Y)$ and $\cos(Y)$ where Y has a uniform distribution are:

$$\text{var}[\sin(Y)] = \text{var}[\cos(Y)] = \int_0^{2\pi} \frac{1}{2\pi} \{\cos(Y)\}^2 \, dY = \frac{1}{2\pi}(\pi) = \frac{1}{2} \tag{141}$$

Thus, the variances of $\mathcal{S}_{pk}$ and $\mathcal{C}_{pk}$ are:

$$\sigma_{\mathcal{S}}^2 = \sigma_{\mathcal{C}}^2 = N_A \times \text{var}[\sin(Y)] = \frac{1}{2} N_A \tag{142}$$

Combining equations (88), (138), and (139) yields:

$$\Upsilon_{pk} = \frac{1}{2}\left(\mathcal{S}_{pk}^2 + \mathcal{C}_{pk}^2\right) \tag{143}$$

Since both $\mathcal{S}_{pk}$ and $\mathcal{C}_{pk}$ can be approximated by the same Gaussian distribution, Υ_{pk} will have a central χ^2 distribution with two degrees of freedom and parameters [21, pp. 26]:

$$\mu_{\Upsilon} = 2\left(\frac{1}{2}\sigma_{\mathcal{S}}^2\right) = \frac{1}{2} N_A \tag{144}$$

$$\sigma_{\Upsilon}^2 = 4\left(\frac{1}{2}\sigma_{\mathcal{S}}^2\right)^2 = \frac{1}{2} N_A^2 \tag{145}$$

Now consider the interference power expression given in equation (90).

$$P_{I_p} = \sum_{k \neq p} \Upsilon_{pk} \beta_k^2 = \beta^2 \sum_{k \neq p} \Upsilon_{pk} \tag{146}$$

We have assumed that β_k^2 is a constant over all k due to perfect power control. This is not a completely valid assumption as we will illustrate shortly.

Let V be the summation in (146) of $N_M - 1$ independent and identically distributed χ^2 random variables which can be approximated as Gaussian via the Central Limit Theorem.

$$\mu_V = (N_M - 1)\mu_\Upsilon = \frac{1}{2}(N_M - 1)N_A \tag{147}$$

$$\sigma_V^2 = (N_M - 1)\sigma_\Upsilon^2 = \frac{1}{2}(N_M - 1)(N_A)^2 \tag{148}$$

Introducing the voice activity factor to equation (146) allows P_{I_p} to be written as:

$$P_{I_p} = \left(\psi + \frac{1-\psi}{\nu}\right)\beta^2 V \tag{149}$$

If perfect beamforming is assumed, then $\phi_{pi} = \alpha_{pi}$ and the received power from the desired mobile will be:

$$P_{M_p} = \frac{1}{2}N_A\beta^2 \tag{150}$$

Now consider finding the required received power levels for the desired E_b/N_0 ratio.

$$\frac{E_b}{N_0} = \frac{E[P_{M_p}]/R_B}{E[P_{I_p}]/B + \sigma_n^2} \tag{151}$$

Substituting for P_{M_p} and P_{I_p} from (150) and (149) respectively and solving for $E[\beta^2]$ yields:

$$E[P] = E\left[\beta^2\right] = \left(\frac{E_b}{N_0}\right)\sigma_n^2 E\left[\left\{\frac{N_A}{2R_B} - \frac{1}{B}\left(\frac{E_b}{N_0}\right)\left(\psi + \frac{1-\psi}{\nu}\right)V\right\}^{-1}\right] \tag{152}$$

Since the random variable V is in the denominator of a fraction together with other terms, the above expression cannot be simplified further. However, the probability distribution of V is known, so the desired expected value can be found via numerical integration.

Equation (149) implies that P_{I_p} will be Gaussian since V can be approximated by a Gaussian distribution. However, the assumption that β was equal for all mobiles in the multi-element scenario was incorrect. For a single antenna element with perfect power control, all of the mobiles will have the same received power at the base station. When an antenna array is used, the interference will be different for each user, so the corresponding received powers from each mobile will also vary, even though all of the users have the same E_b/N_0 ratio.

Figures 7 and 8 show frequency histograms of the interference power (P_{I_p}) from simulations for 50 mobiles and 3 and 7 antenna elements respectively. The right tails of the distributions in both figures are definitely stretched out more than the left sides and thus a Gaussian distribution does not provide a completely accurate model of the true situation.

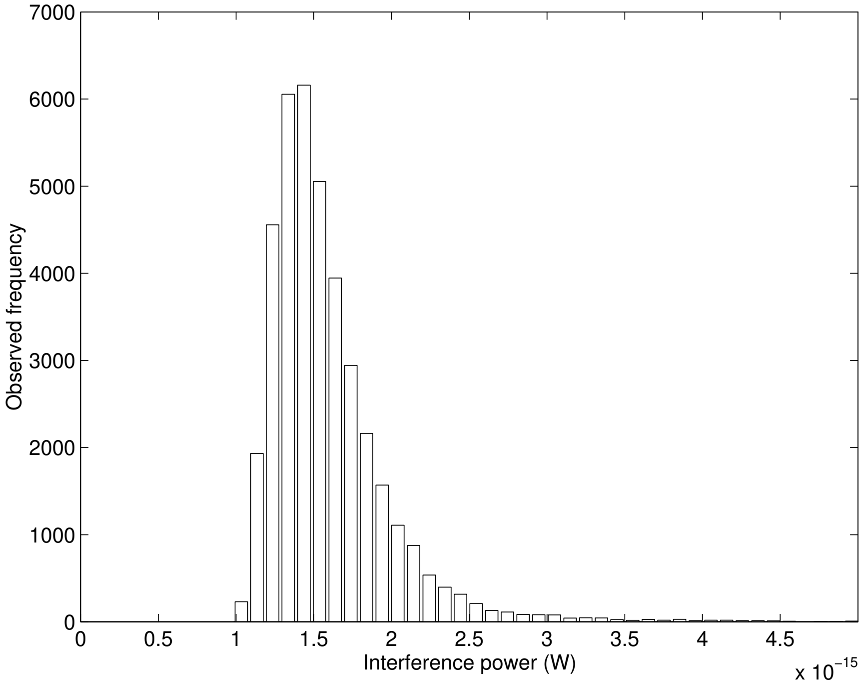

Figure 7: Interference power distribution for 3 antenna elements and 50 mobiles

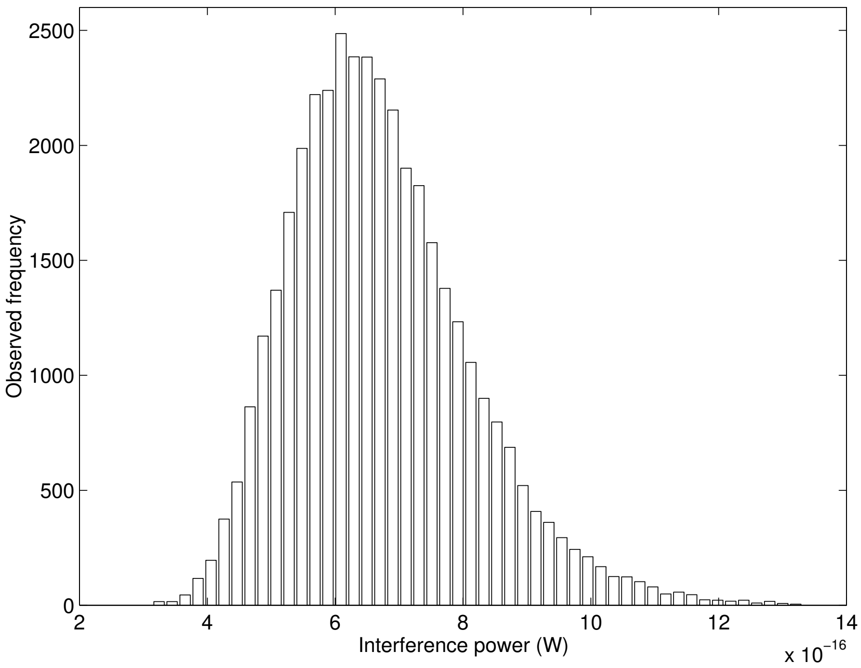

Figure 8: Interference power distribution for 7 antenna elements and 50 mobiles

8 Simulation Results

In this section, sample simulation results are presented to illustrate how the cell capacity of the system is affected by the number of base station antenna elements, the type of power control being used, and whether or not encoded bit interleaving is enabled. Results from both chip-level and power control capacity estimation simulations are included.

8.1 Chip-Level Capacity Simulations

Tables 1 and 2 contain chip-level results from our simulator under different conditions. In each case, the simulation lasted for 200 frames. This was limited by the computational expense involved, but sufficient numbers of data bits were processed to yield comparative estimates of the BER (bit error rate). Call lengths were exponentially distributed with a mean of 50 frames. The lengths of speech-on and speech-off times were also exponentially distributed with means of 10 frames. Thus, the effective speech activity factor was 50%. The simulations took place in a single hexagonal cell with a radius of 500 m, and mobiles were uniformly distributed within this area with their locations changing whenever a new call was started. In the case of multiple antenna elements, the spacing between the elements was equal to half the wavelength of the carrier frequency (2 GHz). Rayleigh fading was assumed, which is a worst-case scenario, and the fading gains were changed every 48 PN chips (roughly corresponding to the time of one data bit). Perfect beamforming was assumed for these simulations (*i.e.* the true array response vector was used to provide the beamforming weights). Power control methods A and B were both imperfect as discussed in Section 4.3. Method A had zero feedback delay and no probability of error in the feedback loop, whereas in method B the power control feedback bits were delayed by one power control period (1536 PN chips) and had a 10% probability of being inverted. The desired E_b/N_0 ratio was set at 7 dB, and the power control step size was 0.5 dB. Power transmitted by a mobile was reduced by a factor of 8 (corresponding to a data rate of 1200 bps) when speech was inactive. Only data bits transmitted during speech-on times were checked for errors.

Using a maximum BER of 10^{-3}, which is acceptable for voice communications, we have included the approximate capacity estimates for the different receiver permutations in Table 3. As would be expected, power control method B appears to reduce the capacity by a small amount. We note that interleaving also seems to reduce cell capacity slightly. Although

Power Control Method	A	A	B	B
Interleaving	N	Y	N	Y
Users				
28				0.00000
30	0.00000			0.00005
32	0.00003		0.00000	0.00010
34	0.00003	0.00000	0.00011	0.00081
36	0.00085	0.00032	0.00060	0.00013
38	0.00044	0.00192	0.00113	0.00285
40	0.00190	0.00298	0.00394	0.00482
42	0.00380	0.00441		

Table 1: Chip-level simulation results for 1 antenna element

Power Control Method	A	A	B	B
Interleaving	N	Y	N	Y
Users				
60		0.00000	0.00573	0.02147
62	0.00037	0.00098	0.00493	0.01733
64	0.00105	0.00023	0.00602	0.02774
66	0.00213	0.00032	0.01035	0.03869
68	0.00233	0.00065	0.01046	
70	0.00320	0.00408	0.01440	
72		0.00181		
74		0.00226		

Table 2: Chip-level simulation results for 2 antenna elements

interleaving is designed to combat the effects of bursty errors, the performance loss could be due to the suboptimal interleaved Viterbi decoding as described in Section 3.4.4. The use of two antenna elements appears to yield approximately a 50% increase in the number of mobiles versus the single antenna case.

We expect that capacity results for the case of imperfect beamforming (where the array response vectors are estimated from the received signal data) should be lower than the values presented here which are for perfect beamforming. This is due to the fact that inexact estimates of the array response vectors will not result in as great a reduction of the interference from other mobiles.

It should also be noted that due to the random speech-on and speech-off times, the number of active data bits can actually be smaller for a larger number of mobiles in some instances. This explains why occasionally the BER may appear to decrease when the number of users increases and possibly also explains some of the slight discrepancies in the simulated capacity figures.

8.2 Power Control Method Simulations

Table 4 contains capacity estimations obtained via the power control method for a voice activity factor of 1. These simulations were performed over 10000 frames. The call lengths were exponentially distributed with a mean of 10 frames. This was short enough to ensure that mobiles would be randomly moved around at regular intervals so that an accurate estimate of the cell capacity could be obtained. Cell capacity was defined as the maximum number of mobiles which could be supported with perfect power control saturating less than 1% of the time.

Table 5 contains similar capacity estimates for a voice activity factor of 0.5. In this situation, the speech on and speech off times were also exponentially distributed with means of 4 frames each. As can be seen, using a voice activity factor which is more realistic for voice communications results in a significant increase in the number of users that can be accommodated within the cell.

Figure 9 contains the same capacity estimates in a graphical form. The solid line represents the capacity values for a voice activity factor $\psi = 0.5$ and the dashed line is for $\psi = 1.0$.

One major difference between the two different voice activity values, $\psi = 1.0$ and $\psi = 0.5$,

Antenna Elements	Power Control Method	Interleaving	Cell Capacity
1	A	N	39
		Y	37
	B	N	37
		Y	37
2	A	N	63
		Y	69
	B	N	< 60
		Y	< 60

Table 3: Approximate cell capacities from chip-level simulations

Antenna Elements	Capacity	% of 1 Element Cap.
1	26	100%
2	40	154%
3	59	227%
4	71	273%
5	101	388%
6	121	465%

Table 4: Power control capacity estimates ($\psi = 1.0$)

Antenna Elements	Capacity	% of 1 Element Cap.
1	36	100%
2	61	169%
3	92	256%
4	115	319%
5	161	447%

Table 5: Power control capacity estimates ($\psi = 0.5$)

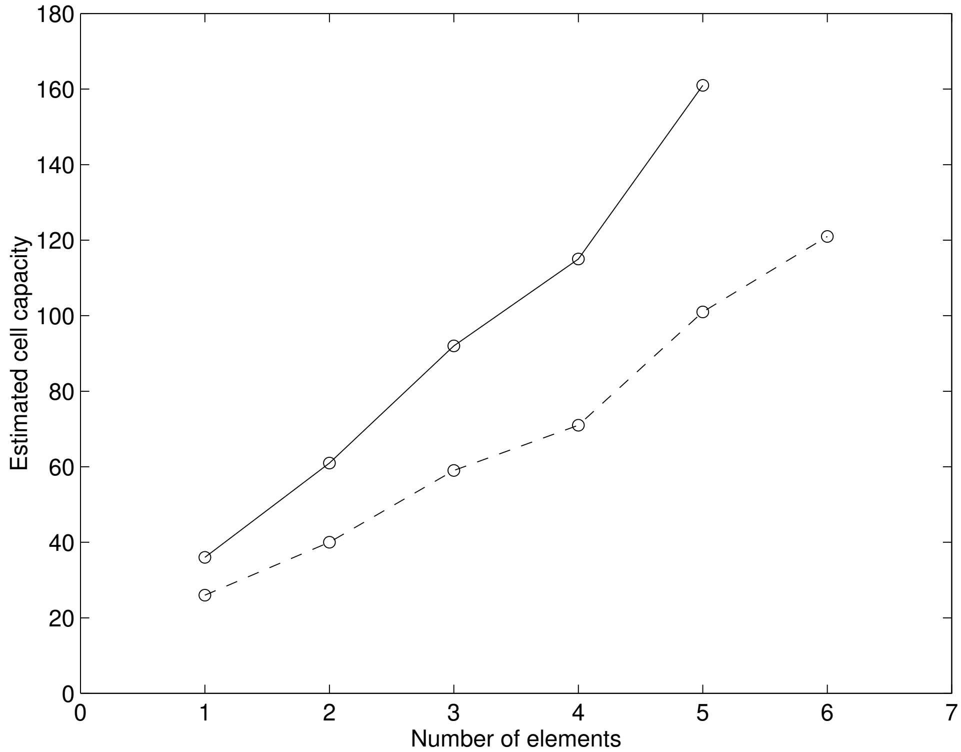

Figure 9: Power control capacity estimates for $\psi = 0.5$ (solid) and $\psi = 1.0$ (dashed)

is that in the latter case, the interference power is not only a function of the spatial locations of the mobiles but also of the number of users who currently have active speech. This number should follow a binomial distribution. A smaller voice activity factor implies that more mobiles can be accommodated in the same cell. However, it is necessary to take into consideration the fact that a significantly above average number of users with active speech at the same time will increase the interference power seen by each user and thus increase the probability of exceeding the cell capacity.

8.3 System Validation Results

This section contains sample results based on the derivations contained in Section 7. Here, the observed and predicted values of various correlation quantities are compared. A close agreement between the two sets of values tends to support the validity of the simulation program.

Table 6 contains a set of results comparing observed and predicted values. These were generated from a simulation of 20 mobiles with 1 antenna element over a simulation length

Quantity	Observed Value	Predicted Value
Average mobile power	7.6068×10^{-16}	7.6103×10^{-16}
Mean interference power	1.4455×10^{-14}	1.4460×10^{-14}
Mean interference and noise	1.9432×10^{-14}	1.9436×10^{-14}
Mean PN chip correlation	9.9382×10^{-15}	9.9480×10^{-15}
Variance PN chip correlation	2.4364×10^{-27}	2.4200×10^{-27}
Mean Walsh chip correlation	7.9505×10^{-14}	7.9584×10^{-14}
Variance Walsh chip correlation	2.0986×10^{-26}	1.9360×10^{-26}
Mean Walsh function correlation (correct)	5.0883×10^{-12}	5.0934×10^{-12}
Mean Walsh function correlation (incorrect)	5.2634×10^{-16}	0.0000
Variance Walsh function correlation	1.3252×10^{-24}	1.2390×10^{-24}

Table 6: Observed and predicted statistical parameters of correlation values ($\psi = 1.0$)

Quantity	Observed Value	Predicted Value
Average mobile power	3.4597×10^{-16}	3.3509×10^{-16}
Mean interference power	6.5740×10^{-15}	6.3667×10^{-15}
Mean interference and noise	1.5421×10^{-14}	1.1343×10^{-14}
Mean PN chip correlation	6.6693×10^{-15}	6.6010×10^{-15}
Variance PN chip correlation	1.2591×10^{-27}	1.2193×10^{-27}
Mean Walsh chip correlation	5.3354×10^{-14}	5.2808×10^{-14}
Variance Walsh chip correlation	1.0764×10^{-26}	9.7541×10^{-27}
Mean Walsh function correlation (correct)	3.4147×10^{-12}	3.3797×10^{-12}
Mean Walsh function correlation (incorrect)	-5.2077×10^{-17}	0.0000
Variance Walsh function correlation	6.7995×10^{-25}	6.2426×10^{-25}

Table 7: Observed and predicted statistical parameters of correlation values ($\psi = 0.5$)

of 20 frames. The equivalent voice activity factor was 1.0. As can be seen, there is close agreement between the two columns.

Table 7 contains another set of results comparing observed and predicted values but with a voice activity factor of 0.5. These were generated from a simulation of 20 mobiles with 1 antenna element over a simulation length of 1000 frames. The much longer simulation length was chosen since the number of speech-active mobiles in the system for $\psi < 1.0$ is a variable quantity. Therefore, a larger number of frames were required in order to obtain accurate observations from the simulation. As can be seen, there is reasonable agreement between the two columns, although perhaps not quite as good as in Table 6 for some quantities. This could be partially due to the fact that the observed voice activity factor from the simulation run was actually $\psi = 0.507$ which was slightly higher than the value used for generating the predicted numbers.

8.4 Imperfect Beamforming

In equation (98), if the spreading factor G between the Walsh and PN chips is sufficiently large, the array response vector can be well-approximated as the principal eigenvector of $\mathrm{R}_{\mathbf{zz}}$ since $GP_p\mathbf{a}_p\mathbf{a}_p^*$ will be the dominant component of this matrix. In IS95, this gain factor is only 4 which is too low to obtain a satisfactory estimate of $\mathbf{a}_p$ using this approach. However, we can modify the IS95 coding scheme so that there are only 8 different Walsh functions instead of 64 with 3 convolutionally encoded bits per Walsh function instead of 6. This quadruples the period of each Walsh chip and thus allows for a spreading factor of 16 between the Walsh chips and PN chips without changing the frequency of the final signal. As a result, improved estimates of the array response vectors can be obtained with less computational expense. Preliminary simulation results for beamforming indicate that significantly better array response vector estimates are obtained for larger spreading factors when the principal eigenvector approach is used. Table 8 contains sample simulation results for a spreading factor of 16 instead of the standard IS95 value of 4. These results indicate that changing the coding scheme parameters from the IS95 specification does not appear to affect the BER significantly.

Power Control Method	Interleaving	Users	Data Bits	BER
A	N	30	623,576	0.00001
		32	641,608	0.00003
		34	691,656	0.00023
		36	752,928	0.00029
		38	766,728	0.00150
		40	776,848	0.00202

Table 8: Simulation results for 1 antenna element with a spreading factor of 16

9 Ongoing Research

We are currently performing a statistical error analysis which should allow us to predict the expected bit error rate for a given number of mobiles. This includes an error analysis of the beamforming coefficient estimation process based on work originally from [2] as discussed in Section 7. Estimating these weights primarily involves determining the principal eigenvector of a correlation matrix which has been corrupted by noise. Agreement between the predicted and simulated values will aid in validating the observed results from the simulations.

Since calculating eigenvectors can be a computationally expensive process, we are also investigating more efficient methods for estimating beamforming weights. This can be accomplished by updating current eigenvector estimates based on new observation data, rather than completely recomputing the eigenvectors.

Due to computational limitations, our current simulations are only for a single cell. We intend to develop an efficient method for modeling interference from mobiles in adjacent cells so that a more realistic scenario can be simulated.

The chip-level simulation is currently required to generate data for performing imperfect beamforming. However, this approach is computationally expensive. If we can analytically derive the probability distribution function for imperfect beamforming weights (*i.e.* the eigenvectors), then it should be possible to estimate cell capacities for imperfect beamforming using the power control method which will be much more efficient. In particular, this will allow us to conduct simulations with more antenna elements and more cells.

Finally, beamforming on the forward link will also be investigated in the future. It is anticipated that this should not be a difficult task since much of the theoretical work and implementation details have already been performed for the reverse link situation.

10 Summary

The design methodology of implementing a chip-level CDMA simulator program has been presented. The level of detail in this simulator allows tasks such as digital beamforming and imperfect power control to be accurately tested. The simulation covers the entire reverse link of a CDMA system, including data encoding, transmission channel modeling, and signal decoding. Special emphasis was placed on power control and beamforming and their effects on cell capacity.

Two techniques for estimating cell capacity via simulations were discussed. The chip-level approach is more realistic, but is computationally expensive. A capacity estimation technique based on the power control levels was introduced as a more efficient method which can model greater numbers of mobiles and antenna elements.

An error analysis which served to validate the performance of the simulator was presented. This assumed perfect power control and a single base station antenna, but aided in verifying that the simulation results agreed with predicted values.

We included preliminary simulation results from a chip-level simulation of an IS95 compliant system. These simulations allow us to estimate cell capacities in a CDMA communication system. The chip-level simulation permits us to more accurately compare various beamforming strategies which are designed to increase cell capacity. Preliminary simulation results have demonstrated about a 50% improvement in capacity when two antenna elements are used instead of one. Capacity results from the power control estimation technique were in good agreement with those obtained from chip-level simulations. In addition, it appears that a modified version of IS95 with a larger spreading factor yields more accurate beamforming estimates.

Finally, the areas of our ongoing and intended future research were discussed. One main goal is to use the current simulation program as a test bed for investigating various beamforming strategies.

References

[1] S. P. Applebaum. "Adaptive arrays". *IEEE Transactions on Antennas and Propagation*, 24(5):585–598, 1976.

[2] A. M. Earnshaw and S. D. Blostein. "The performance of camera translation direction estimators from optical flow: Analysis, comparison, and theoretical limits", 1995. To appear in *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[3] D. Gerlach and A. Paulraj. "Base-station transmitting antenna arrays with mobile to base feedback". In *27th Asilomar Conference on Signals, Systems and Computers*, pages 1432–1436, 1993.

[4] K. S. Gilhousen, I. M. Jacobs, R. Padovani, A. J. Viterbi, J. Lindsay A. Weaver, and C. E. Wheatley III. "On the capacity of a cellular CDMA system". *IEEE Transactions on Vehicular Technology*, 40(2):303–312, 1991.

[5] A. Jalali and P. Mermelstein. "Effects of diversity, power control, and bandwidth on the capacity of microcellular CDMA systems". *IEEE Journal on Selected Areas in Communications*, 12(5):952–961, 1994.

[6] L. M. A. Jalloul and J. M. Holtzman. "Performance analysis of DS/CDMA with non-coherent M-ary orthogonal modulation in multipath fading channels". *IEEE Journal on Selected Areas in Communications*, 12(5):862–870, 1994.

[7] C. Kchao and G. L. Stüber. "Analysis of a direct-sequence spread-spectrum cellular radio system". *IEEE Transactions on Communications*, 41(10):1507–1516, 1993.

[8] J.-W. Liang and A. J. Paulraj. "On optimizing base station antenna array topology for coverage extension in cellular radio networks". In *1995 IEEE 45th Vehicular Technology Conference*, pages 866–870, 1995.

[9] R. A. Monzingo and T. W. Miller. *Introduction to Adaptive Arrays*. John Wiley & Sons, Inc., 1980.

[10] A. F. Naguib. *Adaptive Antennas for CDMA Wireless Networks*. PhD thesis, Dept. of Electrical Engineering, Stanford University, 1995.

[11] A. F. Naguib and A. Paulraj. "Effects of multipath and base-station antenna arrays on uplink capacity of cellular CDMA". In *1994 Global Communication Conference*, 1994.

[12] A. F. Naguib and A. Paulraj. "Performance of CDMA cellular networks with base-station antenna arrays". In *1994 International Zurich Seminar on Digital Communications*, pages 87–100, 1994.

[13] A. F. Naguib and A. Paulraj. "Performance enhancement and trade-offs of smart antennas in CDMA cellular networks". In *1995 IEEE 45th Vehicular Technology Conference*, pages 40–44, 1995.

[14] A. F. Naguib and A. Paulraj. "Performance of DS/CDMA with M-ary orthogonal modulation cell site antenna arrays". In *1995 International Conference on Communications*, pages 697–702, 1995.

[15] A. F. Naguib and A. Paulraj. "Recursive adaptive beamforming for wireless CDMA". In *1995 International Conference on Communications*, pages 1515–1519, 1995.

[16] A. F. Naguib and A. Paulraj. "Performance of wireless CDMA with M-ary orthogonal modulation and cell site antenna", 1996. To appear in *IEEE Journal on Selected Areas in Communications.*

[17] A. F. Naguib, A. Paulraj, and T. Kailath. "Capacity improvement with base-station antenna arrays in cellular CDMA". *IEEE Transactions on Vehicular Technology*, 43(3):691–698, 1994.

[18] A. F. Naguib, A. Paulraj, and T. Kailath. "Performance of CDMA cellular networks with base-station antenna arrays: The downlink". In *1994 International Conference on Communications*, pages 795–799, 1994.

[19] A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, Inc., 1984.

[20] S. U. Pillai. *Array Signal Processing.* Springer-Verlag, 1989.

[21] J. G. Proakis. *Digital Communications.* McGraw-Hill, Inc., 1989.

[22] Qualcomm Inc. Mobile station-base compatibility standard for dual-mode wideband spread system, 1993.

[23] G. L. Stüber and C. Kchao. "Analysis of a multiple-cell direct-sequence CDMA cellular mobile radio system". *IEEE Journal on Selected Areas in Communications*, 10(4):669–679, 1992.

[24] B. Suard, A. F. Naguib, G. Xu, and A. Paulraj. "Performance of CDMA mobile communication systems using antenna arrays". In *1993 International Conference on Acoustics, Speech, and Signal Processing*, volume VI, pages 153–156, 1993.

[25] S. C. Swales, M. A. Beach, D. J. Edwards, and J. P. McGeehan. "The performance enhancement of multibeam adaptive base-station antennas for cellular land mobile radio systems". *IEEE Transactions on Vehicular Technology*, 39(1):56–67, 1990.

[26] B. D. Van Veen and K. M. Buckley. "Beamforming: A versatile approach to spatial filtering". *IEEE ASSP Magazine*, pages 4–24, April 1988.

[27] J. Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford University Press, 1965.

[28] R. E. Ziemer and R. L. Peterson. *Digital Communications and Spread Spectrum Systems.* Macmillan Publishing Company, 1985.