CHANNEL ALLOCATION FOR BROADBAND FIXED WIRELESS ACCESS NETWORKS

A dissertation submitted to the University of Cambridge in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Shin Horng Wong, Queens' College *May 2003*

Laboratory for Communication Engineering
Department of Engineering
University of Cambridge

DECLARATION

I, Shin Horng Wong of Queens' College hereby declare that this dissertation and the work reported in it are originated entirely by myself in the Laboratory for Communication Engineering, University of Cambridge. Apart from those indicated in the report, the contents are entirely original and are not the outcome of work done in collaboration. I further state that no part of this dissertation has been submitted to another university. The length of this dissertation does not exceed 65,000 words.

Shin Horng Wong

Date: 12 May 2003

ACKNOWLEDGEMENT

I would like to thank my supervisor Dr. Ian J. Wassell for his guidance, patience and support, during my research at the Laboratory for Communication Engineering, University of Cambridge. I would also like to thank my advisor Dr. Martin Brown for his advice, and Professor Andy Hopper for giving me an opportunity to pursue my research.

I would also like to thank my parents and my wife for their support and understanding. I would like to acknowledge the help of my colleagues Viraj Abhayawardhana, Chong Eng Tan, Kam Sanmugalingam, Diego López de Ipiña, and Chia Leong Hong.

I am indebted and grateful to the Cambridge Commonwealth Trust and Adaptive Broadband Limited, for their generous sponsorship of my research. I would like to thank Queens' College and the Department of Engineering, for their bursaries during the final term of the research in Cambridge. The research has also benefited from the use of OPNET Modeler simulation tool, which was sponsored by OPNET Technologies.

Finally, I would like to thank the Laboratory for Communication Engineering, the Department of Engineering and Queens' College for their generous funding, that enabled me to publish and present my work in international conferences.

ABSTRACT

This dissertation presents research conducted concerning the channel allocation problem that is associated with wireless network planning. Effective channel allocation is able to reduce the level of interference, reduce the call blocking probability and increase the effective data throughput of a wireless data network. Channel allocation is a NP-complete problem and the increase in demand for wireless services has increased the base station density of networks thereby increasing the difficulty of the channel allocation problem. Most of the existing channel allocation methods are designed for circuit-switched cellular networks. This research concentrates on channel allocation for a packet based Broadband Fixed Wireless Access (BFWA) network offering a data service.

Three new channel allocation methods namely Fixed Channel Allocation using a Genetic Algorithm (FCA-GA), Dynamic Channel Allocation using Game Theory (DCA-GT) and Dynamic Channel Allocation using a Genetic Algorithm (DCA-GA) are proposed for use in a BFWA system. Three popular Dynamic Channel Allocation (DCA) methods namely Random Channel Allocation (RND), the Least Interfered (LI) method and Channel Segregation (CS) that have been proposed previously for use in cellular networks, are implemented for comparative purposes in a BFWA network. Simulations are performed using both the newly proposed and the existing channel allocation methods. It has been found that the performance of a channel allocation method is proportional to the amount of information available to the system and also to how effectively this information is used. FCA-GA uses complete a priori information of the network and exhibits the best performance while RND, which uses the least information, has the worst performance. FCA-GA and RND are used as the best and worst case benchmark performances when evaluating the DCA methods, where FCA-GA has an overall 1-percentile (1%) SNR gain of 28 dB over RND. The two newly proposed DCA schemes, namely DCA-GT and DCA-GA are the best performing DCA schemes with overall 1% SNR gains of 22.5 dB and 15.5 dB respectively over RND. These results compare with overall 1% SNR gains of about 5 dB and 9.5 dB over RND for the existing LI and CS schemes respectively.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF PUBLICATIONS

Part of this dissertation has been published in the following publications:

- Shin Horng Wong and Ian J. Wassell, "Performance Evaluation of a Packet Reservation Multiple Access (PRMA) Scheme for Broadband Fixed Wireless Access," in *Proc. London Communications Symposium 2001*, September 10-11, 2001, London, UK, pp. 179-182.
- Shin Horng Wong and Ian J. Wassell, "Application of Game Theory for Distributed Dynamic Channel Allocation," in *Proc. IEEE 55th Vehicular Technology Conference, Spring 2002*, May 6-9, 2002, Birmingham, AL, USA, vol. 1, pp. 404-408.
- Shin Horng Wong and Ian J. Wassell, "Distributed Dynamic Channel Allocation Using Game Theory for Broadband Fixed Wireless Access," in *Proc. 2002 International Conference on Third Generation Wireless and Beyond*, May 28-31, 2002, San Francisco, pp. 304-309.
- Shin Horng Wong and Ian J. Wassell, "Dynamic Channel Allocation for Interference Avoidance in a Broadband Fixed Wireless Access Network," in *Proc. 3rd International Symposium on Communication Systems Networks and Digital Signal Processing (CSNDSP'02)*, July 15-17, 2002, Staffordshire University, UK, pp. 352-355.
- Shin Horng Wong and Ian J. Wassell, "Dynamic Channel Allocation Using a Genetic Algorithm for a TDD Broadband Fixed Wireless Access Network," in *Proc. IASTED International Conference in Wireless and Optical Communications*, July 17-19, 2002, Banff, Alberta, Canada, pp. 521-526.
- Shin Horng Wong and Ian J. Wassell, "Channel Allocation for Broadband Fixed Wireless Access," in *Proc. 5th International Symposium on Wireless Personal Multimedia Communications (WPMC'02)*, October 27-30, Honolulu, Hawaii, USA, pp. 626-630.

GLOSSARY

1% SNR	1 percentile SNR
3G	Third Generation mobile system
AMPS	Advanced Mobile Phone Service
AP	Access Point
ATM	Asynchronous Transfer Mode
BDCL	Borrowing with Directional Channel Locking
BER	Bit Error Rate
BFWA	Broadband Fixed Wireless Access
BLER	Block Error Rate
bps	bits per seconds
CDD	Code Division Duplex
CDF	Cumulative Distribution Function
CDPD	Cellular Digital Packet Data
CS	Channel Segregation - DCA
CSMA	Carrier Sense Multiple Access
CSVR	Control Server
CS-*X*dB	Channel Segregation with a SNR threshold of *X* dB
CT	Contention Timeslots
CT2	2nd generation cordless phone
CW	Continuous Wave
DACK	Downlink Acknowledgement packet in PRMA
dB	Decibels
dBi	Gain in dB relative to an isotropic antenna
dBm	Power in dB relative to one milliwatt
DCA	Dynamic Channel Allocation
DCA-GA	DCA using Genetic Algorithm
DCA-GT	DCA using Game Theory
DCA-LSWO	DCA using Limited Search with Weight Ordering
DCA-WCAR	DCA with Weight Carrier Ordering
DCA-WCHN	DCA with Weighted Channel Ordering
DCELL	Downlink data packet in defined in a PRMA MAC structure
DECT	Digital Enhanced Cordless Telecommunications
ESRA	Enhanced Staggered Resource Allocation
FA	First Available DCA using interference power threshold
FA-SIR	First Available DCA using an SIR threshold
FCA	Fixed Channel Allocation
FCA-GA	FCA using Genetic Algorithm
FDD	Frequency Division Duplex
HCA	Hybrid Channel Allocation
HNN	Hopfield Neural Network
HTA	Highest Interference below Threshold Algorithm
Hz	Hertz
ISM2	Industrial Scientific Medical band at 2 GHz
ISM5	Industrial Scientific Medical band at 5 GHz
LCA	Lowest Channel below Threshold Algorithm

LI	DCA using the Least Interfered method
LMDS	Local Multipoint Distribution Service
LODA	Locally Optimised Dynamic Assignment
LTA	Least Interfered below Threshold Algorithm
m	metres
MAC	Media Access Control
MAXAVAIL	Maximum available DCA
MAXMIN	Fully centralised DCA that Maximises the Minimum SIR
Mbps	Megabits per second
MHz	Mega Hertz
MIA	Marginal Interference Algorithm
MP	Maximum Packing
NA-LI	Network Assisted Least Interfered method
NA-TO	Network Assisted DCA with Throughput Optimisation
nmsec	Normalised second - time taken to transmit a data packet (DCELL or UCELL)
NP-complete	Nondeterministic Polynomial time complete
PHS	Personal Handyphone System
PRMA	Packet Reservation Multiple Access
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
ReMax1	DCA that applies MAXAVAIL once if the first MAXAVAIL fails
ReMax2	DCA that applies MAXAVAIL once if the ReMax1 fails
RND	DCA using Random Channel Allocation
RR	Reservation Response in PRMA
SCAN	Channel measurement portion of the PRMA MAC employed in the BFWA
SIR	Signal to Interference Ratio
SNR	Signal to Noise Ratio
SONN	Self-Organising Neural Network
SRA	Staggered Resource Allocation
SU	Subscriber Unit
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TSDP	Two Step Dynamic Priority DCA
UACK	Uplink Acknowledgement packet in PRMA
UCELL	Uplink data packet in defined in a PRMA MAC structure
UCELLR	Uplink data packet in defined in a PRMA MAC structure with reservation
Wi-Fi	Wireless Fidelity

MATHEMATICAL NOTATION

K	Capital and italic letters denote constants
j	Small caps and italic letters denote variables or indexes
$\boldsymbol{F}()$	Capital, italic and bold letters denote functions
$\mathbf{S}$	Bold letters denote sets or vectors (small caps)
$\mathbf{\Lambda}$	Capital and bold symbols denote matrixes
$\lvert\mathbf{S}\rvert$	Denotes the number of elements in set **S**
$\cap$	Intersection between two sets
$\varnothing$	Null set
B	Total number of base stations in a network
$\mathbf{B}$	Set of base stations in a network
b_j	Base station j (an element of set of base stations **B**)
C	Total number of channels in a system
$\mathbf{X}$	Compatibility matrix
$\mathbf{C}$	Set of channels in a system
$\boldsymbol{C}(\mathbf{s})$	Cost of using solution **s**
C_{ALC}	Average lowest cost within a specific range of generations in FCA-GA
C_{AVG}	Average of the average cost over a specific range of generations in FCA-GA
C_{DCA}	The cost of using a specific channel assignment calculated using $\boldsymbol{F}_{FCA}(t)$
C_{GA}	Number of channels in the priority list of an AP using DCA-GA
$\chi_{j,k}$	Required channel separation between base station j and k (element of **X**)
c_k	Channel number k (an element of set of channels **C**)
C_L	Theoretical lower bound number of channels for the channel allocation problem
C_{MIN}	Minimum number of channels required to solve the channel allocation problem
C_{SCAN}	Number of channel measurements required in the SCAN portion in a DCA
D	Total traffic demand of load
$\mathbf{D}$	Traffic demand vector
D_{AVG}	Diversity of a population in FCA-GA
$\boldsymbol{D}_{\mathbf{B}}(j)$	Degree of difficulty of assigning channels to base station j in a set of base stations **B**
d_j	Traffic demand at base station j (an element of traffic demand vector **D**)
D_{LI}	Probability of two APs detecting the channel usage of each other using LI
$\boldsymbol{e}$	Exponential function
$\boldsymbol{exp}$	Exponential function
$\boldsymbol{F}(\mathbf{s})$	Fitness function for string (or solution) **s**
$\boldsymbol{F}_{DCA}(t)$	Fitness function in DCA-GA
$\boldsymbol{F}_{FCA}(t)$	Fitness function in FCA-GA
F_{MAC}	Number of MAC frames between two SCANs
$\boldsymbol{f}_u(t)$	Unit step function
γ	Time (nmsec) required to measure a single channel
$\boldsymbol{\Gamma}(T_{TX})$	Required transmit portion for a peak payoff given the other AP's T_{TX}
G_{MAC}	Data throughput - amount of time spent transmitting per nmsec
$\in$	Belongs to a specific set

$\notin$	Does not belongs to a specific set
λ	Wavelength
log	Logarithm function (base 10)
μ	Fraction factor in Elitism selection - in Genetic Algorithm context
M_{MAC}	Maximum length (nmsec) of a single PRMA MAC frame in the BFWA
$\mathbf{N}(\mathbf{s})$	Set of solutions in the neighbourhood of **s**
N_c	Cluster size
$\tilde{N}_c$	Average number of cluster size
$\mathbf{N}_{\mathrm{Co}}(b_j)$	Set of base stations that are the nearest co-channel neighbours of base station j
$\mathbf{N}_{\mathrm{I}}(b_j)$	Set of base stations that is within the interference neighbourhood of base station j
n_{RX}	Receiver's noise
N_{SU}	Maximum number of SUs allowed to transmit in each PRMA MAC frame
$P_{\geq 21}$	Probability of packet received with SNR at least 21 dB
π_{AVG}	Average effective data throughput
p_c	Crossover rate in Genetic Algorithm
P_{C_LI}	Probability of two APs using the same channel in LI
P_{I_LI}	Probability of two APs interfering each other in LI
P_{I_RND}	Probability of two APs interfering each other in RND
$\Pi_j(t)$	Peak payoff for AP j
$\pi_{j,k}(t)$	Payoff function at AP j with respect to AP k at time t
$\pi_{LI}(t)$	Payoff for an AP in a network using LI at time t
p_m	Mutation rate in Genetic Algorithm
p_{MIX}	Probability of an AP playing the explore strategy in DCA-GT
p^*_{MIX}	Probability p_{MIX} that would give the highest expected payoff Π_{MIX} in DCA-GT
Π_{MIX}	Expected payoff playing a mixed strategy in DCA-GT
P_{RX}	Received power
π_S	Saturation payoff in DCA-GT
$\mathbf{P}(t)$	Population (set of strings) at the t^{th} generation in Genetic Algorithm
P_{TX}	Transmit power
$\boldsymbol{R}(s)$	Best response strategy to strategy s
R_B	Cell radius
R_U	Reuse distance
SNR^*	Required SNR threshold that must be meet in a threshold based DCA
T_{EFF}	Effective transmit portion for an AP with varying T_{TX}
T_{INT}	Interference threshold that must be satisfied in a threshold based DCA
$\boldsymbol{T}_j(t)$	Transmit portion of AP j at time t - used in the calculation of $\pi_{j,k}(t)$
T_L	Transmit portion for an AP that plays explore strategy in DCA-GT
T_{MAC}	Packet throughput - number of UCELLs and UCELLR received at an AP per nmsec
T_T	Transmit portion for an AP that plays exploit strategy in DCA-GT
T_{TX}	Transmit portion - time between two channel measurements (SCAN)
ζ	Path-loss exponent

1 INTRODUCTION

In 1888, Heinrich Hertz successfully demonstrated the first transmission and reception of electromagnetic waves, and this lead to the birth of wireless communication, which has been used in radio and TV broadcasting, microwave links, satellite communications, mobile communications and network access.

The major advantage of wireless compared to wired communication is mobility. The most popular mobile communication device at the moment is the cellular phone. Like mains water and electricity, mobile communications has improved our lives considerably. At a smaller scale, mobile communication is found within gadgets and appliances for example TV infrared remote control and Bluetooth enabled devices that make it possible for these devices to form a network among themselves. Wireless communication is also widely used in fixed communications. Fixed communications include microwave links, wireless local loop and broadband fixed wireless access (BFWA) networks. The advantages of using fixed wireless link compared to wired link are:

- faster speed of deployment
- accessibility of difficult to reach areas
- low marginal cost and effort in adding or removing a subscriber compared to the sunk cost required to install cables for wired access.

Wireless communication also has its disadvantages. For example, a wireless signal can be received anywhere within the coverage area, which compromises security since it is easier to gain unauthorized network access. It is more difficult to transmit at a high data rate in a wireless channel than a wired channel since the wireless channel is more hostile.

1.1. Wireless Channel

In the 19th century, James Clerk Maxwell proposed that electromagnetic waves propagate through space via the ether in a similar manner to the propagation of sound (pressure) waves due to the movement of molecules in the medium. Using the concept of the ether, Maxwell's equations were developed to describe the propagation

of electromagnetic waves. He predicted the speed of electromagnetic waves to be similar to that of light and that light belongs to the family of electromagnetic waves. Although the concept of ether has been abandoned, Maxwell's equations are still being used to model electromagnetic propagations.

As with sound waves, electromagnetic waves can be reflected, diffracted and attenuated depending upon the medium and the size of the obstacles the wave encounters. Reflection occurs when a wave hits a surface of an object whose dimension is much larger than its wavelength and the reflected wave follows Snell's Law. Diffraction occurs when a wave hits the edge of an object or an object whose dimension is smaller than its wavelength. The wave will be scattered at the diffraction point.

An electromagnetic wave generated from a point source spreads out as it propagates further from its source. For an input power P_{IN} (Watts), the power density P_D (Watts/m^2) of this wave at a distance d is:

$$P_D = \frac{P_{IN}}{4\pi d^2} \tag{1.1}$$

If a receiver at distance d captures an area A_{RX} experiencing a certain power density, the receive power will be a fraction of the input power P_{IN}. The loss of power at the receiver is termed spatial attenuation. The electromagnetic wave may also face further attenuation due to absorption depending upon the medium it is traversing. For example, microwaves may be absorbed by water or foliage and hence experience large attenuation. The received wave may also lose power when it is diffracted or when reflection is not perfect.

A transmitted radio signal is an electromagnetic wave. Since the transmitted signal can be reflected and diffracted, it may go through many different paths before arriving at a receive antenna. The various signals arriving at the receive antenna may have different phases and powers (due to different attenuation). The received signal is said to have experienced multi-path propagation and the overall received signal is the sum (which may be destructive or constructive) of all these individual multi-path signals. Multi-path propagation causes different areas to have a non-uniform spatial

distribution of signal power and this phenomenon gives rise to so called multi-path fading, where the physical spacing between the fading depends upon the signal's transmitted wavelength.

1.2. Path Loss

In free space, the received signal power P_{RX} (in Watts) at a distance d from a source transmitted with power P_{TX} is [1]:

$$P_{RX} = P_{TX}\left(\frac{\lambda}{4\pi d}\right)^{\zeta} G_{RX} G_{TX} \tag{1.2}$$

Where, ζ is the path loss exponent (which is equal to 2 in free space propagation), λ is the wavelength of the transmitted signal, and G_{RX} and G_{TX} are the receiver and transmitter antenna gains respectively.

Free space propagation does not occur in most environments because of the presence of the ground, which acts as a reflective surface and also owing to physical variations (e.g. buildings or hills) between the transmitter and receiver. These variations cause shadowing, giving variations in the received signal strength. An empirical path loss model taking into account the effects of shadowing and fast fading, can be expressed as:

$$10\log\left(\frac{P_{RX}}{P_{TX}}\right) = \zeta 10\log\left(\frac{\lambda}{4\pi}\right) + 10\log\left(G_{RX}\right) + 10\log\left(G_{TX}\right) - \zeta 10\log\left(d\right) + \boldsymbol{F}_S\left(d\right) + \boldsymbol{F}_M\left(d\right) \tag{1.3}$$

where $\boldsymbol{F}_S(d)$ is the shadowing margin and $\boldsymbol{F}_M(d)$ is the multi-path fading margin. For radio signals where the wavelength is less than a metre, the received signal strength variation over a short distance owes more to multi-path fading than it does to shadowing. Multi-path fading and shadowing are also known as short-term fading and long-term fading respectively. In a typical radio environment, the path loss exponent is usually between 2 and 5 and long-term fading follows a log-normal distribution [1]. The short term fading or fast-fading follows a Rayleigh distribution

where there is no line of sight path between transmit and receive antennas and with a line of sight path it follows a Rician distribution.

1.3. Cellular Concept

A wireless operator is usually given a fixed block of frequency spectrum. In order to satisfy the traffic demand in a large geographical area, the spectrum needs to be reused. Mac Donald [2] introduced the cellular concept where the radio coverage area of a base station is represented by a cell. A regular hexagon is chosen to represent a cell because it covers a larger area with the same centre-to-vertex distance (or radius) compared to a square or an equilateral triangle. Consequently, fewer hexagonal cells are required to cover a given geographical area. These cells are placed in a cellular structure covering a geographical area as shown in Figure 1.1.

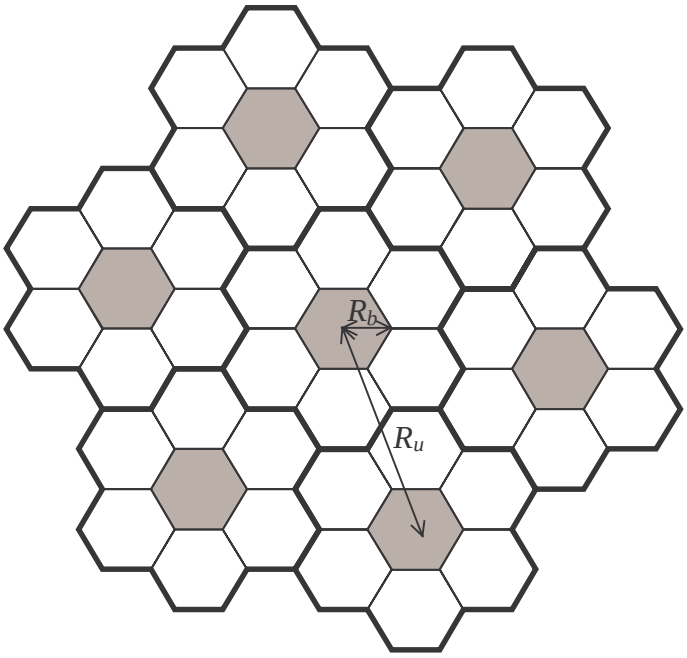

Figure 1.1: Cellular layout and frequency reuse

The cells in a geographical area are grouped into clusters. The entire block of frequencies is completely allocated to each cluster and the cells in each cluster use different frequencies. In this way, the limited block of frequency spectrum is reused. The concept of frequency reuse is shown in Figure 1.1, where the cluster size is 7 cells and a set of co-channel cells – i.e. cells using the same frequencies – are shown shaded. The co-channel reuse ratio is defined as:

$$\frac{R_u}{R_b} = \sqrt{3N_c} \tag{1.4}$$

where R_u is the distance between the two closest co-channel cells, R_b is the cell radius and N_c is a positive integer representing the number of cells per cluster.

1.3.1. Interference

Co-channel interference is the interference caused by users and base stations in co-channel cells. Since the received power is inversely proportional to the distance from the source, the co-channel interference power is dependent upon the distance of the co-channel cells. Therefore, from (1.4), the cluster size N_c is governed by the level of co-channel interference that can be tolerated.

Apart from co-channel interference, adjacent channel interference can also occur when signals from adjacent channels enter the receiver since the receiver is not perfect. A sufficiently wide guard band is required to reduce adjacent channel interference.

1.3.2. Duplex Communication

In most wireless systems, communication is usually two-way, that is the subscriber must be able to communicate with the base station and vice versa. Duplexing is used to permit two-way communication and it can be achieved using Frequency Division Duplex (FDD), Time Division Duplex (TDD) or the recently proposed Code Division Duplex (CDD). The choice of duplexing affects the amount of interference in a network.

Two separate frequencies are used in FDD, one for uplink communication (subscriber to base station) and the other for downlink communication (base station to subscriber). In FDD, interference can occurs in two ways: adjacent subscribers (subscribers from another co-channel or adjacent channel cell) to base station being interfered with (i.e. home base station) and adjacent base station to home subscriber. A system employing FDD requires two separate bands of frequencies and for asymmetric traffic (e.g. if downlink traffic is heavier than uplink traffic) these bands are not fully utilized.

TDD uses the same frequency for uplink and downlink communications. The subscriber and base station take turns to transmit and hence they must agree upon the timeslot for uplink and downlink transmission. In TDD four kinds of interference can occur: adjacent subscriber to home subscriber, adjacent base station to home base station, adjacent subscriber to home base station and adjacent base station to home subscriber. Hence, TDD has a worse interference performance than FDD. However, for asymmetric traffic TDD better utilizes the frequency spectrum compared to FDD as downlink and uplink transmission can occur consecutively on the same frequency band. TDD also eases frequency allocation issues for regulatory bodies since only a single block of frequencies is required as compared with FDD, which requires a pair of frequency bands [3]. Note that the SNR performance of a TDD system can be improved by having synchronization among the base stations. It is shown in [4] that the signal to interference power ratio (SIR) performance of a fully synchronous TDD system approaches that in a FDD system.

A CDD system – similar to a TDD system – uses the same frequency band for the uplink and downlink transmission. However, instead of separating the uplink and downlink transmission in time, CDD separates them using different smart codes [5]. A set of smart codes is able to maintain the required orthogonality property among codes even with a time shift in the received signal. This requires the smart codes to have a low value of auto-correlation and cross-correlation for the received signals that are time shifted less than the time delay spread. Theoretically, multipath effect will not affect a code with an auto-correlation of zero, while a cross-correlation of zero will eliminate multi-user interference. Hence, a system using CDD will not experience any interference and the SNR is only affected by receiver noise. For asymmetric traffic, CDD better utilizes the frequency spectrum compared to FDD.

1.4. Frequency Spectrum

With the growth in communications, there has been a huge increase in demand for frequency spectrum and consequently this has made it very valuable. For example, the British government enjoyed a gain of £22.5 billion on 27 April 2000 while the German government pocketed £30.8 billion on 17 August 2000 by auctioning the Third Generation (3G) Mobile Telephone Licenses to telecom operators [6]. Billions of pounds were gained by apparently selling air. The winners of the auction will incur

billions more pounds in debt in order to build the infrastructure required to provide the 3G Mobile Telephony services. Clearly, the high cost of obtaining a frequency spectrum creates pressure to use it effectively and efficiently.

Unlicensed frequency spectrum such as the ISM2 (2 GHz) or ISM5 (5 GHz) bands can be used free of charge. The unlicensed frequency spectrum (subject to regulatory constraints) allows anyone to act as an Internet Service Provider (ISP) and this has helped fuel the growth of Wireless Fidelity (Wi-Fi) networks. Wi-Fi uses the IEEE 802.11 standards and can be cheaply and easily set up to provide wireless data access. The Wi-Fi service can be free and it is becoming increasingly available in highly populated areas (Hot Spots) e.g. airports, restaurants, shopping malls and hotels. The estimated number of Wi-Fi users in 2002 is 2 million worldwide and this is expected to exceed 5.4 million by 2003 [7]. However, since anyone can use unlicensed spectrum, Wi-Fi base stations will experience high levels of interference from each other. Apart from interference from like equipment, Wi-Fi will also experience interference from other systems e.g. Bluetooth. Hence, effective methods are required to reduce interference.

Whether the frequency spectrum is licensed or unlicensed, it is limited and the operator needs to reuse it. Frequency reuse causes interference and this lowers the received Signal to Noise ratio (SNR), where noise includes both the receiver noise and the interference. For a system with a fixed capacity (bits per second), a poor SNR causes a higher bit error rate and lowers the overall effective data throughput. A poor SNR also decreases the capacity of a system with variable capacity (e.g. one using adaptive modulation). Therefore, reducing the interference will lead to a higher capacity and data throughput for a wireless system.

Since interference (introduced by frequency reuse) is caused by equipment in other cells using the same frequency, good channel allocation among the cells will reduce the overall level of interference.

1.1 Outline of this Dissertation

In this dissertation, the channel allocation problem will be further elaborated and three novel channel allocation schemes are proposed for use in a Broadband Fixed Wireless

Access (BFWA) network providing a data service. The dissertation is outlined as follows:

The current chapter (Chapter 1) has presented the need for channel reuse and how this led to the development of cellular concept. Interference is the major consequence of channel reuse and therefore there is a need for channel allocation methods in a wireless network.

Chapter 2 describes the channel allocation problem in detail and offers a comprehensive survey regarding existing methods. The large number of existing channel allocation methods are classified using a Channel Allocation Matrix, which categorises the various channel allocation methods based upon the techniques used by the system to obtain network information. It is proposed that the performance of a channel allocation method is proportional to the information available to the system and also to how effectively this information is used.

Chapter 3 details the BFWA system used as the basis of study. The simulation tool OPNET, which is used throughout this dissertation, is described and is used to investigate the performance of the Media Access Control (MAC) layer of the BFWA under consideration. The radio propagation and traffic models to be used are also described in this chapter.

In Chapter 4, Random Channel Allocation (RND) is implemented in the BFWA simulation model. A novel fixed channel allocation method using a genetic algorithm (FCA-GA) is also proposed. These two channel allocation methods are used as the lower and upper benchmarks for the performance of the investigated Dynamic Channel Allocation (DCA) schemes. The Least Interfered (LI) method is a popular DCA method that has been evaluated previously in many publications (see Chapter 2) and is shown to be effective in circuit switched networks. For the first time, results are presented for the LI method when it is applied in a BFWA network offering a data service. Simulations performed using RND, LI and FCA-GA show that the performance of the channel allocation methods is proportional to the amount of information available concerning the network.

Chapter 5 frames the distributed DCA problem as a game and an appropriate payoff function is proposed. The distributed DCA is analysed using game theory and a novel

DCA method, namely DCA Game Theory (DCA-GT) is proposed. Although DCA-GT uses the same quantity of network information as LI, it is shown via simulation that DCA-GT makes more effective use of this information by yielding an overall 1% SNR gain of 8.5 dB over LI.

Chapter 6 investigates threshold and priority based DCA schemes and applies the previously proposed Channel Segregation (CS) scheme to the BFWA model under consideration. CS is a popular DCA method that has been described in many publications (see Chapter 2) primarily in the context of circuit switched networks but it has not been applied previously to a BFWA network offering a data service. A novel DCA method, namely DCA-Genetic Algorithm (DCA-GA), is proposed for use in a partially centralised system in order to overcome some of the problems that exist with the CS scheme. In DCA-GA, a genetic algorithm is used to achieve a balance between exploring new channels, (with the potential to yield better solutions in the presence of dynamic interference), and the need to exploit a particular channel with the aim of achieving long term frequency coordination among the base stations in the network. The performance of CS and DCA-GA are both investigated using simulations in this chapter. It is found that CS gives an unbalanced SNR performance where the uplink has a good received SNR performance whilst the downlink experiences a poor SNR performance. DCA-GA is shown to be able to rectify this problem and give an overall 1% SNR gain of 13 dB over CS.

Finally, Chapter 7 presents some conclusion and suggests future work in this area.

To summarise, the novel algorithms and results presented in this thesis are:

- a FCA using a Genetic Algorithm (FCA-GA) in a BFWA network scenario
- a DCA using Game Theory (DCA-GT) in a BFWA network scenario
- a DCA using a Genetic Algorithm (DCA-GA) in a BFWA network scenario
- results for LI and CS when applied to a BFWA network offering a data service.

2 CHANNEL ALLOCATION

Channel allocation is one of many ways to reduce interference in a cellular network. Reduced interference leads to an increase in capacity and throughput of the system and hence good channel allocation can lead to more effective use of the spectrum. Apart from reducing interference, channel allocation algorithms can also be used to adapt to traffic changes in a network, and together with reduced interference, the traffic that can be supported is higher.

The task of channel allocation in a cellular system is to allocate C available channels to B base stations, where each base station j has a specific traffic demand (e.g. calls) that requires d_j channels (e.g. one channel per call). Hence, channel allocation is a permutation problem with a search space of

$$\frac{D!}{(D-C)!} \tag{2.1}$$

where D is the total traffic demand of all the B base stations (i.e. $D = \Sigma d_j$). If $C = 15$ and $D = 50$, there are about 2.94×10^{24} possible solutions. If a computer can evaluate a solution in 1 microsecond, it would take 93 billion years to evaluate all possible solutions. However, not all the solutions are valid. The same channel is allocated to two different base stations only if there is a sufficient distance between them such that the interference is tolerable. In order to ensure the interference between the co-channel cells is tolerable, the channel allocation solutions must conform to a Compatibility Matrix $\mathbf{X}$. The Compatibility Matrix gives the minimum required frequency (or channel) separation between two cells (or base stations) and it is defined as a $B \times B$ symmetric matrix where an element $\chi_{j,k}$ in the matrix is the minimum frequency separation required between cell j and k.

In this dissertation, a free channel is defined as an unused channel that will not violate the compatibility matrix if it is used. In a voice service, a call block occurs when a new call is denied service if no free channel is available at the serving base station. The number of radio transceivers required in a base station is dependent upon the

expected traffic and the grade of service (i.e. probability of call block). The traffic demand vector $\mathbf{D} = [d_1, d_2, \ldots d_B]$ is the number of radio transceivers needed in each base station.

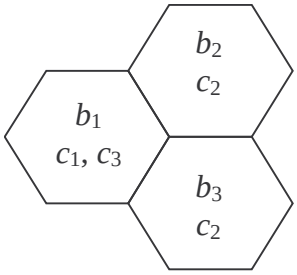

Figure 2.1: An example of a cellular network with 3 cells and a possible channel allocation solution.

For an example of a channel allocation problem, consider a cellular network with 3 cells, each with one base station and let the set of base stations $\mathbf{B} = \{b_1, b_2, b_3\}$ as shown in Figure 2.1. The cellular system requires a channel separation of at least 2 for calls within the same cell and 0 for calls in neighbouring cells. The Compatibility Matrix $\mathbf{X}$ is therefore:

$$\mathbf{X} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{2.2}$$

For this example, let the traffic demand (represented by a vector) be $\mathbf{D} = [2\ 1\ 1]$ and the set of channels (e.g. frequencies) be $\mathbf{C} = \{c_1, c_2, c_3\}$ where $|\mathbf{C}| = C$. Define $\mathbf{S}$ as the set of all solutions (i.e. all possible channel assignment) and the matrix $\mathbf{s}=[s_{j,k}]$ for $1 \le j \le B$, $1 \le k \le C$, where j and k are integers – is one of the solutions in $\mathbf{S}$ where an element $s_{j,k}$ in $\mathbf{s}$ is defined as:

$$s_{j,k} = \begin{cases} 1, & \text{if channel } k \text{ is assigned to base station } j \\ 0, & \text{otherwise} \end{cases} \tag{2.3}$$

A possible solution is shown in Figure 2.1 and the corresponding matrix $\mathbf{s}$ is:

$$\mathbf{s} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{2.4}$$

The Compatibility Matrix is formed on the basis that the interference between two cells using the same frequency is tolerable. In a voice oriented service, the quality of the received audio is important and it is considered to be acceptable for an AMPS system [2] if the SNR is at least 18 dB, which corresponds with a Signal to Interference Ratio (SIR) of at least 17 dB. For a given propagation channel and the required SIR, the co-channel reuse ratio can be determined. If the cells are laid out uniformly, the cluster size N_c can be found using (1.4).

The task of allocating a set of **C** channels to a set of **B** base stations with traffic demand **D** constrained by a Compatibility Matrix **X** is known as a Nondeterministic Polynomial time complete (NP-complete) problem. NP-complete is a decision problem that can be solved using a nondeterministic Turing Machine in polynomial time [8]. A decision problem gives only a yes or a no answer. In this case, the decision problem is whether there is a channel assignment to the given channel allocation problem that uses C or less channels. For a given channel allocation problem, the optimum solution is one that uses the least number of channels C_{MIN}. Finding the optimum solution is a NP-hard problem. The task of finding a solution (or an optimisation) increases exponentially with the size of the problem (e.g. the number of base stations). The Channel Allocation Problem can be transformed into a graph colouring problem [19], which is also a NP-complete problem. Since the Channel Allocation problem can be transformed to another known NP-complete problem, the Channel Allocation problem itself is NP-complete [8].

2.1 Handoff

When a call is in progress in a mobile cellular network, owing to the mobility of users, the SIR of the call may drop below the required threshold. If another base station can better serve the call (i.e. meet the required SIR), the call is handed off to this base station. Note that a call will be dropped if the desired new base station does not have any free channels to serve the handoff call. From the user's perspective, a dropped call is worse than a blocked call and therefore in assigning a channel, handoff

calls are often given priority over new calls [1]. To improve the call dropping probability the concepts of guard channels or a handoff queue can be used [18]. Guard channels are channels reserved in a base station to serve only handoff calls and hence reduce the call dropping probability. The remaining channels in the base station can be used to serve new calls and handoff calls with equal priority. However, the use of guard channel increases the call blocking probability as fewer channels are available to serve new calls. In a network with low handoff rate, it is clear that the channels are not utilised efficiently using this approach.

During handoff a call can still be served by the current base station until its signal strength becomes unacceptable. During this time interval, if no channels are available for handoff in the handoff base station, the handoff calls can be put into a queue until a channel becomes free. The handoff base station will block all new calls until the handoff queue is served. This method also reduces the blocking probability but it uses the channels more efficiently.

In [9] new calls are allowed to use the guard channels according to a request probability. The request probability is dependent upon the "call mobility" which is the ratio of the handoff call arrival rate to the new call arrival rate. If the "call mobility" is high, then the request probability is reduced and vice-versa. This method strikes a better balance between call blocking and call dropping.

The concept of a "shadow cluster" is introduced in [10] where the base station forms a cluster of neighbouring base stations for each mobile that it is serving. The base station will estimate number of guard channels that each base station in the cluster needs to reserve so that the mobile will not experience a dropped call. This estimate is based on the probability that the mobile under consideration will move into the coverage area of a particular base station. If the probability is low, the base station will reserve fewer channels and vice-versa. The size and shape of the cluster is dependent upon the speed of the mobile and its average call length.

Handoff is often more frequent in a network with small cells and users with a high mobility and speed. In [68] a two-layer hierarchical cellular system is proposed, where a cluster of microcells (smaller cells) are covered by one macrocell (large cell) as shown in Figure 2.2. To reduce the number of handoffs, a user that travels at high

speed (e.g. in a vehicle) is assigned a channel in the macrocell and one with a low speed is assigned to a channel in the microcell. This requires the base stations to estimate the speed of the user.

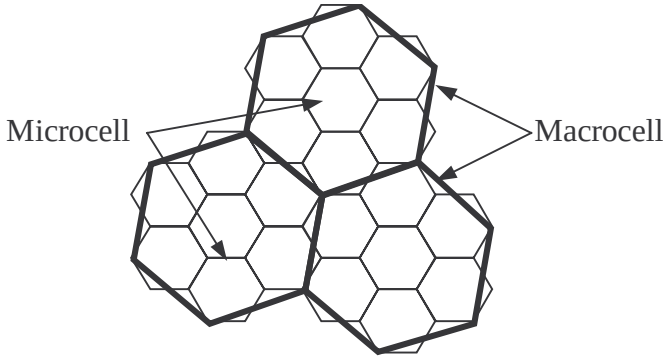

Figure 2.2: Microcell and macrocell in a two-layer hierarchical cellular system.

Handoff is a Quality of Service (QoS) issue and apart from handoff, certain other call types (e.g. emergency calls, priority users) may also be given priority over others call types. The problem in meeting a QoS requirement is to allocate or reserve the number of channels (i.e. how *many* channels to reserve). This may be compared to channel allocation where the task is to determine *which* channel to select. As QoS is itself a large area of research, it is not considered further in this thesis.

2.2 Channel Allocation Matrix

Channel Allocation algorithms can be divided into the following categories, namely Fixed Channel Allocation (FCA), Dynamic Channel Allocation (DCA) and Hybrid Channel Allocation (HCA) [18]. In FCA, channels are allocated to a base station for its exclusive use and the allocation is usually performed prior to network operation. Since the channels are static in FCA, it is difficult to adapt to changes in interference and traffic conditions. In DCA, any channel in the system can be assigned to any base station when it is needed (i.e. during network operation) contingent upon a set of conditions (e.g. Compatibility Matrix) being satisfied. Unlike FCA, the base stations employing DCA do not own any particular channels and a channel is released when a call is completed. Since the channels are assigned during network operation, DCA is able to adapt to interference and traffic changes. In a circuit switched network, DCA performs better than FCA under light to moderate traffic but it performs worse than FCA under conditions of heavy traffic [18]. HCA is a combination of FCA and DCA whereby a set of channels are statically assigned to a base station as in FCA while

another set are placed in a central pool and are assigned in a DCA manner. In this way, HCA is able to inherit the advantages of both DCA and FCA.

The classification into FCA and DCA does not reveal much about the underlying algorithms. To allocate a channel to a base station, information regarding the system and the layout of the network is required. Information can be a priori and/or current information gained during network operation. The more information the scheme obtains combined with better use of the available information, will yield improved channel assignment decisions. The channel allocation algorithms can be reclassified in a Channel Allocation Matrix, which is based on the strategy used to obtain the required information used during a channel assignment. The Channel Allocation Matrix is introduced in [87] and is shown in Figure 2.3 (with citations to the methods that fall into each category).

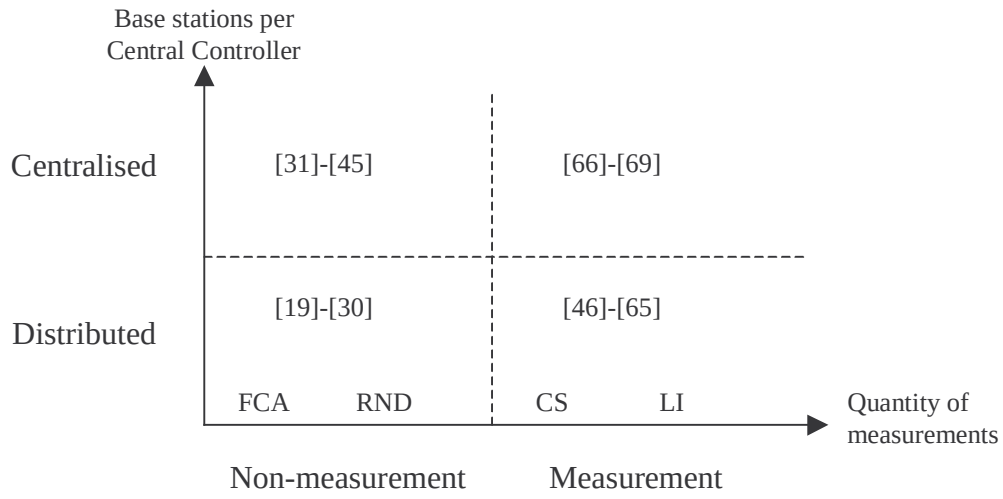

Figure 2.3: Channel Allocation Matrix

The vertical axis is a measure of the degree of centralisation required by the algorithm. The degree of centralization is quantified by the number of base stations required to communicate with a central controller in order to obtain the required information to make a single channel allocation decision. In a fully centralized system, all the base stations in the network needs to communicate with a central controller and thus each base station will have global knowledge of the entire network. It also requires signalling between base stations and central controllers and this will introduce delay in assigning a channel. The complexity of a fully centralized system increases non-linearly with the number of base stations and hence it may not

be practical in a network with a large number of base stations, since the complexity may introduce long delays causing system instability. For example, the channel usage of a neighbouring base station may change by the time this information reaches a base station that is assigning a channel. In a voice service this delay may increase the number of dropped calls leading to user dissatisfaction while in a data service delay may reduce data throughput if frequent channel changes occur. On the other hand, in a fully distributed system the base station is able to make a channel allocation decision independently (i.e. without communication with a central controller or other base station). The base station can obtain knowledge of the interference environment using measurement or may rely upon a priori information of the network configuration. A fully distributed system may not have global or current knowledge of the network, but it permits fast and simple allocations to be made. However, without global knowledge, the channel allocated may not give an optimum performance.

The horizontal axis represents the quantity of measurements performed by a base station and/or the subscriber's equipment in order to make a channel allocation decision. The measurements can be the interference power of a channel or an estimated SNR (or SIR) and in either case the aim is to evaluate the interference environment prior to allocating a channel. Measurement may introduce delay decreasing the overall throughput of a network. However, if it is performed too quickly, it may not give an accurate picture of the interference environment.

The Channel Allocation Matrix is divided into four quadrants specifically: Distributed Non-measurement, Centralized Non-measurement, Distributed Measurement and Centralized Measurement.

2.2.1 Distributed Non-Measurement

In the Distributed Non-measurement quadrant the base station does not have access to the current interference environment information. Therefore, a priori knowledge of the system and network are required to assigned channels to calls. The information required may be the location of the base stations, an estimate of the offered traffic, the propagation model of the coverage area and the number of available channels. FCA belongs to this quadrant since it relies heavily on a priori information and does not require up to date interference information during system operation. However, the

channels do not change during system operation and hence the initial channel assignment is important. Another method falling into this category is the random channel allocation (RND). The methods surveyed in this section have primarily been designed with voice services in mind.

2.2.1.1 Random Channel Allocation

Given a list of available channels, Random Channel Allocation (RND) [11], [56], [50] selects a channel randomly based on a uniform distribution. The idea is similar to Frequency Hopping, where the channels are picked at random and are usually changed at random during a call. For example, if there are five channels where one is bad, the call will only spend a short time in a bad channel and hence averaging out the interference among all channels. This is effective if the number of available channels is large and is very simple to implement and does not require a lot of prior planning in the network. Averaging interference has a poorer performance compared to an algorithm that avoids interference [12]. RND is usually used as the lower bound performance for channel allocation methods. This is because the RND algorithm does not use any current or prior information to allocate a channel.

2.2.1.2 Fixed Channel Allocation

Consider a cellular network with uniform traffic (i.e. each cell is the same size and has the same amount of offered traffic) and the layout shown in Figure 2.1. If the required SIR is 17 dB and the cell radius is fixed at R_b, for a given propagation model the reuse distance R_u and cluster size N_c can be found. In FCA, the set of channels **C** is usually divided into channel groups [1]. If $N_c = 7$, each cell in the cluster will take one of the channel groups, i.e., the channels are divided into 7 groups $\mathbf{C} = \{\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3, \mathbf{C}_4, \mathbf{C}_5, \mathbf{C}_6, \mathbf{C}_7\}$, where $\mathbf{C}_j$ is a distinct set of channels. Using this a priori information, the channels can be allocated with the reuse pattern shown in Figure 2.4.

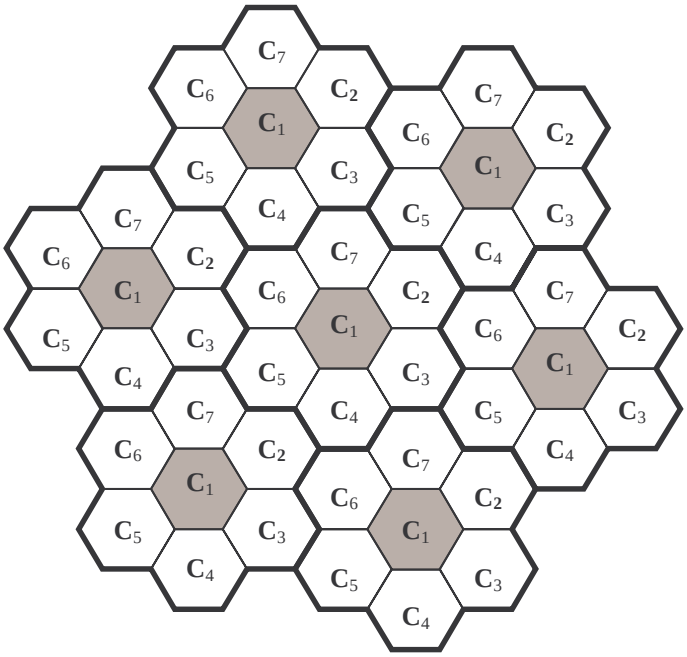

Figure 2.4: Channel Allocation in a network with uniform traffic distribution.

It is common for a network to have non-uniform traffic. For example, a cluster of cells may cover a busy shopping area (heavy traffic) and a neighbouring cluster of cells may cover a residential area (low traffic). If the reuse pattern shown in Figure 2.4 is applied to this situation, the channels may not be utilized efficiently. For example, the cells with heavy traffic may not have enough channels to ensure a low blocking probability while the cells with low traffic may not use all the channels assigned to them.

For a fixed number of channels, a smaller reuse ratio gives higher capacity. This is because a smaller reuse ratio leads to a smaller cluster size and therefore the same number of channels is distributed to a smaller number of cells. Reuse partitioning can be used to reduce the reuse ratio. In reuse partitioning, a cell is divided into concentric sub-cells [18] as shown in Figure 2.5 and each sub-cell is assigned a different group of channels.

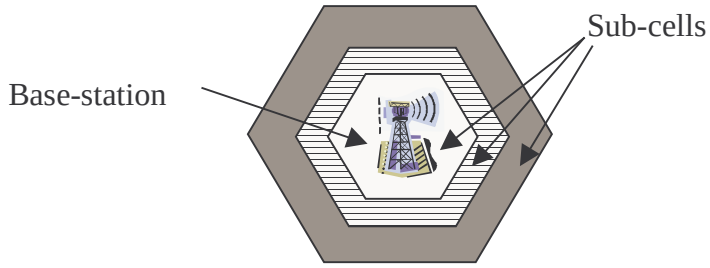

Figure 2.5: Sub-cells in reuse partitioning.

A mobile station at a sub-cell closer to the base station will require less transmit signal power than will a mobile station at the outer sub-cell to achieve the same SIR. Therefore, the inner sub-cell channels will have a smaller reuse ratio compared with those at the outer sub-cell and consequently the overall capacity is increased. Reuse partitioning has a more complex channel reuse pattern than do conventional techniques. The reuse pattern may also differ if the cell radii are not constant. Hence, the uniform reuse pattern in Figure 2.4 may not be applicable in these cases.

In a cellular network with non-uniform traffic, the compatibility matrix is used as part of the a priori information. The frequency separation between two base stations depends on the level of interference that would give an acceptable SIR. If two base stations are a sufficient distance apart such that the co-channel interference is acceptable, the frequency separation between these two base stations is zero. By estimating the frequency separation between every base stations the compatibility matrix is formed.

The optimum solution (i.e. one that uses the least number of channels) to a channel allocation problem is difficult to obtain particularly for a large network. Although finding the optimum solution is possible (e.g. using brute force), it may not be practical. It is desirable to have a solution that is as close as possible to the optimum solution and there are a large number of publications devoted to achieving this aim, some of which are highlighted later in this section. The theoretical lower bound solution C_L ($C_{MIN} \geq C_L$) is proportional the product of the highest traffic demand d_{j^*} and the channel separation χ_{j^*,j^*} at base station j^* thus [22]:

$$C_L = \chi_{j^*,j^*}\left(d_{j^*} - 1\right) + 1 \qquad (2.5)$$

Where it is assumed that $\chi_{j,j} > \chi_{j,k}$ for all k and usually $\chi_{j,j} = \chi_{k,k}$ for $j \neq k$. For a solution that uses C channels, it is desirable that the channel numbers are as close together as possible, which is known as channel packing. For example, if an assignment requires three frequency bands, an assignment that uses c_1, c_2, c_3 is preferred to one that uses c_1, c_5 and c_7, where c_i is the i^{th} continuous frequency band in the spectrum. Channel packing leaves the remaining channels with more free channels (i.e. channels that does not violate the compatibility matrix) for future assignment. This in effect squeezes more channels into a smaller area. One way of achieving high channel packing is to minimize the average distances between the co-channel base stations.

For $C > C_{MIN}$, the difficulty of a channel allocation problem is proportional to the consumption of free channels by each base station. The degree of difficulty $\boldsymbol{D}_{\mathbf{B}}(j)$ of assigning channels to a base station j in **B** is given as [21]:

$$\boldsymbol{D}_{\mathbf{B}}(j) = \left(d_j \sum_{k=1}^{B} \chi_{j,k} \right) - \chi_{j,j} \qquad (2.6)$$

The problem can be broken down into smaller sub-problems and each sub-problem solved one at a time starting with the most difficult sub-problem. In [19], the channel assignment problem is transformed to a graph colouring problem (another NP-complete problem [8]). In the graph colouring problem, a graph consists of nodes and edges (where each edge joins two nodes) and the task is to colour the nodes such that no two nodes sharing the same edge has the same colour. The colours represent the channels and the nodes and edges represent the requirements. The graph is divided into sub-graphs and they are ordered according to their level of assignment difficulty. A frequency exhaustive method is used to assign channels to each sub-graph in order of the sub-graph's level of assignment difficulty. The frequency exhaustive method starts by assigning the lowest numbered channel to as many requirements as possible and this is followed by the next lowest numbered channel. The frequency exhaustive method achieves high channel packing since it gives priority to lower numbered channels. The level of assignment difficulty can be proportional to the number of edges connected to a node (node-degree ordering) or the size of the complete sub-

graph (node-colour ordering). A complete sub-graph of size G has G nodes where each node is connected to all the other G-1 nodes (i.e. each node has G-1 edges).

The ordering can also be done in a heuristic manner as in [20]. In this method, the base stations are ranked according to their level of "difficulty" (a value) and the ranking is done in an iterative manner. The number of channels C is fixed and each base station is assigned a "difficulty" value, which is set to be the same at the beginning of the search. For each iteration, channels are assigned to the base station in the order of the list. After the channel assignment attempt for the final base station, the "difficulty" value for each base station that has call blocks (i.e. base stations without a valid channel assignment) is increased by a random value drawn from a uniform distribution. The list is re-sorted according to the level of difficulty and the iteration process is repeated until all the base stations have valid channels. This method may not give the optimum solution (if $C \neq C_{MIN}$) and the number of iterations may be large for difficult problems or may not stop if the number of channel C does not lead to a solution.

The ordering technique settles upon the first solution even if it is a poor solution (i.e. C is much greater than C_{MIN}). A heuristic method that considers a set of ordered lists is considered in [21], where each list is a random order of base stations. For each list, the algorithm assigns channels for the first base station in the list until all the traffic demand is satisfied. The assigned channels may hinder other base stations (lower in the list) from using them since it may violate the compatibility matrix. The algorithm will consider the next base station in the list until all base stations and their traffic demands are satisfied. This algorithm successfully finds C_L for some problems, but for difficult problems, the algorithm may require a large number of ordered lists.

There is no measure of the cost or effectiveness of a solution in the ordering techniques. Another approach to achieve a sub-optimal solution is to search through part of the total number of possible channel assignments. The search space (possible channel assignments) is large and therefore the number of searches that can be performed is usually only a small fraction of the total search space. Hence, an efficient search method is essential to find a sub-optimal solution. A typical search method involves defining a search space **S** (the set of all possible solutions), where each point (possible solution) in the search space is usually represented by a matrix

$\mathbf{s} \in \mathbf{S}$, a method to move from one solution **s** to another **s**' and a cost function $\boldsymbol{C}(\mathbf{s})$ to give a numerical evaluation of the solution **s**.

An adaptive local-search method is used in [22], where an ordered list of calls is used. An ordered list of calls is given by say, $\mathbf{a} = [a_{4,1}\ a_{5,2}\ a_{2,3}, \ldots]$, where each element $a_{j,k}$ represents a call (i.e. the k^{th} call from the base station j, each call occupies a radio transceivers). This approach is similar to the list used in the ordering techniques in [20] and [21] described previously, but instead of having an ordered list of base stations, it has an ordered list of traffic demands (i.e., calls). For an example, consider a set of base stations $\mathbf{B} = \{b_1, b_2, b_3\}$ with $\mathbf{D} = [3\ 2\ 2]$. This means b_1 has three calls $a_{1,1}$, $a_{1,2}$ and $a_{1,3}$, b_2 has two calls $a_{2,1}$ and $a_{2,2}$; and b_3 has two calls $a_{3,1}$ and $a_{3,2}$. If $\mathbf{a} = [a_{1,3}, a_{2,1}, a_{2,2}, a_{1,1}, a_{3,1}, a_{3,2}, a_{1,2}]$, a channel is assigned to b_1, followed by two channels to b_2, followed by one channel for b_1, then a two channels for b_3 and finally a channel for b_1. Note that if each base station has one call demand then the nature of the lists in these three techniques ([20], [21] and [22]) are the same. The cost $\boldsymbol{C}(\mathbf{a})$ is the number of channels assigned to **a** (ordered list of calls) using the frequency exhaustive method. The goal is to find a call list that uses as few channels as possible to meet the traffic demand. In a local search for each call list **a**, there is set of possible call lists called the neighbourhood $\mathbf{N}(\mathbf{a}) \in \mathbf{A}$ (**A** is the set of all possible call lists), where each possible call list $\mathbf{a}' \in \mathbf{N}(\mathbf{a})$ is a slight modification of **a**. For example in this case, **a**' is found by swapping the orders of two calls in **a**. The first $\mathbf{a}' \in \mathbf{N}(\mathbf{a})$ that has a lower cost ($\boldsymbol{C}(\mathbf{a}') < \boldsymbol{C}(\mathbf{a})$) replaces **a** (i.e. $\mathbf{a}=\mathbf{a}'$) as the next point of consideration. This process is iterated until an estimated optimum (i.e. $\boldsymbol{C}(\mathbf{a}) = C_L$) solution is found or after a predefined number of iterations has taken place. The search space of a large network can be reduced by representing a non-uniform traffic network as the superposition of two networks [22]. The first network has a uniform traffic distribution, where each base station has the minimum traffic demand of the original network. The second network has a non-uniform traffic distribution where the traffic in each base station is the difference between the traffic of the original network and the traffic of the first network. The first network can be solved using the method described in Figure 2.4. The second network has a smaller search space and therefore the chance of finding an optimum solution is much greater.

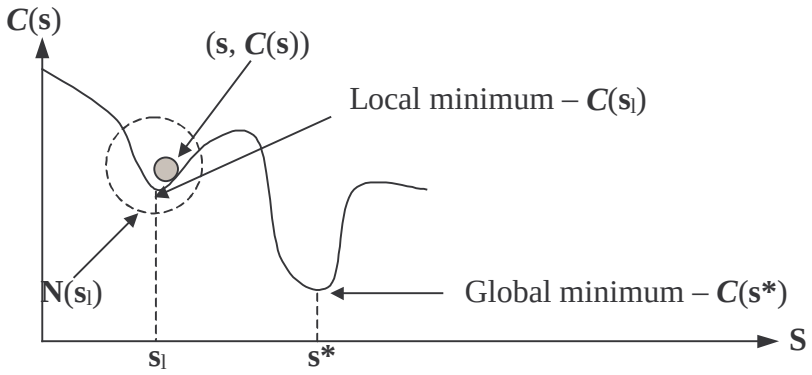

*Figure 2.6: Local and global minima. A point (***s**, **C**(**s**)*) is represented as a rolling ball.*

In a search method, it is easy to get trapped in a poor local minimum, e.g. at $\mathbf{s}_l$ (in Figure 2.6), if $\boldsymbol{C}(\mathbf{s}') > \boldsymbol{C}(\mathbf{s}_l)$ for all $\mathbf{s}' \in \mathbf{N}(\mathbf{s}_l)$ and $\boldsymbol{C}(\mathbf{s}_l) > \boldsymbol{C}(\mathbf{s}^*)$ where $\mathbf{s}^*$ is the optimum solution (or global minimum) with the lowest cost (i.e. $\boldsymbol{C}(\mathbf{s}) \geq \boldsymbol{C}(\mathbf{s}^*)$ for all $\mathbf{s} \in \mathbf{S}$). This can be avoided if the algorithm accepts **s**' as the next transition even if $\boldsymbol{C}(\mathbf{s}') > \boldsymbol{C}(\mathbf{s})$, which is equivalent to giving an upward push to the ball in Figure 2.6 so that it moves out of the valley (local minimum) to explore other regions. In [22], this is done by accepting **s**' if $\boldsymbol{C}(\mathbf{s}') \geq \boldsymbol{C}(\mathbf{s})$ after a predetermined number of iterations. Examples of algorithms that are capable of escaping from a poor local minimum are Simulated Annealing, the Tabu Search, Neural Networks (modified version) and the Genetic Algorithm.

Simulated Annealing escapes from a poor local minimum by accepting $\mathbf{s}' \in \mathbf{N}(\mathbf{s})$ for $\boldsymbol{C}(\mathbf{s}') > \boldsymbol{C}(\mathbf{s})$ with a probability $\boldsymbol{P}_{sa}(\mathbf{s}', \mathbf{s})$, which is known as the Metropolis test and is given as:

$$\boldsymbol{P}_{sa}(\mathbf{s}', \mathbf{s}) = \boldsymbol{min}\left(1, \quad \boldsymbol{exp}\left(\frac{-\left(\boldsymbol{C}(\mathbf{s}') - \boldsymbol{C}(\mathbf{s})\right)}{T}\right)\right) \tag{2.7}$$

where T is the temperature. At a high temperature the algorithm accepts more solutions with high cost and in effect it searches a large search space **S**, while at a low temperature the algorithm prefers solutions with low cost. At the beginning of the iteration, the temperature is set to a high value and decreases with the number of iterations [13]. Simulated Annealing is used in [23] to find a solution for a FCA. The solution **s** is a $B \times C$ matrix where B is the total number of base stations and C is the total number of channels and an element $s_{j,k}$ in **s** (j=1 to B and k=1 to C) is given in

equation (2.3). The cost $\boldsymbol{C}(\mathbf{s})$ encourages assignments that conform to the compatibility matrix, that meet the traffic requirement and have a high channel packing. The neighbouring solution $\mathbf{s}' \in \mathbf{N}(\mathbf{s})$ is generated by replacing a used channel with a currently unused channel in base station j.

The Tabu Search maintains a list called the Tabu List $\mathbf{T}_{LIST}$ that contains the attributes of a solution that the algorithm has considered in past iterations. In moving from **s** to $\mathbf{s}' \in \mathbf{N}(\mathbf{s})$, the algorithm will not select **s**' if it is in the Tabu List (i.e. if $\mathbf{s}' \in \mathbf{T}_{LIST}$) [13]. If the size of the Tabu List $\|\mathbf{T}_{LIST}\|$ is large, the search will progress move quickly away from $\mathbf{N}(\mathbf{s})$ thereby exploring a larger search space and if $\| \mathbf{T}_{LIST}\|$ is small, it will intensify its search around $\mathbf{N}(\mathbf{s})$. The Tabu Search is used in [24], where the solution **s** is the same as that described in (2.3) and the neighbourhood solution $\mathbf{s}' \in \mathbf{N}(\mathbf{s})$ is generated in the same way as is done in the Simulated Annealing approach described previously. However, [24] recognises and shows that the compatibility matrix is not unique for a given SIR threshold. Rather than using a cost function, it uses a gain or a fitness function $\boldsymbol{F}(\mathbf{s})$ where the aim is to maximize $\boldsymbol{F}(\mathbf{s})$. For a given assignment **s**, $\boldsymbol{F}(\mathbf{s})$ is the number of channels assigned to a base station that have a SIR above a predefined threshold, where the SIR takes into account the cumulative interference from all co-channel base stations.

A popular method employed to solve the FCA problem is the Neural Network. A neuron in a Neural Network is shown in Figure 2.7, where $\mathbf{x} = [x_1, x_2, \ldots x_N]$ is the input and q_i is a bias current for neuron i [14]. The internal state u_i of neuron i is given by:

$$u_i = \sum_{j=1}^{N} w_j x_j + q_i \qquad (2.8)$$

where w_j is the weight for input x_j. The activation function $\boldsymbol{A}(u_i)$ is a function of the internal state u_i and is usually $0 \leq \boldsymbol{A}(u_i) \leq 1$ (e.g. a unit step function) and the output of neuron i is $y_i = \boldsymbol{A}(u_i)$.

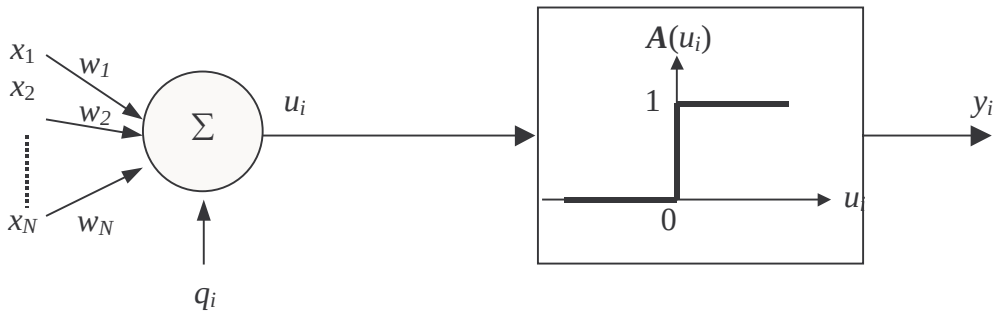

Figure 2.7: Representation of neuron i.

In a Hopfield Neural Network (HNN) [15], the output of all neurons are connected to the input of every other neuron except itself. This is shown in Figure 2.8, where $\mathbf{x}=[x_1, x_2, \ldots, x_N]$ is the output from the neurons, x_i is the output of neuron i (i.e. n_i) and $w_{j,k}$ is the weight from n_j to n_k (which weights x_j prior to entering neuron n_k) The weights in a HNN are symmetrical ($w_{j,k} = w_{k,j}$) and $w_{j,j} = 0$. The state of each neuron is its current output $x_j \in \{0,1\}$ and the energy $\boldsymbol{E}(\mathbf{x})$ in the HNN decreases for each update of $\mathbf{x}$ until it converges to a local minimum, which is dependent upon the weights. The energy $\boldsymbol{E}(\mathbf{x})$ in HNN is given as:

$$\boldsymbol{E}(\mathbf{x}) = -\frac{1}{2}\sum_{j=1}^{N}\sum_{k=1}^{N} w_{j,k}x_j x_k - \sum_{j=1}^{N} q_i x_j \qquad (2.9)$$

Hence, by constructing an appropriate $\boldsymbol{E}(\mathbf{x})$ corresponding to the cost of an optimisation problem and expressing the state of the neuron $\mathbf{x}$ as a possible solution, a HNN with the appropriate weights can find a solution to an optimisation problem.

In [25], a HNN is used to find a sub-optimal solution for FCA. Here, a neuron is an element $s_{j,k}$ in $\mathbf{s}$ (j=1 to B and k=1 to C) as in (2.3) giving $B \times C$ neurons. The energy $\boldsymbol{E}(\mathbf{s})$ is equivalent to the cost function $\boldsymbol{C}(\mathbf{s})$ and $w_{j,k,j',k'}$ is the weight from neuron $s_{j,k}$ to neuron $s_{j',k'}$ and it is a function of the interference between $s_{j,k}$ and $s_{j',k'}$. The weight $w_{j,k,j',k'}$ decreases if the channel separation $|k - k'| < \chi_{j,j'} \in \mathbf{X}$ (the channel separation between base station j and j' in the compatibility matrix). The bias current q_j is proportional to the traffic demand of base station j yielding an energy $\boldsymbol{E}(\mathbf{s})$ in the form of (2.9) that is a function of the overall interference of the network and the traffic supported for a given assignment $\mathbf{s}$. A HNN minimising $\boldsymbol{E}(\mathbf{s})$ is equivalent to finding a solution $\mathbf{s}$ with a low cost $\boldsymbol{C}(\mathbf{s})$.

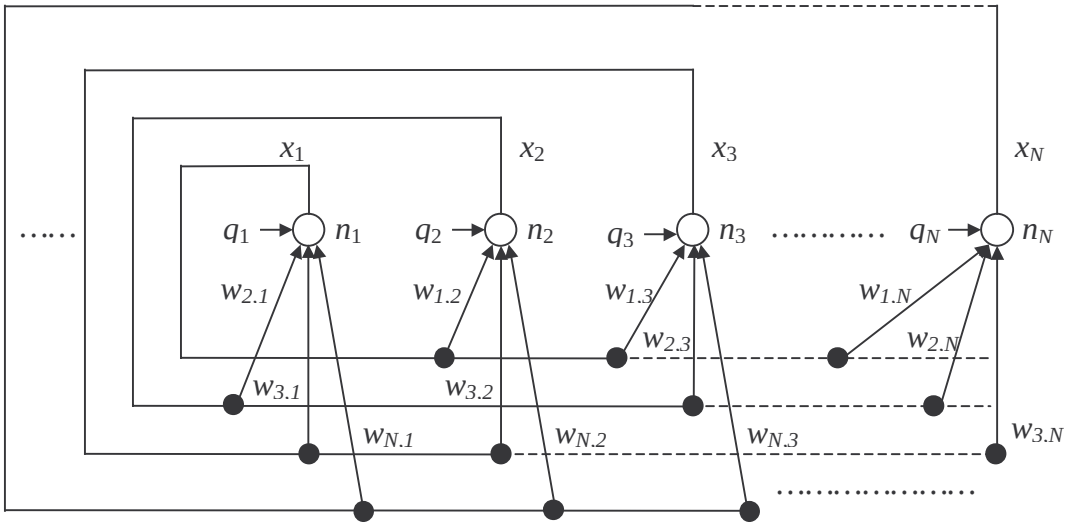

Figure 2.8: Hopfield Neural Network (a black solid dot represents a connection).

The HNN technique is not able to escape from a local minimum and realising this an improved HNN is used in [26] where a temporary increase in $\boldsymbol{E}(\mathbf{s})$ is permitted so that it can escape from local minima. The increase in $\boldsymbol{E}(\mathbf{s})$ is achieved by introducing noise via a random function into the internal function u_i in (2.8). This will change the value of $\boldsymbol{A}(u_i)$, thereby causing a random increase in $\boldsymbol{E}(\mathbf{s})$. The random increase in $\boldsymbol{E}(\mathbf{s})$ becomes less as time proceeds in a similar way to the process employed in Simulated Annealing.

The Genetic Algorithm [16] is inspired by the natural process of the adaptation of species in an environment. Instead of a neighbourhood $\mathbf{N}(\mathbf{s})$ of possible solutions centred around $\mathbf{s}$ as in Simulated Annealing and Tabu Search, the Genetic Algorithm considers a population $\mathbf{P}(t) \in \mathbf{S}$, which is a set of possible solutions where at the t^{th} iteration, $\mathbf{P}(t)=\{\mathbf{s}_1, \mathbf{s}_2, \ldots \mathbf{s}_K\}$, where $K = |\mathbf{P}(t)|$. Hence, the Genetic Algorithm searches various points in $\mathbf{S}$ rather than those centred around $\mathbf{N}(\mathbf{s})$. Each solution $\mathbf{s} \in \mathbf{P}(t)$ is called an individual and it is evaluated using a fitness function $\boldsymbol{F}(\mathbf{s})$, which measures the gain of using solution $\mathbf{s}$ in a similar way to the cost function $\boldsymbol{C}(\mathbf{s})$. Using $\boldsymbol{F}(\mathbf{s})$ the optimum solution is the global maximum as compared to $\boldsymbol{C}(\mathbf{s})$ where the optimum solution is the global minimum. The Genetic Algorithm starts off with an initial population $\mathbf{P}(0)$ and it is followed by Selection, Crossover and Mutation processes. These processes are shown in Figure 2.9.

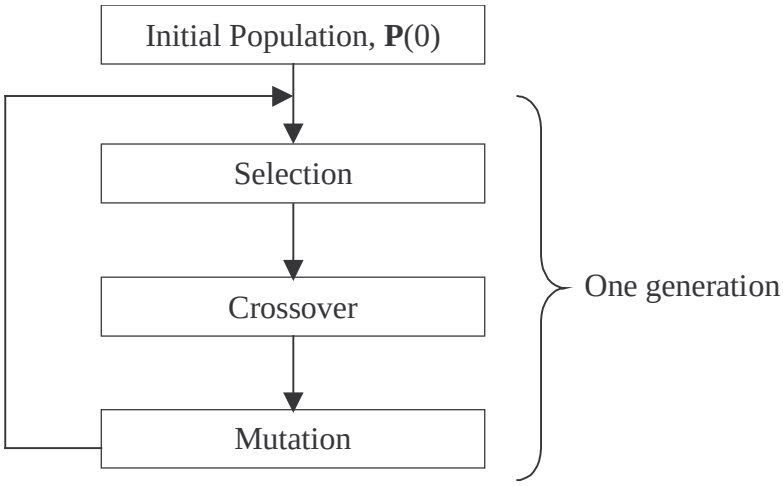

Figure 2.9: Processes in a Genetic Algorithm.

In the Selection process, individuals are selected for the Crossover process where individuals with a higher fitness $\boldsymbol{F}(\mathbf{s})$ have a higher probability of being selected. Figure 2.10 shows the Crossover process with two individuals, where a portion of Individual 1 is cut and placed into the same portion of Individual 2 to produce a new individual (the offspring). The two individuals selected have a probability of p_c of being subjected to Crossover.

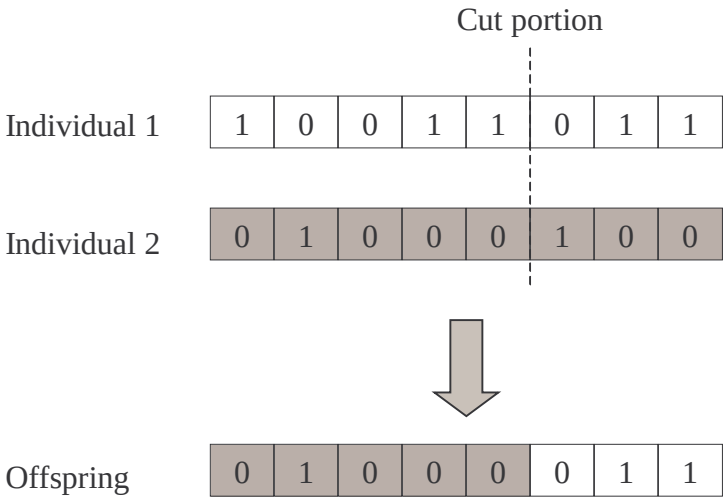

Figure 2.10: Crossover process.

In the Mutation process, each element in an individual is changed (in the case of binary representation, the element is complemented) with a small probability of p_m. These 3 processes are performed on $\mathbf{P}(t)$ to produce a new generation of individuals $\mathbf{P}(t+1)$. Although Selection favours fitter individuals, individuals with poor fitness also have a chance of being selected so that the algorithm will not converge to a poor

local maximum. The average fitness of the population will increase and converge after a number of generations (i.e., iterations) as the individuals in the population become more similar to each other. Mutation increases diversity in the population and thus allows the algorithm to search more distinct points.

The Genetic Algorithm is applied to the FCA problem for both uniform and non-uniform traffic in [27]. The size of string **s** is equal to the sum of the traffic demand for all base stations, and each element $s_{j,k}$ in **s** is a channel number that is assigned to base station j for the k^{th} call. This is illustrated in Figure 2.11. The traffic demand for each base station is included in the structure of the string and therefore the fitness function consists only of the co-channel and adjacent channel constraints.

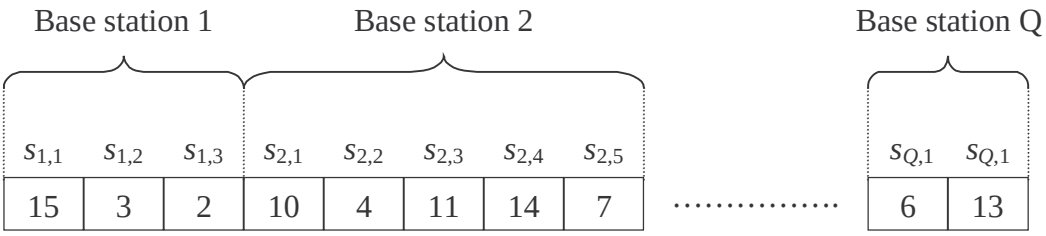

Figure 2.11: String representation **s** *of an individual.*

Roulette wheel selection is used in [27], where each section in the wheel corresponds to an individual and the size of each section is proportional to the individual's fitness. This is illustrated in Figure 2.12. Selection is random but biased towards individuals with a higher fitness. The probability of being selected is shown in Figure 2.12.

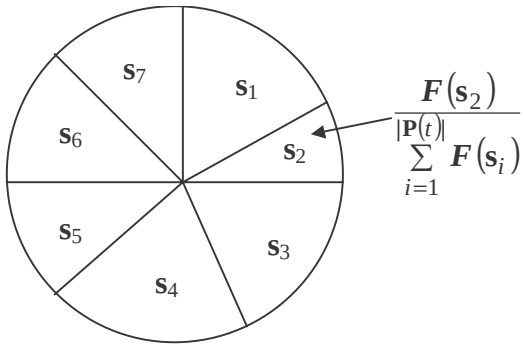

Figure 2.12: Roulette wheel selection – section size proportional to fitness (shown here for $\mathbf{s}_2$*).*

The Genetic Algorithm is used in [28] to search for an ordered call list **a** instead of an assignment. The call list **a** is the same as that used in the local search method [22] described previously. As in [22], a frequency exhaustive method is used to assign channels to **a**. However, in [28] the number of channels C is given and the fitness is a function of the number of blocked calls (i.e. the number of calls without valid assignments) and the degree of channel packing. To find the C_{MIN}, this method first uses an estimated lower bound for C and if the algorithm fails to find a solution (i.e. one with no blocked calls) within a predefined number of iterations it increases C and restarts the algorithm. This method also uses roulette wheel selection depicted in Figure 2.12.

All the methods described are able to find a solution satisfying the compatibility matrix if the given number of channels C is larger than C_{MIN}. However, for a given compatibility matrix, if $C < C_{MIN}$, no solution exists to the channel allocation problem without giving rise to call blocks. Rather than having a hard constraint (i.e. conforming fully to the compatibility matrix), a soft constraint that allows a channel assignment to violate the compatibility matrix can be applied to guarantee a solution at the expense of higher interference. In applying a soft constraint, a violation with a higher channel separation is preferred to one with a smaller channel separation. For example if two base stations require a channel separation of 5 and no channel allocation can satisfy this, it is preferable to select an assignment with a channel separation of 4 compared with one with a channel separation of 2. However, if the channel separation condition is satisfied then in order to achieve higher channel packing, it is preferred to have an assignment with a channel separation of 5 than one with a channel separation of 7. If $C > C_{MIN}$ applying a soft constraint may produce a solution with high interference if the solution is trapped in a bad local minimum. The energy function used in the improved HNN in [26] implements a soft constraint on the compatibility matrix and a hard constraint on the traffic demand. A soft constraint is also implemented into the cost function of the Simulated Annealing technique in [23]. A soft constraint is also used in [29], where the network is divided in groups of cells and in each group, the cells are further grouped into co-channel sub-groups. For each co-channel sub-group in a cluster, channels are assigned to the cell with the highest traffic demand. The remainder of the cells in the co-channel sub-group can thus use these channels. For cells that do not belong to any co-channel sub-group, and if no

channel that conforms to the interference constraint is found, a soft constraint is applied where up to 13% of interference power is allowed to exceed the required minimum.

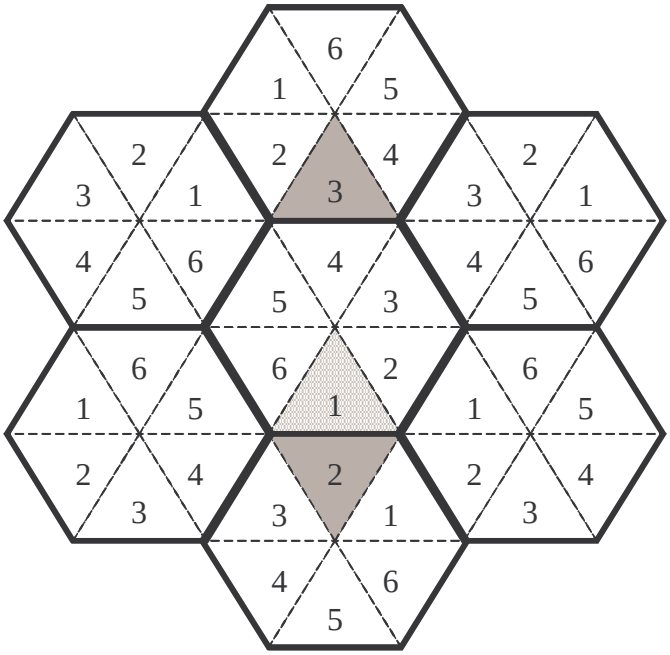

Figure 2.13: Time slots arrangement in SRA (numbers indicate time slot number).

Most of the FCAs considered are used for mobile cellular communication. In [30] a Channel Allocation method called Staggered Resource Allocation (SRA) is introduced for use in a Broadband Fixed Wireless Access Network where the spectrum is reused in every cell. For a frequency reuse of one, the cells are divided into sectors where each base station uses a sectored antenna. The SRA method uses TDMA and ensures that base stations with the front lobe of the antenna pointing towards each other do not transmit at the same time. Consider the cell layout shown in Figure 2.13, where the system has six time slots in a TDMA frame and six sectors in each cell.

The number in Figure 2.13 indicates the time slot used by the sector. The sector using time slot 1 (bottom sector) in the middle cell (lightly shaded) will face the highest interference from the two shaded sectors shown. The subscriber of this sector (bottom sector) in the middle cell will be interfered by the base station in the sector using time slot 3 (bottom sector) in the upper cell while its base station will be interfered by the base station in the sector using time slot 2 (upper sector) of the bottom cell. SRA

assigns different time slots to these main interferers so that they will not transmit together. However, it is possible that a sector will need more than one time slot. For this case, consider the middle cell; now if the bottom sector requires two time slots, it will transmit its packets in its designated time slot (i.e. time slot 1) and wait until time slot 4 to transmit the rest of its packets. Time slot 4 is selected because at this time slot, the upper sector starts to transmit its first packet and since the antennas of these two sectors do not face each other, the concurrent transmission of these two sectors will cause the least interference (within the cell). For the same sector (bottom sector) if a third time slot is required, it will transmit at time slot 5 for the same reason. Each sector will pick the time slot in an order corresponding to the amount of interference that will be incurred. The number of concurrent transmission within a cell is limited (e.g. to 3 base stations) depending upon the SNR required for acceptable reception.

2.2.2 Centralised Non-measurement

The systems in this quadrant depend less upon a priori knowledge but more upon current knowledge and hence the channel is assigned dynamically during network operation. The base stations are connected to a central controller where they obtain information relating to current interference and traffic conditions. The system assumes that the network is operating in a licensed band and that only same network co-channel and adjacent channel interference exists. Therefore, measurements are not performed since interference and traffic condition can be extracted via information concerning the channel usage of the base stations in the network.

Since the interference and traffic conditions change during network operation, the base stations using this information can assign the channel more efficiently thereby adapting to traffic conditions and reducing call blocking. For example, consider the three cell network in Figure 2.1 with base stations $\mathbf{B} = \{b_1, b_2, b_3\}$, demand $\mathbf{D} = [2\ 1\ 1]$ and channel set $\mathbf{C} = \{c_1, c_2, c_3\}$, where $C = |\mathbf{C}|$. Now consider the situation if the channel separation between neighbouring cells is changed from zero to one with other separations remaining the same as those in the example presented at the beginning of the Chapter 2. This scenario does not have a solution (i.e. $C<C_{MIN}$) and if FCA is used one of the base stations will experience call blocking with the previously specified **D**. For example, if c_1 and c_3 are assigned to b_1 and c_2 assigned to b_3, b_2 will

not have a free channel and will experience a call block whenever a new call arrives. This is illustrated in Figure 2.14.

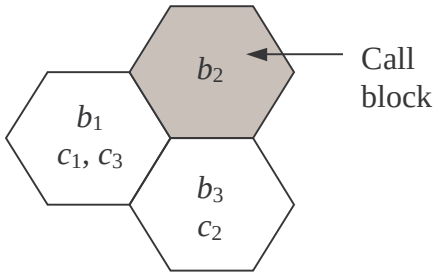

Figure 2.14: Call block in b_2 using FCA for (C<CMIN).

However, during network operation, not all the channels are used simultaneously, that is the traffic demand vector $\mathbf{D}(t)$ is a function of time t. Therefore, using DCA the assignment $\mathbf{s}(t)$ is also a function of time and changes according to $\mathbf{D}(t)$. Assume at time t_k the traffic demand is $\mathbf{D}(t_k)$=[1 0 1] and the assignment $\mathbf{s}(t_k)$ using the notation in (2.3) is (where channels are represented by columns and base stations by rows):

$$\mathbf{s}(t_k) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.10}$$

At time t_k+1, there is a new call at b_2, that is $\mathbf{D}(t_k+1)$=[1 1 1]. Now if b_2 knows the channel usage of b_1 and b_3 using centralised information, it can assign c_1 (i.e. $s_{2,1}(t_k+1) = 1$) to this new call and still conform to the compatibility matrix thereby admitting the call. This is in contrast to the FCA scenario described previously where a call in b_2 would have been blocked. Assume at time t_k+2 there is no call in b_2 (i.e. $s_{2,1}(t_k+2) = 0$) and the traffic demand is changed to $\mathbf{D}(t_k+2)$=[1 0 2], centralised information will permit b_3 to assign c_1 to this new call (i.e. $s_{3,1}(t_k+2) = 1$). In the FCA scenario this new call at b_3 will be blocked because it was designed for $\mathbf{D}$=[2 1 1] not withstanding that the total traffic in $\mathbf{D}(t_k+2)$ is actually smaller than that in $\mathbf{D}$. Clearly, FCA is at a disadvantage since it is unable to adapt to small changes in traffic. Note that although b_3 has a free channel at time t_k+2 in the DCA scenario to service this new call, the call will still be blocked if no radio transceiver is available to support this channel. Therefore, it is possible for methods in this quadrant to adapt to fluctuations in traffic demand but at a cost of installing extra radio transceivers in each base station. Since

no measurements are made, a call may still be blocked if a bad compatibility matrix (or one that uses soft constraints) or interference from another system (e.g. if the system operates in an unlicensed band) gives rise to a poor SNR, even if a free channel and radio transceiver is available.

FCA is designed for high traffic demand because the solution **s** found is close to the optimum value with the assumption that all channels are occupied. Employing DCA in this quadrant will give a solution **s**(t) similar to that of an FCA technique using call ordering **a** (e.g. [19], [20],[21], [22] and [28]). However for DCA the call order **a** is determined by **D**(t), which may result in a poor local optimum solution. At high traffic, d_j is high for all $b_j \in \mathbf{B}$ and in (2.6) the degree of assignment difficulty $\boldsymbol{D}_{\mathbf{B}}(j)$ for the base stations becomes harder as less free channels are available causing the DCA to select poor channels leading to a poor solution. Therefore, at low to medium traffic, DCA may perform better than FCA, but at high traffic, FCA tends to perform better than DCA.

The degree of centralisation is dependent upon the algorithm used. Full centralisation may have problems with scalability particularly if the algorithm requires frequent updates and a significant processing time. Consequently, a fully centralised method may be limited to small networks. There are many algorithms where the degree of centralisation includes only the base stations that fall within the reuse distance of base station j, (i.e. b_j) that is base stations that would interfere with b_j if any of them uses the same channel as b_j. The set of base stations that fall within the reuse distance of b_j (i.e. R_U) are called the interfering neighbours $\mathbf{N}_{\mathrm{I}}(b_j)$ of b_j and are shown in Figure 2.15. Hence, a channel is free in b_j if it is not used by base stations in $\mathbf{N}_{\mathrm{I}}(b_j)$. Now define $\mathbf{N}_{\mathrm{Co}}(b_j)$ as the set of base stations that are the nearest co-channel neighbours of b_j – that is a channel can be used simultaneously in b_j and $\mathbf{N}_{\mathrm{Co}}(b_j)$.

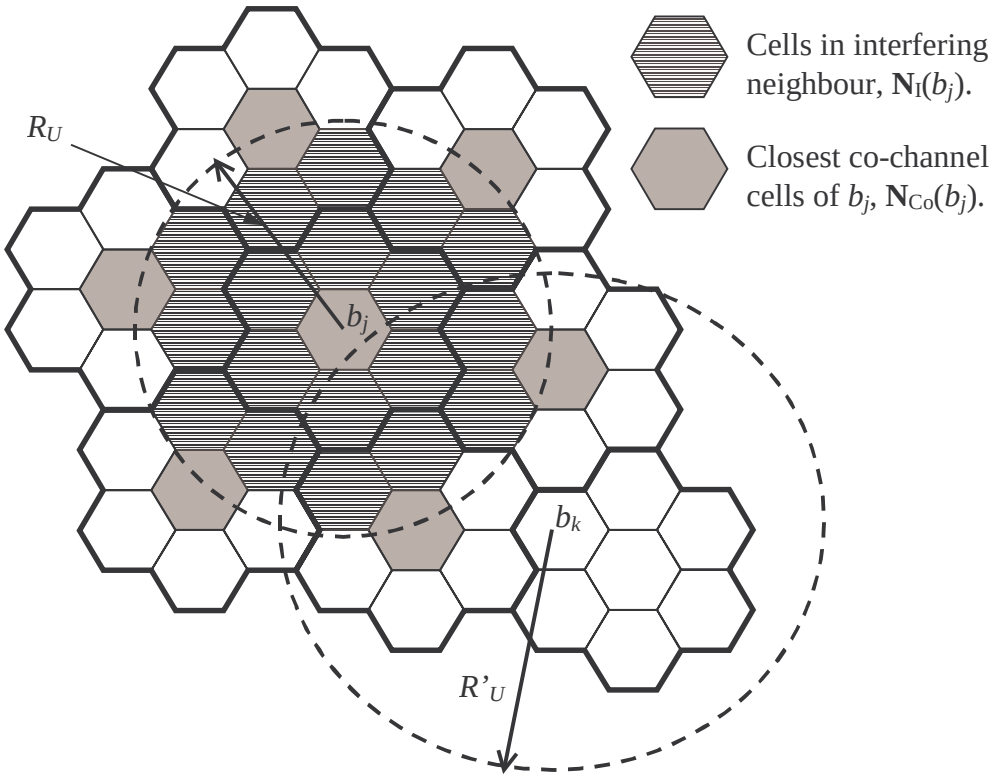

Figure 2.15: Interfering neighbours $\mathbf{N}_I(b_j)$ *and closest co-channel cells* $\mathbf{N}_{Co}(b_j)$

Assume that each group of 7 distinct cells of the network shown in Figure 2.15 are connected to a base station controller (BSC) (each group of cells is outlined in bold). Although b_j needs only to communicate with base stations in $\mathbf{N}_I(b_j)$, the BSC connected to b_j (BSC j) needs to communicate with six other BSCs since the base stations in $\mathbf{N}_I(b_j)$ are connected to them. If base station k, namely b_k has a reuse distance R'_u (the reuse distance may not be the same for all cells) that includes base stations connected to BSC j falling within $\mathbf{N}_I(b_k)$, BSC j will also need to communicate with the BSC connected to b_k. Let $\mathbf{B}_j=\{b_m, b_{m+1}, \ldots b_{m+Nc}\}$ be the set of all base stations connected to BSC j. Hence, BSC i needs to communicate with BSC j and/or BSC k if $\{\mathbf{B}_j \cap \mathbf{N}_I(b_l)\} \notin \varnothing$ for $b_l \in \mathbf{B}_i$ and/or $\{\mathbf{B}_j \cap \mathbf{N}_I(b_n)\} \notin \varnothing$ for $b_n \in \mathbf{B}_k$, where $\varnothing$ is the null set. In order for two BSCs to communicate, they need to be either linked directly or connected to a main central controller that routes all signals between BSCs. Although, a central controller may be involved, centralisation up to the interfering neighbour is scalable since the amount of signalling is limited and less than that in a fully centralised system. The lesser amount of signalling also means shorter delays in assigning a channel but note that each assignment will not have channel usage knowledge of the entire network.

2.2.2.1 Centralisation up to the Interfering Neighbour

Centralisation can be introduced into a network with a set of base stations $\mathbf{B}=\{b_1, b_2, \ldots b_B\}$ using FCA to utilise the channels more efficiently. The so called nominal channels are a set of channels assigned to base station j using the FCA technique. If b_j uses up all its nominal channels and a new call arrives, b_j can still serve this call (if a radio transceiver is available) by borrowing a channel from one of its neighbouring base stations that has not used up all its nominal channels. This method is called Simple Channel Borrowing [31]. The channel borrowed by b_j must not interfere with other base stations (i.e., it must conform to the compatibility matrix) and base stations in $\mathbf{N}_I(b_j)$ are prohibited from using it. The borrowed channel is said to be locked and this channel is unlocked when the call is completed (or is handed off to another base station). In light traffic, Simple Channel Borrowing performs better than pure FCA but in heavy traffic FCA performs better. In heavy traffic, channel borrowing becomes more frequent leading to excessive channel locking and a consequent increase in the blocking probability. The set of nominal channels can be divided into two subsets where one set can be borrowed while the other is used only by b_j. This method is called Hybrid Channel Borrowing and it reduces the effect of excessive channel locking at high traffic demands [31]. When a channel is borrowed by b_j, instead of locking all base stations in $\mathbf{N}_I(b_j)$, for a system using a sectored antenna, Borrowing with Directional Channel Locking (BDCL) [31] locks only the base stations in $\mathbf{N}_I(b_j)$ with antennas facing directly toward b_j. This reduces the number of channel locks. A call using a borrowed channel can be reassigned to a nominal channel whenever one becomes available, thereby reducing the channel locking period.

Channel borrowing is built upon FCA and therefore requires the a priori knowledge described previously, which will be time consuming to acquire and maintain in a large network. In some systems, the base station may be mobile and hence a priori knowledge is not available [39]. As described previously, DCA avoids going through the FCA process since all the channels in DCA are made available for use. A Simple DCA method [32] is to attach an ordered list of channels $\mathbf{C}_j$ to each base station (i.e. $b_j \in \mathbf{B}$) and whenever a call arrives at b_j, the algorithm will pick a free channel that has the highest order in $\mathbf{C}_j$. Once a channel is selected, b_j will lock this channel preventing the base stations in $\mathbf{N}_I(b_j)$ from using it. For a $b_k \in \mathbf{N}_I(b_j)$ the ordered list is

arranged such that a high ordered channel in $\mathbf{C}_k$ is a low ordered channel in $\mathbf{C}_j$ and the base stations in $\mathbf{N}_I(b_j)$ and b_j should have unique lists. A call using a lower ordered channel is reassigned to a higher ordered channel whenever the higher ordered channel is freed. The ordered list can also make use of a priori network information since more information will lead to a better channel allocation. In [33] a Geometric DCA is introduced that uses a priori information in the ordered list, where the set of channels $\mathbf{C}$ is divided into distinct groups $\{\mathbf{C}_1, \mathbf{C}_2, \ldots \mathbf{C}_K\}$, where each group $\mathbf{C}_j$ is a subset of $\mathbf{C}$ and each group consists of an ordered list of channels (usually separated by $\chi_{j,j}$ so that they can be used in the same base station). Each base station say b_j within a cluster is assigned a different channel group (e.g. $\mathbf{C}_j$) and the size of the cluster N_c is $|\mathbf{N}_I(b_j)| + b_j$ for the scenario with uniform traffic and a constant size cluster determined using the method described in Figure 2.4. Hence, the number of channel groups is N_c in uniform traffic. A unique channel group ordered list is assigned to each base station such that $\mathbf{C}_j$ is the highest priority group in b_j and for any two base stations b_j and b_k, a cluster high priority group in b_j is a low priority group in b_k. Using additional a priori information improves the performance (e.g. blocking and dropping probabilities) compared to the Simple DCA at the expense of additional work off-line.

The ordered channel list in Simple DCA and Geometric DCA are constructed prior to network operation and they remain static during network operation. Rather than having a static ordered channel list, a cost function can be used to dynamically order the channels. By giving a cost to each free channel, the algorithm will assign a free channel with the lowest cost and reassign a high cost channel currently in use to a lower cost channel when it becomes free. For example, in the Nanda-Goodman DCA [34], the cost of using a channel by b_j is the number of base stations in $\mathbf{N}_I(b_j)$ that would be locked from this channel (excluding those base stations $b_k \in \mathbf{N}_I(b_j)$ that are already locked from this channel owing to this channel being used by a base station in $\mathbf{N}_I(b_k)$). Note that this cost function promotes channel packing. In [35], a Two-Step Dynamic-Priority DCA (TSDP) algorithm is proposed whereby a channel can be designated either primary or secondary, where primary channels have a higher priority than secondary channels. A channel in b_j is a primary channel if the number of base stations in $\mathbf{N}_{Co}(b_j)$, using this channel, is above a predefined threshold, otherwise it is a secondary channel. All channels start off as secondary channels and are ordered

according to the Nanda-Goodman cost function until they become a primary channel. Here channels used by a greater number of base stations in $\mathbf{N}_{Co}(b_j)$ have a higher priority and in a tie, the channel that is freed in more base stations in $\mathbf{N}_{Co}(b_j)$ has a higher priority. The primary channel is introduced in TSDP to encourage higher channel packing giving a better performance (in terms of call blocking and dropping probabilities) than Nanda-Goodman and Geometric DCAs. In [36], primary channels are assigned a priori and the cost function is the same as that in Nanda-Goodman but with an additional cost to the base station that uses a non-primary channel. In a Locally Optimised Dynamic Assignment (LODA) [31] the cost encourages packing of channels such that channels are reused as closely as possible.

An aggressive DCA is used in [32] where the channels in $\mathbf{C}_j$ are divided into primary $\mathbf{C}_{1j}$ and secondary $\mathbf{C}_{2j}$ groups, where $\mathbf{C}_{1j}$ are the highest ordered channels in $\mathbf{C}_j$ and have a higher priority than those in $\mathbf{C}_{2j}$. Primary channels in $\mathbf{C}_j$ are owned by b_j, while channels in $\mathbf{C}_{2j}$ can be borrowed by b_j. Whenever a channel is released, the owner of this channel (i.e. the base station where this channel is a primary channel) is given priority to acquire it over other base stations. In the aggressive DCA, when base station b_j runs out of free channel it can take a channel from another base station b_k in $\mathbf{N}_I(b_j)$ (e.g. channels owned by b_j) and force b_k to seek another channel. However, this aggressive channel acquisition is applied only to important calls (i.e. handoff) and the channel is acquired if it doesn't cause too many calls in $\mathbf{N}_I(b_j)$ to seek another channel, which may lead to call termination. Aggressive DCA improves call dropping probability and the rearrangement of channels within the network causes the base stations to use their own channels, thereby further reducing the channel locking period.

The methods described here require the base station b_j to communicate with base stations in $\mathbf{N}_I(b_j)$ to obtain the channel usage of these base stations. This can be done in the backbone network (usually wired) using a search or an update method [37]. In a search method, the requesting base station, b_j announces that it wants a channel and all the base stations in $\mathbf{N}_I(b_j)$ will respond with a list of channels that they are not using. The requesting base station can construct a list of free channels from these lists. The request is time-stamped so that a $b_k \in \mathbf{N}_I(b_j)$ who is also searching for a channel will defer its response to b_j until b_k has finished searching for a channel for

itself, if b_j's request has a later time-stamp than b_k's time-stamp. In an update method, each base station maintains a list of free channels. The requesting base station b_j selects a free channel from this list and requests permission to use this channel from all the base stations in $\mathbf{N}_I(b_j)$. If all the base stations in $\mathbf{N}_I(b_j)$ grant the use of this channel, b_j will select this channel, otherwise b_j will consider another channel in its list. Whenever b_j acquires or releases a channel it will inform all the base stations in $\mathbf{N}_I(b_j)$ so that they can update their list of free channels. Each request is also time-stamped in the update method so that a $b_k \in \mathbf{N}_I(b_j)$ will reject a channel request by b_j if b_k is also requesting the same channel and has an earlier time-stamp. The search method doesn't maintain a list of free channel and hence channel reassignment is difficult. Channel reassignment is easy to implement in the update method but it requires more signalling as each b_j needs to inform all the base station in $\mathbf{N}_I(b_j)$ of its channel usage.

Categorising the channels into primary and secondary as in the Aggressive DCA and TSDP can reduce the amount of signalling in an update scheme. Here, base station b_j acquires a primary channel whenever it is free and if a secondary channel is required, b_j goes into the update process [37]. Hence signalling is only required for secondary channels. The signalling is reduced further in the Advanced Update scheme introduced in [37] whereby during the update process, b_j needs only to seek permission from the base station k, where the free channel b_j seeks is a primary channel of b_k and $b_k \in \mathbf{N}_I(b_j)$.

In the search method, the first base station to send a request gets to pick a channel from a list of free channels. In the update method, the first base station to select a contention free channel gets to use it. In high traffic, there are less contention free channels and the update method takes longer to find a channel than the search method. Based on this argument, [38] proposes an UpdateSearch scheme that combines both the update and the search method. The UpdateSearch scheme has two phases, where the first phase uses an update method while the second phase uses a search method. This method begins with the first phase (update method) and moves into the second phase if the first phase fails (e.g. after a few trials). As the update method is more likely to fail in high traffic, this method will move into the search method that has a lower and bounded time to find a channel.

2.2.2.2 Fully Centralised System

A fully centralised DCA has global knowledge of the network, and knowing the channel usage of each base station, each call arrival can be assigned a channel such that the overall assignment $\mathbf{s}(t)$ approaches the optimal solution. Given that FCA by itself is a NP-complete problem, finding the best channel assignment during network operation demands high processing resources and may be limited to small networks only. Hopfield Neural Networks can be implemented using parallel processors as proposed in [40] for a fully centralised DCA. Here, each base station (e.g. b_j) has an energy $\boldsymbol{E}(\mathbf{s}_j(t))$ that is a function of the channel assignment of b_j (i.e. $\mathbf{s}_j(t) = \{s_{j,1}, s_{j,2}, \ldots s_{j,C}\}$, where $\mathbf{s}_j(t) \in \mathbf{s}(t)$) and $\mathbf{s}(t\text{-}1)$. $\boldsymbol{E}(\mathbf{s}_j(t))$ is defined such that it conforms to the compatibility matrix, satisfies the traffic demand at time t for b_j, promotes channel packing, favours assignments that follow a reuse pattern (similar to Figure 2.4) and tries to maintain the previous assignment $\mathbf{s}_j(t\text{-}1)$ so that less channels are reassigned. $\boldsymbol{E}(\mathbf{s}_j(t))$ also balances the importance of each factor by assigning weights to them, and for each call arrival it converges quickly to the first sub-optimal value giving its corresponding channel assignment $\mathbf{s}_j(t)$. Although $\boldsymbol{E}(\mathbf{s}_j(t))$ settles upon the first sub-optimal value, it performs better than DCA with centralisation up to the interference neighbourhood. An additional factor that encourages the network to reserve channels for handoff is introduced in $\boldsymbol{E}(\mathbf{s}_j(t))$ in [41] in order to improve the dropping probability. The handoff constraint is a soft constraint since it is handled in $\boldsymbol{E}(\mathbf{s}_j(t))$ and hence produces a better balance between call blocking and call dropping probabilities compared to [40] and methods that reserve a number of channels solely for handoff.

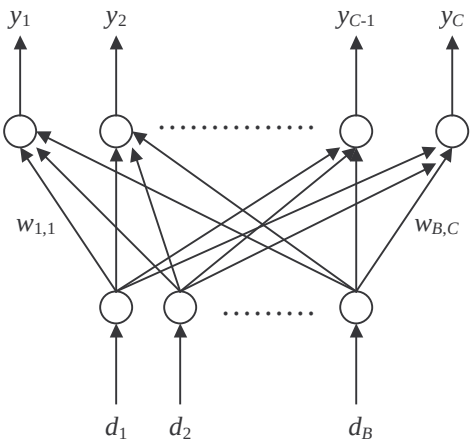

Figure 2.16: Self-organising neural network, SONN

Consider the neural network in Figure 2.16, with input $\mathbf{D}=\{d_1, d_2, \ldots d_B\}$ and output $\mathbf{Y}=\{y_1, y_2, \ldots y_C\}$. Each input d_j is weighted by multiplication with $w_{j,k}$ before connecting to output y_k. Each output is a summation function of the weighted inputs. For a specific input **D**', the output node that is "closest" (user specific definition) to the input will produce a maximum output y_{j*}. The weights connected to the winning node y_{j*} are changed so that y_{j*} will be "closer" to **D**'. In a Self-Organising Neural Network (SONN), the neighbouring nodes $\mathbf{Y}_{Nj*} \in \mathbf{Y}$ of the winning node y_{j*} are also allowed to change their weights. In [26], a SONN as in Figure 2.16 is proposed for use in DCA where the inputs are the traffic demand vector $\mathbf{D}(t)$ and the output y_k is the cost of using channel $c_k \in \mathbf{C}$. The weights $w_{j,k}$ represents the probability that channel j is assigned to base station k in a similar manner to the assignment vector in (2.3) and each output represents the cost of using the channels. The cost is proportional to the degree each channel violates the compatibility matrix (i.e. a soft constraint). The "closeness" in this model is the cost. The weights are probabilities and are transformed into an actual assignment $\mathbf{s}(t)$ in the form of (2.3) using a HNN with an appropriate energy function. This method was proposed for use in DCA. However, it is tested in FCA and its performance is worse than one using a HNN with hill climbing as described in the same publication [26].

If no free channel is available in b_j, a new call can be serviced by reassigning a call in another base station so that a channel is freed in b_j. In a fully centralised system, channel reassignment can be made more aggressive such that a call is only blocked if no channel reassignment is possible in the entire network to free a channel. The idealised method would reassign the channels such that only the minimum number of channels C_{MIN} is used at all times. This idealised method is called Maximum Packing, (MP) [42] and it is the optimal DCA strategy among the Non-Measurement channel allocation methods. However, the complexity of the MP algorithm is high, there is a large amount of communication with all the base stations and finding C_{MIN} is NP-hard. Hence, implementing MP is difficult and is practical only for small networks.

In MAXAVAIL [43], each call arrival is assigned a free channel such that the selected channel leaves the maximum number of free channels for the entire network. A call is blocked if there is no free channel in the base station when a new call arrives. This requires the system to go through every free channel in the base station and compute

the total number of free channels in every base station if this channel is used. Although MAXAVAIL tries to maximise the number of free channels, there is no way of telling whether C_{MIN} channels are in use at all times. A more aggressive algorithm ReMax1 [43] applies channel reassignment and MAXAVAIL. In ReMax1, if a call arrives at b_j and no free channel is available, it will consider all the channels not in use by b_j that will be interfered with exactly one on going call, if it was used. Let $\mathbf{L}_1$ be the list of calls using these channels. ReMax1 applies MAXAVAIL to each call in $\mathbf{L}_1$ as if it were a "new call arrival" at the corresponding base station. The call in $\mathbf{L}_1$ that returns the highest number of free channels is reassigned so that the channel can be used by b_j to service its new call arrival. A call is blocked in ReMax1 if every call in $\mathbf{L}_1$ is blocked. If a call is blocked using ReMax1, a more aggressive method ReMax2 will use ReMax1 on every call in $\mathbf{L}_1$ instead of MAXAVAIL. The reassignment can be taken to higher levels (e.g. ReMax3, etc.) but the complexity of doing so increases exponentially.

MAXAVAIL closely implements MP and is claimed to be one of the best strategies in this class [44], however the amount of computation for each assignment is high. Most DCA methods consider the immediate cost of assigning a channel and ignore the long-term cost (immediate and cumulative future cost). In Reinforcement Learning [17], an agent learns a policy (e.g. a DCA strategy) that minimizes the *expected* long-term cost by repeatedly interacting with the environment. If a system can be modelled in a state-space diagram, the expected long-term cost is calculated using the cost at each possible state and the probabilities of transiting to each state. This information is usually not available. Q-learning which is a subset of Reinforcement Learning that does not require this information but manages to estimate the cost by learning. In [44] a DCA method using Q-learning is proposed, where a state is a call arrival in a base station and the cost encourages high channel packing in a similar way to the one proposed in LODA [31]. As the network operates, the DCA method will eventually converge to a policy that will select a channel for each call such that the expected long-term cost is minimized. This method requires less computation, and it is shown in [44] to perform as well as MAXAVAIL.

An MP implementation called the Simplified Maximum Packing method is proposed in [45]. In this method, each cell is given a set of primary channels in a similar

manner to the channel borrowing methods. If no primary and non-primary (channels not used within the interference neighbourhood) channels are available to service a call, a method similar to ReMax1 is used to reassign an existing call to free a channel. Here the lowest cost channel is the highest packed channel (i.e. the co-channel base stations are as close as possible). Channel reassignment is also performed when a call is completed, where the highest cost channel is reassigned to a newly freed lower cost channel.

2.2.3 Distributed Measurement

The channel allocation methods in the non-measurement quadrant depend strongly on the compatibility matrix. However, most publications assume the compatibility matrix is given or that the interference neighbourhood is defined. In constructing the compatibility matrix, the reuse distance can be found either by measurements or by estimation using an appropriate propagation model. In a typical network, the cell radius may be different, and hence the interference neighbourhood for each base station may be different and needs to be defined prior to network operation.

If the cell radius is constant in a network as in Figure 1.1 the reuse distance will be the same for all base stations and thus the interference neighbourhood is the same for all base stations. However, the number of co-channel cells still needs to be considered in order to construct the compatibility matrix and to determine the SIR. Let $\mathbf{B}_{Cj}$ be the set of K nearest co-channel base stations of b_j in the first ring of interferers. The worst case SIR (i.e. at the edge of the cell), considering only the first ring of co-channel interferer, is calculated using:

$$\text{SIR} = \frac{P_{RX}}{P_{INT}} = \frac{P_{RX}}{\sum_{k=1}^{K} P_k} \approx \frac{1}{K}\left(\frac{R_u}{R_b}\right)^{\zeta} \tag{2.11}$$

where, P_{RX} is the received power at b_j from a subscriber at the edge of the cell, P_{INT} is the total received interference power, P_k is the received power from a subscriber in $b_k \in \mathbf{B}_{Cj}$ (base station k), R_b is the cell radius, R_u is the reuse distance and ζ is the path loss exponent. For example, if K = 6 and ζ = 3.8, a cluster size N_c = *7* would give a SIR of 17dB. However, if the cumulative interference from the 2nd and 3rd ring of co-channel cells are also considered the SIR may drop by 1dB thereby underestimating

the reuse distance. Shadowing and fast fading may alter the instantaneous received and interference power by up to 10 dB (if no averaging is undertaken). On the other hand, not all co-channel base stations use the same channel at once. If there are only 2 co-channel base stations using the same channel, it is possible to use the same channel closer than the worst case reuse distance since the total interference power may still meet the required threshold SIR thereby packing the channels tighter. Hence, a compatibility matrix considering the worst case (i.e. all 6 co-channel interferers) may have over estimated the reuse distance leading to a bigger interference neighbourhood and an optimal solution requiring a higher C_{MIN}. Only FCA using the Tabu Search [24] considers the cumulative interference but the compatibility matrix is still static (i.e. it considers the worst case scenario). It is shown in [46] using simulation and numerical methods, that the reuse ratio for a FCA using a static compatibility matrix (i.e. one that caters for the worst case scenario) is higher than that of a DCA method using measurement (e.g. DECT, PHS and CT2). The reuse ratio also changes according to the traffic pattern giving a different compatibility matrix, which leads to a different optimal solution for each traffic pattern. The optimal solution for the worse case scenario is hence similar to the optimal solution in the non-measurement methods.

In addition to having a static compatibility matrix, non-measurement methods may not cope in a system operating in an unlicensed band. The network may satisfy all compatibility matrix requirements but there is no allowance for avoiding an interferer from a foreign network.

Instead of estimating the reuse distance prior to network operation, measurements of the interference power or SIR can be performed by the base station or the subscriber unit to evaluate the interference environment during network operation. With measurement, the system can judge whether using a particular channel is able to satisfy the required SIR. Using measurement the network obtains the most current channel usage thereby achieving higher channel packing (i.e. the compatibility matrix is now dynamic) and can avoid interference from a foreign network if the system uses an unlicensed band. Without any centralisation, the DCA algorithms in this category are simple and easy to implement. However, measurements cause delay which translates to slow call setup in a voice network or lower data throughput in a data

network. Also since each base station has only local knowledge (via measurements) of the network, it is difficult to achieve the optimum solution and the channel selection is usually used to benefit itself rather than for the network as a whole.

The accuracy of measurement is not perfect and may be affected by the duration that the measurement is performed on a specific channel, how often it is performed and the quality of the equipment used (e.g. frequency synthesiser switching speed). A more accurate measurement can be obtained from a synchronised system since specific time slots can be used by the base stations to transmit pilot tones while the subscribers (or other base stations) perform measurements [61]. It is shown in [57], that a synchronous system performs better than an asynchronous system and the difference is higher in a TDD system. Blind spots [57] can happen where one unit (e.g. subscriber unit or base station) is performing a measurement and another unit is not transmitting (e.g. the second unit also performing measurement, is switching frequency or is in a guard time slot), thereby causing the first unit to "miss" the second unit. This may cause the same channel to be selected by both units thereby interfering with each other.

2.2.3.1 Least Interfered (LI) Method

The most popular DCA in this class is the Least Interfered (LI) method. This method is used in the DECT [47] and CT2 (cordless telephone equipment) [48] systems. In LI, the base station or subscriber measures the interference power of all available channels and selects one with the least interference power [49], [50], [52], [53], [56]. In [50] LI is used in a "quasi-fixed" channel assignment, where channel assignment is periodically performed off-line (e.g. at night when traffic is low) or whenever there are changes to the interference environment (e.g. the addition of new cells). An iterative process is used, where in each iteration, the base stations take turns to measure the interference and select a channel using LI while the others transmit (i.e. only one base station performs measurements at a time). Using LI, a channel is selected one at a time and a final channel assignment is obtained when no base station changes its channel after two consecutive iterations or when a specified number of iteration is reached. This process can be automated and is able to achieve an acceptable assignment within five iterations. Hence, it uses less resource than FCA (using a priori planning) but central coordination is required to ensure only one base

station performs measurement at a time. For a distributed version, the base station performs measurement at a random time and if the measurement is performed quickly, the probability of more than one base station performing a measurement is low.

When a new call uses a channel, it will change the interference environment and may cause interference for existing users, changing their desired SNR especially in a system using power control. As LI ignores the required SNR, the call admitted using LI might not have an acceptable SNR. This newly admitted call could also jeopardise the existing calls. Channel probing is proposed in [51], where a subscriber selects a channel using LI and tests this channel prior to using it. Here, the base stations are synchronised and power control is used. The channels are divided into FCA and DCA channels. A new call will be assigned to an FCA channel, and the subscriber will test a selected channel by transmitting at a specified time. As the base stations are synchronous, the existing subscribers using this channel can measure the interference change and adapt to it using power control. The new call will then check to see if the changed interference is acceptable. The FCA channels are also used as a buffer such that whenever an existing call cannot meet its SNR requirements, it can use a FCA channel while probing for another channel. Power control also improves the performance of LI and the simulation performed on a one-dimensional network in [66] shows that LI with power control has a lower call blocking probability than one that does not use power control (i.e. pure LI).

2.2.3.2 Threshold Based DCA

LI doesn't differentiate whether a channel will provide an acceptable SNR, the channel is selected simply if it is the least interfered channel. There are a group of DCA algorithms that consider whether a call has acceptable SNR. These DCA algorithms are called threshold based DCAs, where the algorithm selects a channel based on a predefined interference threshold. The interference threshold is determined from the required SNR and the signal power of a subscriber at the corner of a (hexagonal) cell. Examples of such algorithms are the Least Interference below Threshold Algorithm (LTA), the Highest Interference below Threshold Algorithm (HTA), the Marginal Interference Algorithm (MIA) and the Lowest Channel below Threshold Algorithm (LCA) [52], [53]. In LTA, the least interfered channel below a

threshold is selected to ensure that only tolerable interference is accepted. HTA selects the highest interfered channel that is below a threshold and this encourage high channel packing as the co-channel base stations are made closer together. MIA is a hybrid of HTA and LI that tries to achieve higher channel packing and maintain good signal quality. MIA is same as HTA if the measured interference is below a threshold otherwise it selects the least interfered channel (similar to LI). LCA selects the lowest numbered channel that is below a threshold and it tries to pack the channel as closely as possible. In these algorithms, the threshold is important since if the interference threshold is set too low (i.e. high signal quality) more calls will be blocked, since this is difficult to achieve. On the other hand, if the interference threshold is set too high (i.e. low signal quality), this may result in uneven channel utilisation for a threshold based DCA that encourages channel packing (e.g. HTA, MIA and LCA). Channel utilisation is defined as the number of base station using a specific channel and in a highly utilised channel, the base stations/subscribers will experience a higher interference level compared to those in a lower utilised channel. It is shown in [52], that LI is able to achieve even channel utilisation and in the "quasi-fixed" system using the technique described in [50] (where the base station performs the measurements and they are synchronised) convergence to a stable channel assignment is guaranteed (assuming that each measurement correctly finds the least interfered channel). LI also consistently performs better – in terms of signal quality (SNR) and call blocking probability – compared to the threshold based DCAs (LTA, HTA, MIA and LCA) in both a quasi-fixed and in a DCA system [52], [53]. The performance of LTA approaches that of LI at a high interference threshold but degrades at a low interference threshold. The performance of MIA is opposite to that of LTA – performance approaches LI at a low interference threshold and degrades at a high interference threshold. In light traffic, HTA and LCA have a performance approaching LI at a low interference threshold and degrade at a high interference threshold. However, in heavy traffic, the performance of HTA and LCA peak at a moderate interference threshold and degrade at both low and high interference thresholds. The performance of MAXAVAIL is compared with LI in [53] where shadowing, fading and the received signal strength of each call are considered. In most publications, which evaluate DCA methods in the non-measurement class, the simulation results presented ignore the received signal quality and the propagation model (where shadowing can change the signal strength by 10dB) since they assume

that the compatibility matrix will ensure that the SNR threshold is met. In [53], it is shown that LI performs better than MAXAVAIL and that the MAXAVAIL performance is comparable to LI only at light traffic loading and a high reuse factor (e.g. 19). Simulations in [53] conclude that DCA methods that try to achieve high channel packing (e.g. HTA, LCA and MAXAVAIL) perform worse than DCA methods that try to minimize interference.

Instead of using interference power as in LI, an estimate of the SIR/SNR can be used. Using SIR/SNR the system can immediately determine whether a call using a specific channel is acceptable and unlike in an interference threshold based DCA the task of pre-defining an interference threshold is eliminated. However, SIR/SNR is difficult to estimate. When a mobile subscriber selects a base station for a call, it measures the signal strength of several base stations and usually selects the base station with strongest signal strength. In [54] the mobile also measures the interference power of all the channels and using the signal strength from the selected base station, it can compute the downlink SNR. The mobile requests permission from the base station to use the channel with the highest SNR. The base station will estimate the uplink SNR and will use this channel if both downlink and uplink SNR are above a threshold. This is a crude way of estimating the SNR since the signal strength from the base station or mobile also contains interference power and it is difficult to separate interference from the signal. Apart from SIR and interference power, the error rate in the packets can also be used. For example, in the Cellular Digital Packet Data (CDPD) used in AMPS to provide a data service, the block error rate (BLER) is used instead of SIR and is continuously monitored [55].

A SIR threshold based DCA (Marginal SIR) similar to the MIA is introduced in [56]. As in MIA, the Marginal SIR selects the highest SIR channel not exceeding a predefined threshold and if the SIR exceeds the threshold, the lowest SIR channel exceeding the threshold is selected. Marginal SIR is compared with LI and RND using various SIR thresholds for the Marginal SIR algorithm. The 1-percentile SIR (1% SIR) performance – in the Cumulative Distribution Function (CDF) – shows that RND performs the worst and over a specific range of SIR thresholds, Marginal SIR performs (1% SIR) better than LI. As described previously, the performance of MIA in [52] and [53] can approach that of LI, albeit that a different performance

measurement is used. Hence a well defined threshold can have a significant effect on the performance of a threshold based DCA.

In a measurement based DCA, the base station has an advantage over the subscriber's unit in measuring interference. Firstly, the base station is usually located at a higher position and the interference (from other base stations) measurement will be less likely to be affected by shadowing [50]. Secondly, better equipment can be used at the base station (since it is more costly to put such equipment into every subscriber's unit). However, the uplink and downlink may experience different levels of interference especially for a TDD system. A balanced-DCA method is proposed in [57] whereby the base station and subscriber perform the measurements. In this method, the base station measures the interference of all the channels and selects N_s lowest interfered channels. The list of N_s channels is passed to the subscriber unit. The subscriber measures the interference power of the best channel in the list and estimates its downlink SNR. If the downlink SNR is above a threshold, the subscriber will transmit a request using this channel to the base station, where the base station will estimate the uplink SNR of the channel. If both downlink and uplink SNR are above a threshold, the channel is selected otherwise the next channel in the list is considered. Each channel is given several trials before proceeding to the next channel in the list. Simulations performed in [57] show the proposed DCA method performs better than a DCA method where either the base station or the subscriber alone performs the measurements. This method was also compared with a FCA method for both non-uniform and uniform traffic distributions and the results show that the DCA method has lower blocking probability than FCA for a range of offered traffic levels (in Erlangs) [58]. FCA is designed for the worse case scenario and consequently for those calls that are admitted (SIR above threshold) the quality (SIR performance) of these calls are slightly better than those when using DCA.

2.2.3.3 Priority Based DCA

The LI algorithm and the threshold based DCA algorithms described previously require the base station or subscriber units to measure the interference power of all the available channels. This takes a long time causing delay at the call setup and lowering the data throughput in a data network. Rather than going through all the channels, the First Available (FA) DCA [59] takes the first channel in a list that meets

the threshold requirement. Evaluation of the SIR and interference autocorrelation over time gives a measure of the changes in SIR and interference respectively (e.g. after measurements) and it is found that reducing the call setup delay increases these autocorrelation values [59]. This means that the interference measured at the start of a call will experience greater changes (during the actual call) if the call setup delay is high. Simulations performed in a packet data network in [59] show that FA has a higher SIR and interference autocorrelation than LI. However, FA outperforms LI (in SIR performance) only over a range of low SIR values and at higher measuring rates, LI out performs FA. It is also found that in a circuit switched network, the call setup delay has little effect on the SIR/Interference autocorrelation. In [62] FA-SIR selects the first channel that has a SIR above a defined threshold. If all the base stations consider each channel in the same order (e.g. from lowest numbered channel to the highest numbered channel), the lower numbered channels will be highly utilised. Thus, each base station would have to spend time measuring these highly utilised channels (the lower numbered channels) in order to get to the less utilised channels (i.e. the higher numbered channels). This can be avoided if each base station considers the channels in a different order.

A priority based DCA is used to avoid spending time measuring highly utilised channels prior to reaching a good channel (as happens in FA) by assigning a priority (e.g. a value) to each channel. The channels are ordered according to their priorities, such that the highest priority channel is considered first for assignment. Each base station keeps a different list of ordered channels and in this way, the problem experienced in FA is avoided. This idea is similar to that employed in most Non-measurement Centralised (up to interference neighbour) methods such as Simple [32] and Geometric [33] DCAs. A Channel Segregation (CS) scheme is introduced in [60] where each base station assigns priorities to the channels through learning from experience. The base station performs a measurement and selects the first channel in the list that has a measured interference power below a predefined threshold. No a priori knowledge is required, since the channels are arranged during network operation (the initial order can be random or may follow the channel number). Each base station maintains an ordered list of channels and the priority of each channel is updated using the following:

$$P_{Cj} = \frac{N_A}{N_T} \tag{2.12}$$

Where P_{Cj} is the priority value of channel j, N_A is the number of times this channel is successfully used and N_T is the number of times this channel is considered for use (i.e. being measured). However, as each base station has only local information of the network, the system will converge to a sub-optimal ordered channel list.

A different priority based DCA using a staggered frame is introduced in [61], for a TDMA data network where each base station transceiver maintains a priority ordered channel (a time slot and frequency) list. The base stations are synchronised and grouped so that a staggered frame method is used for the subscribers to estimate the SIR. TDMA is used where in each time frame only one base station in the group is allowed to assign a channel. This base station broadcasts a paging signal to its mobile stations and then turns off its transmitter. This is followed by a pilot tone transmitted by the rest of the base stations in the group. Pilot tones are transmitted only for channels (i.e. time slot and frequency) that are being used. The mobile stations use the paging signal and the pilot tone to estimate the downlink SIR for each channel. This staggered frame method also avoids blind spots in the measurement. Using a similar method to balanced-DCA [57], the mobile station estimates the downlink SIR of several channels and sends a list of channels that have acceptable quality (downlink SIR above a threshold) to the base station. The base station assigns the highest priority channel in the list given by the mobile station with acceptable quality to the mobile station. The channel priority is updated after each channel assignment, where the first channel having an acceptable quality (i.e. downlink SIR > threshold) is swapped with the first channel failing to meet the threshold criteria. The staggered frame DCA has a better SIR performance than CS but only a comparable SIR performance to that of LI (with subscribers performing interference power measurements).

Channel reassignment is included in a priority based DCA in [62]. Two such methods are proposed namely DCA with Weighted Channel Ordering (DCA-WCHN) and DCA with Weighted Carrier Ordering (DCA-WCAR). In DCA-WCHN the priority of a channel (time slot and frequency) is proportional to the number of times it has

been successfully used in a similar manner to that in CS. In DCA-WCAR, the priority is given to each carrier (frequency) rather than channel (combination of time slot and frequency) and it is proportional to the number of active time slots in it. The base station performs the measurement and only acceptable channels are used (i.e. SIR > threshold). In DCA-WCHN the highest priority channel that is acceptable is selected for use, while in DCA-WCAR the first time slot in the highest priority carrier that is acceptable is selected. In both methods, channel rearrangement is implemented such that when a call is released, it is the highest priority channel which is released. This is the opposite of what is done in non-measurement methods and the rational is that the next call will have a higher chance of using a high priority channel. This method is compared with LI and First Available SIR (FA-SIR). In a synchronous system, DCA-WCHN has a slightly lower call blocking probability than LI, while FA-SIR has an inferior call blocking probability to LI, with DCA-WCAR having the worst performance. However, in an asynchronous system, LI has the lowest call blocking probability while DCA-WCHN has the worst call blocking probability. LI performance is consistent but it has the highest call set up delay. A DCA method combining LI and priority based DCA (DCA-LSWO – limited search with weight/priority ordering) is introduced in [62] where the number of channel measurements is limited to a specific number and channel priority ordering is used to compensate for the smaller number of measurements. DCS-LSWO performs marginally better than LI in an asynchronous system.

In [63] a threshold based DCA method is proposed for application to a Packet Reservation Multiple Access (PRMA) protocol in a packet data network. In this method, the base station estimates the SIR of all unused slots by measuring the interference power of the unused slots and taking the received signal power as that of the reservation packet from the subscriber unit. Rather than using an ordered channel list, the base station selects a slot out of K slots at random starting from the first unused slot that has a SIR above a predefined threshold. The value K is decreased if the packet is transmitted successfully otherwise it is increased. By taking a random slot rather than using the first available slot this method avoids the problem faced by FA where different base stations will all begin by searching the lower numbered slots (or channels) that are highly utilised.

2.2.3.4 Other Methods

A permission probability method is introduced in [64] to allocate timeslots in a Packet Reservation Multiple Access (PRMA) protocol applied to a packet data network for a voice service. A low permission probability slot will be less likely to be acquired by a subscriber and vice-versa for a high permission probability slot. The permission probability is dependent upon the interference of the slot and a measure of the traffic intensity. The measured interference determines the link quality while the traffic intensity determines the number of packet collisions. The traffic intensity is estimated by measuring the rate of occupied slots. A high value is given to the permission probability if both interference and traffic intensity are low, while a low value is given if interference and traffic intensity are high. Fuzzy logic is used to determine the degree of interference and traffic intensity and the combination of these two functions gives the permission probability. This Fuzzy-DCA method was compared with RND and LI. The SIR performance of LI is better than that of Fuzzy-DCA, but Fuzzy-DCA has a lower packet dropping probability (i.e. receiving a packet that has a SIR below a threshold) compared to that in LI. RND has the worst SIR and packet dropping probabilities.

A technique known as the smallest difference SNR method is introduced in [65] and is compared with LI. Instead of selecting the highest SNR as in LI, this method selects the channel that will give the smallest SNR difference between the call with the highest SNR and the call with the lowest SNR. In both the LI and the proposed method, the algorithm ensures that a channel selected will not degrade the SNR of the existing calls below a required threshold. No SIR-performance is given but the proposed method has a better blocking probability than that of the LI method.

2.2.4 Centralised Measurement

DCA methods in the centralised measurement category have knowledge of both the channel usage of other cells and also the interference environment. It has the advantage of global knowledge using centralised network information making it capable of achieving optimum solutions. It is also able to achieve lower reuse patterns and a higher SIR by using the measured information. However, it also inherits the disadvantages of centralised and measurement methods: i.e., large amounts of signalling and the delay and accuracy associated with measurements and consequently it can be difficult to implement.

A fully centralised and measurement method is described in [66] called MAXMIN. In MAXMIN the base station assigns a channel that maximises the minimum SIR of all the existing subscribers. This method somehow requires all channels to be tested (e.g. using channel probing) and all the base stations to estimate the SIR of the existing subscriber. A method may be required to test these channels without interfering with existing calls and hence is difficult to implement.

Having centralised information, the information exchange between the subscriber and its base station can be reduced, especially in DCA algorithms where both the subscriber and base station need to perform measurements (e.g. [57]). The use of centralised information to reduce the quantity of measurements is implemented in [67]. Two algorithms utilising the centralised and measurement information are proposed namely the Network Assisted Least Interfered (NA-LI) method and the Network Assisted DCA with Throughput Optimisation (NA-TO). In both methods, the subscriber performs measurements to estimate the path losses from base stations (both serving and are nearby base stations) to itself and this information is sent to its serving base station. The base station uses the channel usage of nearby base stations – presented in the form of a matrix as in (2.3) – and the estimated path loss to calculate the SNR for each channel. NA-LI selects the channel with the highest calculated SNR. The data throughput (bps) is assumed to be a function of the Block Error Rates (BLER) and the throughput for each active user is stored in the base station. NA-TO uses the SNR information to estimate the data throughput of each channel and selects the channel that would give the maximum throughput for the active users including the new subscriber (i.e. it maximises the summation of all throughputs). Simulation results show that NA-TO achieves a higher data throughput compared to NA-LI.

A DCA method is proposed in [68] for use in a hierarchical cellular system consisting of macrocells and a microcells as described in Figure 2.2. This method uses fuzzy logic to measure the degree of channel occupancy and the average time a subscriber spends in a cell such that a fast moving subscriber is assigned to a channel in a macrocell while a slow moving subscriber is assigned to a channel in a microcell. In conditions of low channel occupancy, a slow moving subscriber can also be allocated to a macrocell. This reduces the number of handoffs in the network. A compatibility matrix is defined for the macrocells and each macrocell maintains a table of the

channel usage of its interference neighbouring macrocells and microcells. For the macrocell, the channel that does not violate the macrocell compatibility matrix and has the least number of cells (macro and micro) using it within its interfering neighbours is selected. At the microcell level, the channel with a measured interference power below a threshold and also has the highest usage among other microcells within the same macrocell is selected. This is to encourage channel packing. Otherwise, the channel that cannot be used by the overlaying macrocell is selected (assuming the interference of this channel is confined within the macrocell and therefore does not interfere with other macrocells). If no such channel exists, a channel that is available to its overlaying macrocell is selected (this reduces the capacity of the overlaying macrocell). The blocking probabilities of this method out perform that of FCA under all traffic conditions. However, no comparisons with other DCA methods are presented.

In a Broadband Fixed Wireless Access network, DCA can be employed so that the frequency reuse is aggressive (i.e. frequency reuse of one). A Staggered Resource Allocation (SRA) [30], described in the FCA section and in Figure 2.13 is used for BFWA utilising a frequency reuse of one. An Enhanced Staggered Resource Allocation (ESRA) is introduced in [69] where measurement and coordination among base stations (i.e. centralisation) is included in SRA to improve the packet throughput. ESRA classifies the subscribers into different levels based upon the number of concurrent transmissions that it can tolerate in a cell. To do this a SIR measurement is performed where the base station under consideration transmits a pilot tone while the others stop their transmissions. The subscriber uses the pilot tone as an estimate of the signal strength. The other base station then transmits while the base station under consideration is silent. Using these signals the subscriber can estimate its receive SIR. The classification is done periodically during transmission and if a subscriber continuously needs retransmissions, it will be reclassified to a lower level to reduce interference. In a similar manner to that in SRA, each time frame is divided into six time slots and each sector is given a dedicated slot (1^{st} choice slot) for transmission (if additional slots are required, the sector can transmit in another predefined slot – 2^{nd} choice slot, etc. as in Figure 2.17). ESRA further divides each time slots into six mini-slots (i.e. total of 36 mini-slots). Each mini-slot allows

different degrees of concurrent packet transmission and hence a subscriber that can tolerate six (level 6) concurrent packet transmissions transmits at mini-slot 6.

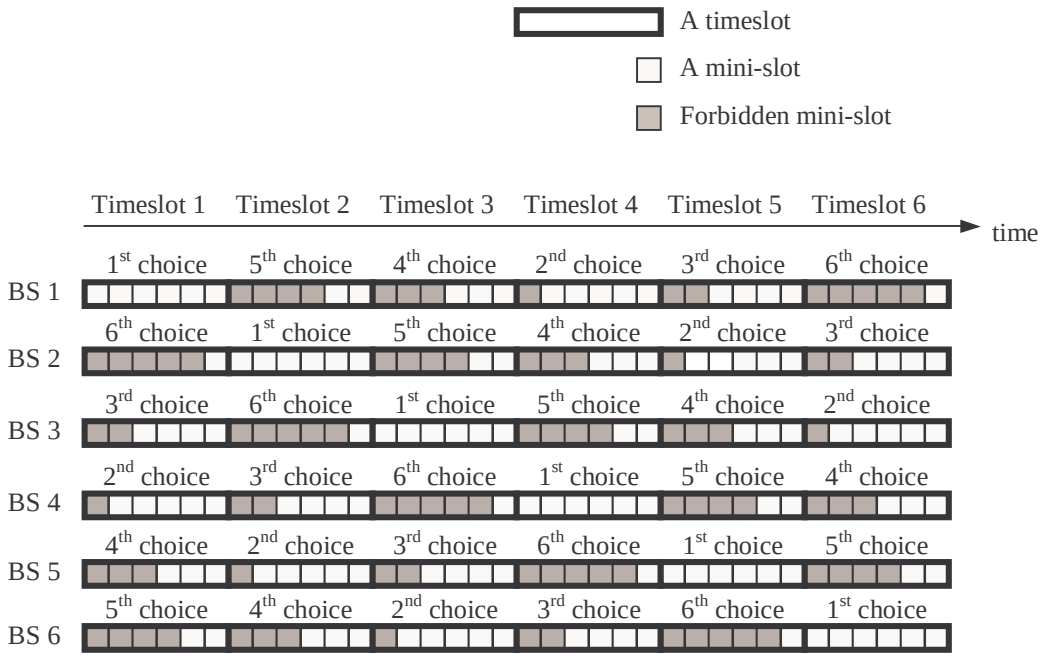

Figure 2.17: ESRA timeslots and mini-slots arrangement in a cel (BS = base station)l

Figure 2.17 shows the timeslot and mini-slot arrangement in a cell with six base stations (as in Figure 2.13) and each base station can transmit only in the non-shaded mini-slots. For example, a level 3 subscriber of base station 1 transmits at the 3rd mini-slot of its 1st choice slot and if it has extra packets, it can transmit them in the 3rd mini-slots of its 2nd choice and 3rd choice slots (3rd mini-slots of higher choice slots i.e. 4th choice and above are not allowed for transmission). This subscriber (level 3) can transmit at a lower mini-slot (e.g. mini-slot 1 or 2) if it is available and not forbidden (i.e. not shaded). Although this level 3 subscriber will not affect other base stations if it transmits in a higher mini-slot (e.g. mini-slot 4), its packet may not meet the required SIR (as at mini-slot 4, there are a maximum of 4 concurrent transmission). ESRA is shown to have a better SIR performance and packet throughput than SRA.

2.3 Conclusion

In this chapter, the Channel Allocation Problem is described and a comprehensive literature review of existing channel allocation methods is presented. The existing methods can be classified into a Channel Allocation Matrix depending upon the technique each method uses to acquire network and/or interference information. The

Channel Allocation Matrix is divided into four quadrants namely Distributed Non-measurement, Centralised Non-measurement, Distributed Measurement and Centralised Measurement.

In the next section, the features of the Broadband Fixed Wireless Access (BFWA) system to be investigated in this dissertation are introduced. Unlike most publications where the emphasis is on voice traffic in a circuit switched network, in this dissertation, the aim is to investigate data traffic being carried in a packet switched network. The simulation model used will also be discussed.

3 SYSTEM DESCRIPTION AND MODELLING

The channel allocation methods described in Chapter 2 are mainly designed for a cellular network offering basic telephony services (i.e., voice) using a circuit switched network where a call occupies a channel for the entire duration of the call. In most channel allocation methods for a voice service, the received signal needs to achieve a threshold SNR otherwise the call will be dropped. From the user's perspective the level of satisfaction does not increase if the SNR improves beyond the threshold. This can be depicted in Figure 3.1, where the payoff quantifies the user's level of satisfaction [74].

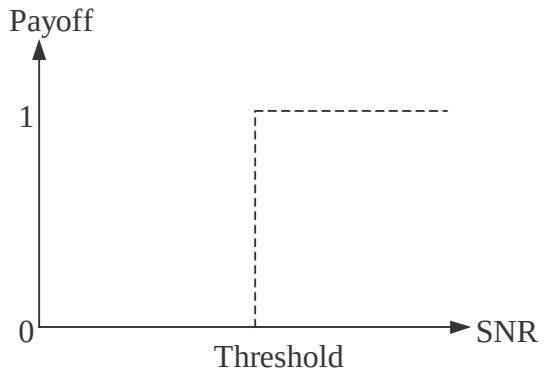

Figure 3.1: User's satisfaction (payoff) against SNR in a voice service.

Since any improvement beyond the required SNR threshold does not benefit the voice user, the SNR performance is usually not used to evaluate the different channel allocation methods. Hence, call blocking and sometimes call dropping probabilities are more important in a voice service. Channel packing is therefore important in channel allocation for voice services since it conserves channels without sacrificing the user's payoff.

In a packet switched network, a channel is used by a subscriber for a short time (perhaps long enough to send several packets). Unlike a voice service, the payoff in a data service utilising a packet network increases with increasing SNR. A high SNR will yield a lower bit error rate (BER) and this leads to reduced packet retransmission and an improved data throughput. In a system with adaptive modulation, a high SNR will allow a user to use a higher order modulation scheme thereby achieving a higher

data throughput. Thus, it is appropriate to use the SNR performance to evaluate channel allocation methods in a network offering a data service. This is in contrast to call blocking and call dropping probabilities, which are appropriate in a voice service. For a fixed signal power and receiver noise, the lower the interference power level in a received signal the higher the SNR (i.e. a higher payoff). As interference power is dependent upon the distance and the position of the interferers, channel packing that minimises the distance among co-channel base stations is not important in a data network and may cause an increase in interference power. DCA methods used for mobile data networks can be found in [55], [59], [61] and [67], while DCA schemes used in Broadband Fixed Wireless Access (BFWA) networks are proposed in [30] and [69]. DCA methods employing a packet switched network for voice services are explored in [63] and [64].

There are relatively few channel allocation methods proposed for a packet switched network offering a data service compared to those for circuit switched networks offering voice services. The focus of this dissertation is on DCA methods appropriate for a packet switched network offering a data service.

3.1 Broadband Fixed Wireless Access (BFWA) System

Broadband Fixed Wireless Access (BFWA) is one of the many potential last mile solutions. The layout of a typical BFWA network is shown in Figure 3.2 and its basic components are the Subscriber Unit (SU), the Access Point (AP) and the Control Server (CSVR).

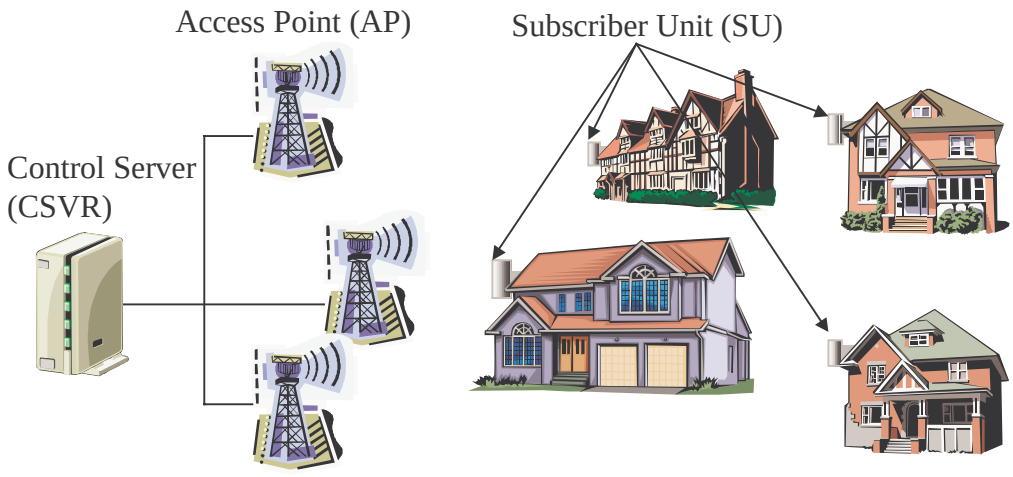

Figure 3.2: BFWA basic components and layout.

The BFWA system under consideration follows the AB-Access system from Adaptivebroadband Ltd and it operates in the unlicensed ISM5 spectrum (at 5 GHz). There are 15 available channels each occupying 15 MHz giving broadband access to the subscribers.

The SU is an integrated device that contains the antenna, radio transceiver, modem and software, and it is mounted on the subscriber's premises as shown in Figure 3.2. The SU uses a directional antenna with a vertical and horizontal 3 dB beamwidth of 20° and a maximum gain of 18 dBi.

The AP is mechanically similar to the SU and functions as a base station. It uses a sectored antenna with a horizontal 3 dB beamwidth of 60°, a vertical 3 dB beamwidth of 7° and a maximum gain of 17 dBi. Six APs are able to provide a 360° coverage for a cell. Several APs are connected to a CSVR. The SU and AP have a transmission rate of 25 Mbps.

The CSVR is a server that provides configuration, authentication and enables the set-up of virtual circuits (in an ATM system). The CSVR can be co-located at the AP's site or it can be connected remotely using a high-speed connection. The CSVRs can all be connected to a central facility where billing and network management operations are performed.

The BFWA system uses Time Division Duplex (TDD) communication. The time frames for the uplink and downlink vary according to the offered traffic and hence it is able to adapt to the asymmetrical traffic that is typical in a data network. The TDD frame size is controlled by the MAC layer and the guard time between the uplink and downlink is also adaptive and is proportional to the round trip propagation delay of the furthest SU. TDD better utilises the available spectrum compared to FDD and it also reduces the complexity of the transceiver as transmission and reception do not occur simultaneously. However, as discussed in previous chapters, TDD has the highest number of possible interference scenarios (SU to SU, SU to AP, AP to SU and AP to AP). As each AP uses an asymmetrical TDD, the APs in the network are not synchronised and therefore DCA methods that assume a synchronised system (as described in Chapter 2) cannot be used here. The interference environment is worse than that in FDD and synchronised TDD system [4].

3.1.1 Media Access Control (MAC) Layer

MAC protocols can be divided into three main categories namely, random access, dedicated assignment and reservation based assignment [70].

In a random access MAC, a user accesses the medium only when it has data to send and the user has to contend for an available channel. Packets will collide when two or more users transmit a packet in the same channel. Examples of random access MACs are ALOHA, Slotted-ALOHA and Carrier Sense Multiple Access (CSMA). Random access is good for packet switched network that are not sensitive to delay (e.g. a data service).

In dedicated assignment, a channel is used by a user for the entire duration of a call. This type of access is good for traffic that is sensitive to delay (e.g. voice, video). However, the bandwidth is not utilised efficiently since the user seizes the channel even if it has nothing to transmit.

Reservation based MAC also known as Packet Reservation Multiple Access (PRMA), requires the user to reserve a channel prior to sending its packet. The system determines the amount of resources (e.g. timeslots) that need to be reserved for each user. The reservation process may be similar to the process in random access where in this case the users contend for a reservation. PRMA is suitable for both types of traffic (i.e. delay and non-delay sensitive). PRMA is used in ETSI's HIPERLAN Type 2 [71] and in Local Multipoint Distribution Service (LMDS) networks [72]. PRMA is able to increase the capacity of a voice service by enabling more than one call per channel [74]. This is possible because typical conversational speech consists of short pauses (silences) and talkspurts. PRMA is used to reserve channels for a user during the talkspurt period and when a user is silent, another user can access the same channel.

The PRMA scheme used in the BFWA under consideration (owing to its flexibility in adapting to different traffic types) is similar to that proposed in [75]. A MAC frame is a transmission cycle comprising of a downlink and an uplink transmissions and has the structure shown in Figure 3.3.

A Frame Descriptor packet is transmitted in the "Header" at the start of each MAC frame, which describes the entire content of the current MAC frame. A SU with

packets to transmit will contend by sending a Reservation packet at the start of one of the Contention Timeslots (CT). The Reservation packet gives the number of uplink timeslots required by the SU. If a SU has won the contention, the AP will assign an appropriate number of uplink timeslots (one UCELLR and/or one or more UCELLs) for this SU in the next MAC frame. This information will be sent to the SU in the next MAC frame in the Reservation Response RR timeslot. Each data packet has a constant size (an ATM cell) and the SU granted the uplink timeslots sends its data packet in the UCELLR and UCELL timeslots. The UCELLR timeslot is slightly longer than that of a UCELL since the SU can request for additional uplink timeslots in the next MAC frame by piggy backing on the UCELLR thereby avoiding the need for contention in the next frame. The AP acknowledges each SU's uplink data burst in the DACK timeslots. If the SU fails to receive a Reservation Response (RR), it will time-out and back-off a random time before contending for a reservation slot (UCELLR and/or UCELL) again. The AP broadcasts downlink data packets using a downlink timeslot (DCELL – one per packet), which is acknowledged by the relevant SUs in the UACK timeslots.

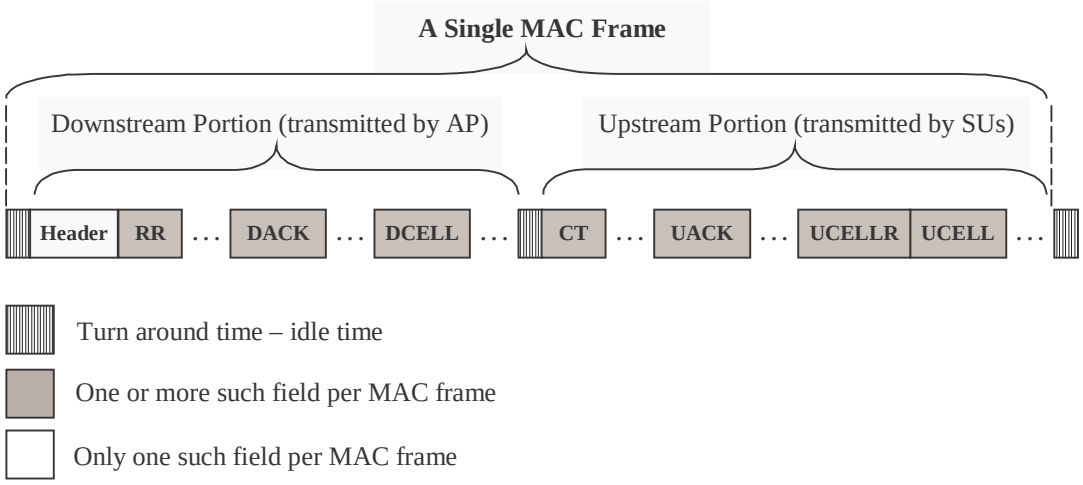

Figure 3.3: A single MAC frame structure.

The AP decides which SU is able to transmit in each MAC frame using a round robin scheme. The number of different types of timeslots (shaded timeslots in Figure 3.3) varies according to traffic load. The maximum number of these timeslots and the maximum number of SUs, N_{SU} allowed to transmit in each MAC frame are fixed. The packet throughput T_{MAC} is defined as the number of data packets (i.e. UCELL) that can be received per normalised second (nmsec). A normalised second (nmsec) is the

amount of time it takes a transmitter to transmit a data packet (i.e. transmission delay for a UCELL of fixed size). Theoretically, the packet throughput T_{MAC} will increase linearly with offered traffic load D up to a point where T_{MAC} is saturated (i.e. it does not increase with offered load).

3.1.1.1 PRMA Performance

The performance of the PRMA described previously is compared with two popular random access MAC schemes namely the ALOHA and Slotted-ALOHA.

In an ALOHA scheme also known as Pure-ALOHA, all the SUs are free to transmit whenever they have data to send. Collisions will occur and packets that collide are destroyed. The theoretical relationship between the packet throughput T_{MAC} and the offered traffic load D is given in (3.1) where the total number of SUs is assumed to be infinite [77]. The packet throughput T_{MAC} and offered traffic load D are measured in packets per nmsec.

$$T_{MAC} = D\boldsymbol{e}^{-2D} \tag{3.1}$$

In a Slotted-ALOHA scheme, time is divided into equal discrete periods. Packets arriving are transmitted at the start of the next timeslot and hence the SUs must be synchronised with each other. Similarly, packets that collide are destroyed. The relationship between the packet throughput T_{MAC} and the offered traffic load D is given in (3.2) where the total number of SUs is assumed to be infinite [77].

$$T_{MAC} = D\boldsymbol{e}^{-D} \tag{3.2}$$

A simulation to investigate the performance of the MAC layers is performed using the OPNET Modeler/Radio software package. A scenario with one AP surrounded by 20 SUs is investigated where each packet arrival follows a Poisson distribution with the total mean traffic due to all the SUs ranging from 0.2 packet/nmsec to 5 packets/nmsec. The maximum number of SUs allowed to transmit per MAC frame, namely N_{SU} is 6 and the maximum number of downlink (DCELL) and uplink (UCELLR + UCELL) data packets allowed per MAC frame are 32 in each direction. The downlink traffic is set at the maximum (i.e. 32 DCELLs transmitted in each MAC

frame). The packet delay is the time a packet spends waiting in a queue before being transmitted and excludes the transmission delay. There is no retransmission of data packets failing to reach the AP. The results are published in [76] and are also presented here.

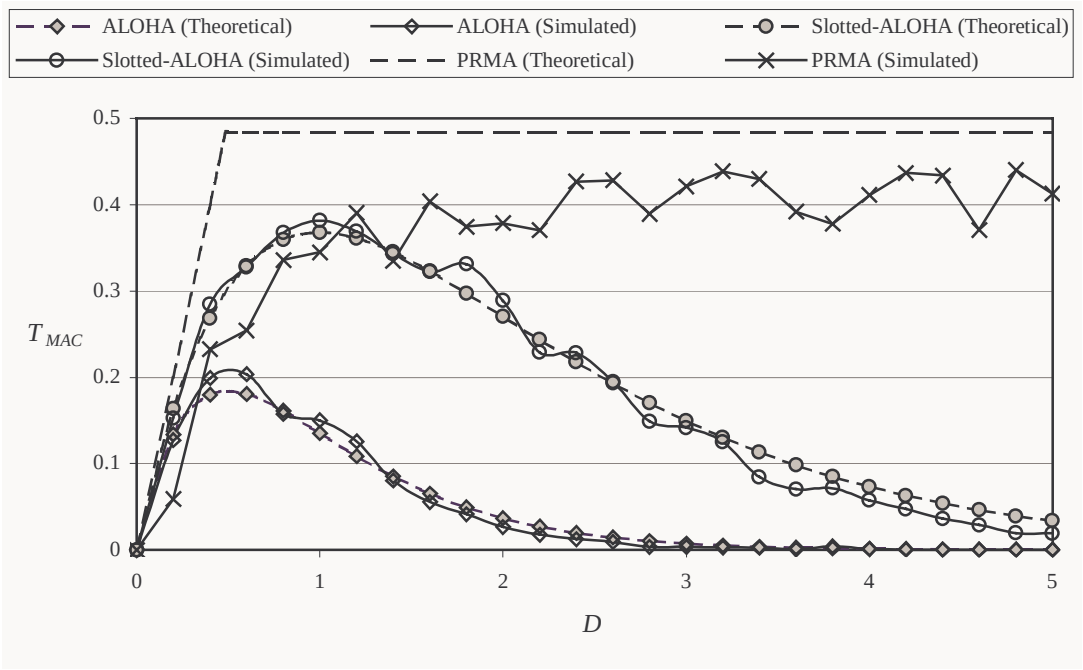

Figure 3.4: Packet throughput T_{MAC} performance (packets per nmsec).

The packet throughput T_{MAC} as a function of the total mean traffic D is shown in Figure 3.4. The simulated maximum value of T_{MAC} for ALOHA and Slotted-ALOHA are 0.189 packets/nmsec and 0.398 packets/nmsec respectively. These values are very close but slightly higher than the theoretical values of 0.184 packets/nmsec for the ALOHA and 0.368 packets/nmsec for Slotted-ALOHA. This is because in the simulation only a finite number of SUs are used instead of the infinite number that is assumed in calculating the theoretical value.

The theoretical PRMA T_{MAC} can be calculated by taking the ratio of the total number of bits occupied by the maximum number of UCELLs in a MAC frame to the total maximum number of bits occupied by a MAC frame (i.e. assuming a maximum number of DCELLs are transmitted in the downlink). The simulated maximum value of T_{MAC} for PRMA is 0.44 packets/nmsec, which is slighty lower than the theoretical value of 0.484 packets/nmsec. This is because the SUs that are not successful during contention need to back-off a random time before trying again. If the number of SUs

transmitting is larger than N_{SU} some of the SUs that are successful during contention may not get any uplink timeslots in the next MAC frame since the AP uses a round robin scheme to allocate slots. This causes the SUs to delay packet transmission, hence lowering the packet throughput. The theoretical maximum value of T_{MAC} assumes that all the SUs need to contend only once and continuously transmit thereafter. This is possible only if the number of SUs per AP is less than or equal to N_{SU}. The maximum value of T_{MAC} is higher for PRMA than that of ALOHA and Slotted-ALOHA. Unlike the situation with a Random Access MAC (i.e., ALOHA and Slotted-ALOHA) the packet throughput of PRMA does not degrade when the offered traffic load increases beyond its maximum packet throughput.

The packet delay (excluding transmission delay and packet retransmission) in ALOHA and Slotted-ALOHA is small. In ALOHA, a packet is transmitted once it is created and hence there is no packet delay (excluding transmission delay and packet retransmission). The packet delay for Slotted-ALOHA is around one nmsec, as the packet will be transmitted in the next available slot.

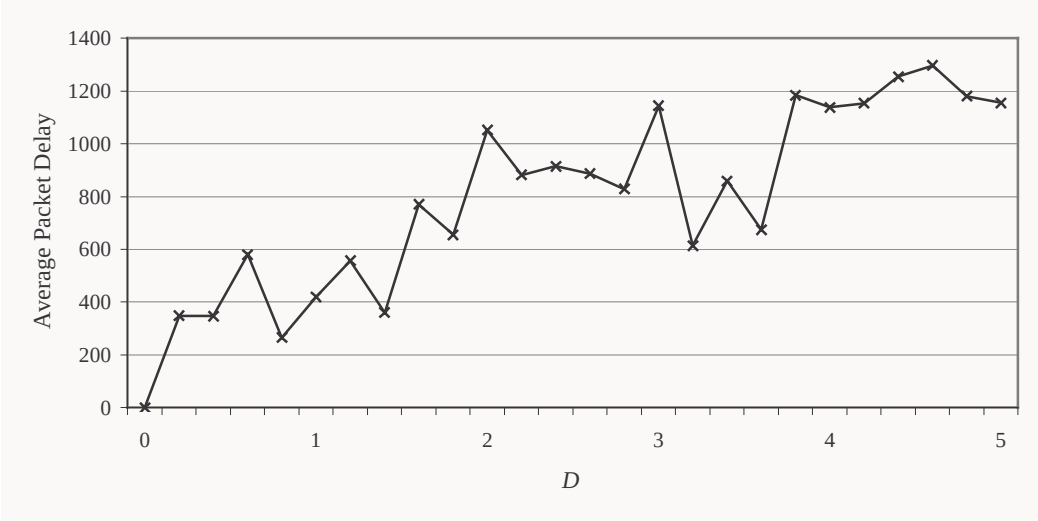

Figure 3.5: Average packet delay measured in number of nmsecs for PRMA.

The average packet delay for PRMA is shown in Figure 3.5 and it can be seen to increase exponentially when the offered traffic load is higher than the maximum value of T_{MAC}. The service time of PRMA is at best one MAC frame, which is much longer than that in Slotted-ALOHA. When the offered traffic load is above the peak packet throughput of 0.44 packets/nmsec, the queue becomes unstable causing the queue size to increase. Hence, the queue delay is several times the MAC frame period causing it

to increases exponentially in terms of nmsec units. The average packet delay becomes worse as N_{SU} becomes larger. Figure 3.6 shows the average packet delay in the vicinity of the maximum value of T_{MAC} for the situation where the number of SUs is made equal to N_{SU} (i.e. six SUs in this case). The queue becomes unstable at demand values to the right of the vertical dotted line (the maximum value of T_{MAC}).

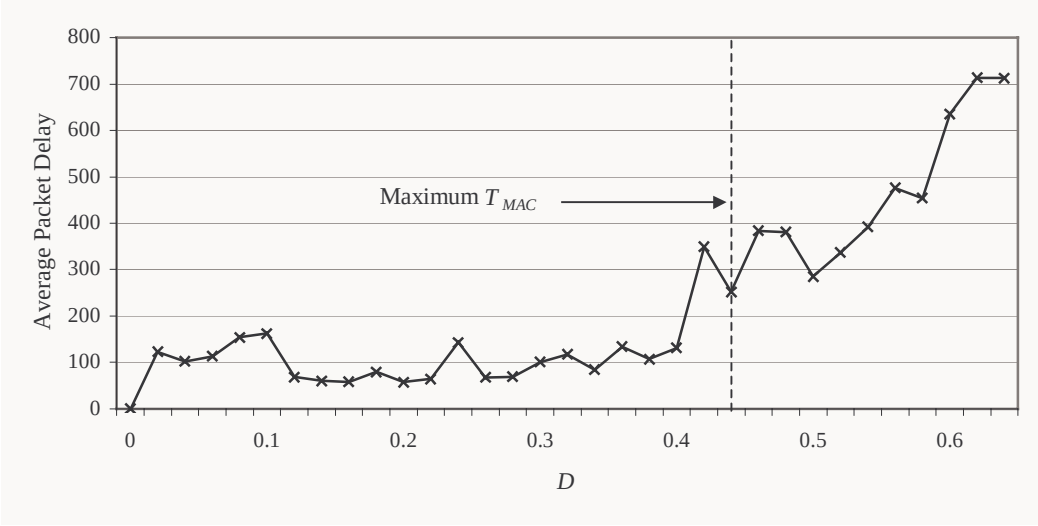

Figure 3.6: Average packet delay at vicinity of maximum T_{MAC}.

PRMA is good for delay and non-delay sensitive traffic owing to its ability to reserve timeslots. The packet throughput does not degrade as compared with ALOHA and Slotted-ALOHA but the average packet delay increases exponentially if the offered load increases beyond the maximum packet throughput.

3.2 Modelling the BFWA System

As highlighted previously, most existing channel allocation algorithms are designed for voice services using a circuit switched network. In this dissertation, the study of channel allocation algorithms is focused on data services using a packet switched network. The BFWA system described previously is used as the basis for this study and its components (SU, AP and CSVR) are modelled using OPNET Modeler.

3.2.1 OPNET Modeler

OPNET Modeler is a program used to model a network for simulation purposes. It allows the communication models to be written in a structured manner. The program has three main domains: the network domain, the node domain and the process domain.

The network domain describes the physical layout and connections (wired or wireless) of the communication devices that form a network. For example, in a cellular network, the cell size, the number of base stations and their locations can be defined here. The network domain enables these communication devices to interact with each other.

A node is a communication device (e.g. an AP or SU) in the network domain. The functionality of a node is defined in the node domain. The node domain contains objects that perform specific tasks. Examples of an object in a node domain are a transmitter, receiver, antenna, queue and server. The objects in the node domain are connected to each other using streams that can pass either a packet or a value.

The functionality of an object in the node domain is defined in the process domain. The process domain is a state space diagram that consists of states and conditions. An object can be in one of these states within the process domain and transit to another state if a specified condition is true or if an interrupt occurs. Interrupts are caused either by another object or by itself (e.g. owing to a time-out event), and the next state, which the object transits to depends upon the present conditions. For example, an object can be in the idle state until a packet arrives at the object causing an interrupt. The object upon this interrupt will check the contents of this packet and if the packet belongs to this object (i.e. the condition is right), it transits from the idle state to another state (e.g. a state that processes the packet).

3.2.1.1 OPNET Radio Pipeline Stages

In OPNET Modeler, the wireless channel is modelled using 14 radio pipeline stages. Each packet sent out by a radio transmitter will go through these 14 stages before the packet reaches the receiver. The packet has built-in attributes that record information such as transmit time, receive time, propagation delay, noise power, etc., which can be retrieved by the program. The values of these attributes are calculated within the 14 radio pipeline stages. The first six stages are performed at the transmitter while the remaining eight stages are performed at the receiver.

In first pipeline stage, the program identifies a receiver group for each transmitter at the start of the simulation. The receiver group contains a set of receivers (within a

node) in the network domain that the transmitter can reach. By default, all receivers are included in the receiver group.

The second stage of the 14 radio pipeline stages calculates the transmission delay at the transmitter. This is taken to be the data rate (bits per second) multiplied by the packet size in bits.

The third pipeline stage is executed at the transmitter and it determines the set of receivers in the receiver group (set at the beginning of the simulation) that the transmitter can reach. A ray-tracing method is used to filter out all the receivers that do not have line-of-sight due to the curvature of the earth between the transmitter and receiver (this model ignores any wave-bending effect). This is performed since the receiver (and/or transmitter) may have moved to a new location since the start of the simulation. For a network that covers a small geographical area, the effect of the curvature of the earth will be small.

The fourth pipeline stage is also performed at the transmitter. Here the program further filters out receivers that are not capable of processing the packet. This is done by matching the bandwidth, frequency (transmit and receive frequency), modulation scheme, data rate and code (for CDMA) of the transmitter and the receiver. The receiver that has the same characteristics as the transmitter will be able to process the packet and this packet is marked as a Valid Packet. If one of the characteristics doesn't match but a receiver's bandwidth overlaps the transmitter's bandwidth, the packet will still reach the receiver but it will be treated as interference.

For each receiver that can process the packet (as a Valid Packet or interference) the transmitter antenna gain is computed at the fifth stage. This depends upon the transmitter antenna and the angle at which the receiver is facing the transmitter. The antenna pattern is defined using the antenna model in OPNET Modeler.

At the sixth pipeline stage, the propagation delays between the transmitter and the receivers (that have satisfied the fourth stage) are calculated. The propagation delay is dependent upon the distance between the transmitter and receiver and the speed of light. As the transmitter and/or receiver can be mobile, two propagation delays are computed, one at the start and one at the end of the packet transmission.

The seventh pipeline stage is performed at the receiver and it computes the receiver antenna gain. This is similar to the fifth stage.

The power of the received packet (Valid Packet and interference) is calculated at the eighth pipeline stage. A free space path loss model is used to calculate the spatial power loss. If a packet is received as an interferer, the total interference power caused by this packet is dependent upon the percentage bandwidth overlap between the transmitter and receiver bands. The program assumes a perfect filter i.e. one that has a perfect cut off at the edge of the pass band as shown in Figure 3.7. For example, in Figure 3.7 the unwanted interference packet occupies a channel with a bandwidth from 100 MHz to 200 MHz while the receiver occupies a bandwidth from 150 MHz to 250 MHz. The packet power is 100 pico-Watts and since there is a 50% bandwidth overlap, this packet will contribute 50 pico-Watts of interference power to the receiver.

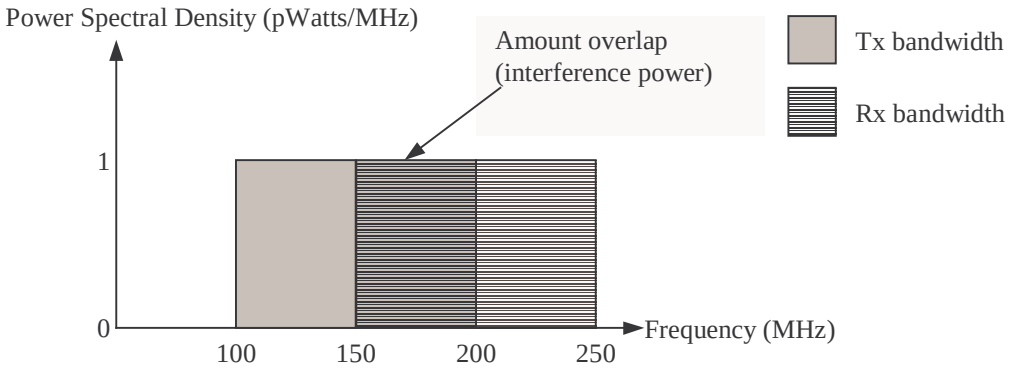

Figure 3.7: Adjacent interference power and the amount of bandwidth overlap.

The background noise is calculated at the receiver in the ninth pipeline stage. The background noise consists of noise other than that produced by other transmitters. For example, the background noise may in part be due to thermal noise due to active and passive components in the receiver.

A new packet may arrive at a receiver while the receiver is still receiving a Valid Packet. This will cause a collision and introduce interference to the Valid Packet. The tenth stage of the radio pipeline stage keeps track of the number of collisions and the interference power. In Figure 3.8, a receiver is receiving a Valid Packet and three packets collide with the Valid Packet at different times. Each packet collision is called a segment as shown in Figure 3.8. For each segment, the interference power is

recorded and the program will return to this stage (following the thirteenth stage) until reception of the Valid Packet (or final segment) is completed.

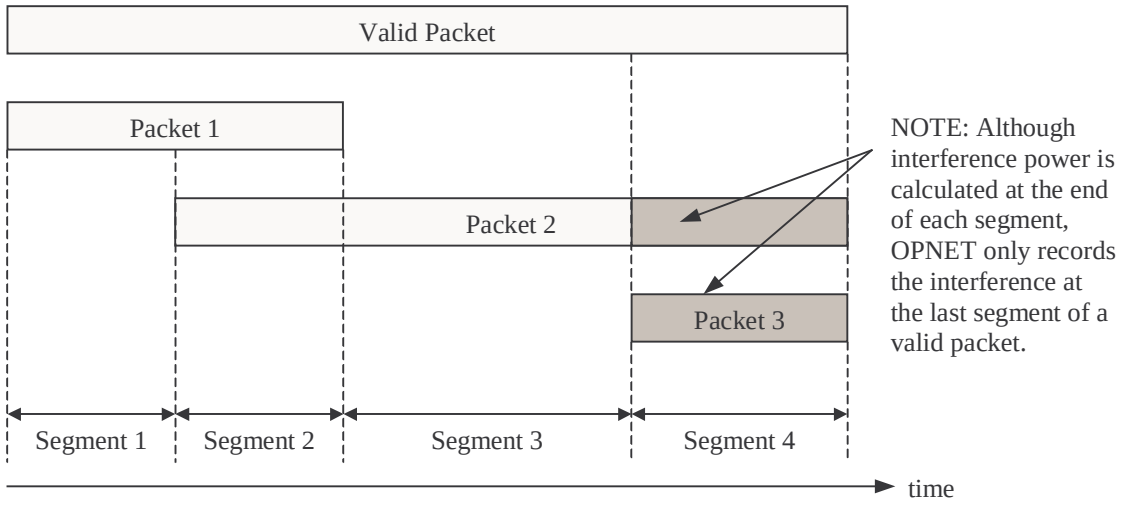

Figure 3.8: Segments used in stages 10 to 13.

At the eleventh pipeline stage, the Signal to Noise Ratio (SNR) for each segment of a Valid Packet is calculated. The signal power is taken to be the received power of the Valid Packet calculated at the eighth stage while the noise power is the sum of the background noise and the total interference power calculated in the ninth and tenth stages respectively. The SNR is calculated for each segment.

Using the calculated SNR, the twelfth pipeline stage calculates the bit error rate (BER) for each segment of the Valid Packet. Apart from the SNR, the BER also depends upon the modulation scheme used. The modulation scheme is defined by a modulation curve, which forms a look up table for the BER as a function of the SNR.

At the thirteenth pipeline stage, the program calculates the total number of bits that are in error based on the BER estimated in the twelfth stage. The program will return to the tenth stage if reception of the Valid Packet is not complete and the interference power, SNR, BER and number of bit errors are calculated for each segment.

In the final stage, the fourteenth pipeline stage, the program decides whether the Valid Packet can be accepted based on the number of bit errors it contains. If the ratio of the number of bit errors to the total number of bits in the packet is below a threshold value, the Valid Packet is accepted. The threshold value is set by the user.

The interference power and SNR are recorded in the packet attributes and the previous segment values are replaced by values from the following segment. Hence, the program only records the interference power and SNR of the final segment of the Valid Packet. For example, in Figure 3.8 three packets collide and interfere with the Valid Packet but only the interference power of the final segment (i.e. from Packet 3 and part of Packet 2) is recorded after the packet passes through to the final stage. If the Valid Packet is longer than the interfering packet (i.e., Packet 2 and Packet 3 end earlier than the Valid Packet in Figure 3.8), the last segment will contain no interference and lead to no interference being recorded. The overall SNR (and interference power) is important in the performance evaluation of a channel allocation algorithm for a data service. Consequently, the radio pipeline is modified so that it also calculates and records the average interference power (of all segments) of the Valid Packet.

For a network that has $N_{Tx\text{-}Rx}$ pairs of "valid" transmitters and receivers (i.e. a transmitter and receiver pair where a Valid Packet exists), the number of times the program needs to go through all 14 stages is proportional to $(N_{Tx\text{-}Rx})^2$. Since the radio pipeline is very detailed and the program has to go through the radio pipeline numerous times (particularly stages 10 to 13), OPNET Modeler consumes a lot of processing power. A simulation with a reasonably sized network with a few hundred APs and SUs takes up to 14 days to complete in a dual processor (Pentium III, 700 MHz) PC. Consequently, modifications are made to the radio pipeline stages in order to reduce the amount of processing. The third pipeline stage is eliminated since it is assumed that the effect of the curvature of the earth is negligible for the BFWA network (that covers a geographical area of about 8km × 7km). As the BFWA system uses a TDMA mode, the code (CDMA feature) used in OPNET is used so that packets from co-channel APs do not pass as Valid Packets (and hence will not go through all 14 stages unnecessarily) but they will act as interference.

3.2.2 Wireless Propagation Model

The wireless propagation model employed by OPNET Modeler assumes a free space environment. This is unrealistic and a Random Height propagation model suitable for BFWA is employed [78]. In the Random Height model, the path loss exponent ζ is 2 (i.e. free space) for distances up to 1km and increases to 3.8 thereafter. This

characteristic is shown in Figure 3.9. Shadowing is implemented using a log-normal distribution with a standard deviation of 3.5 dB [78].

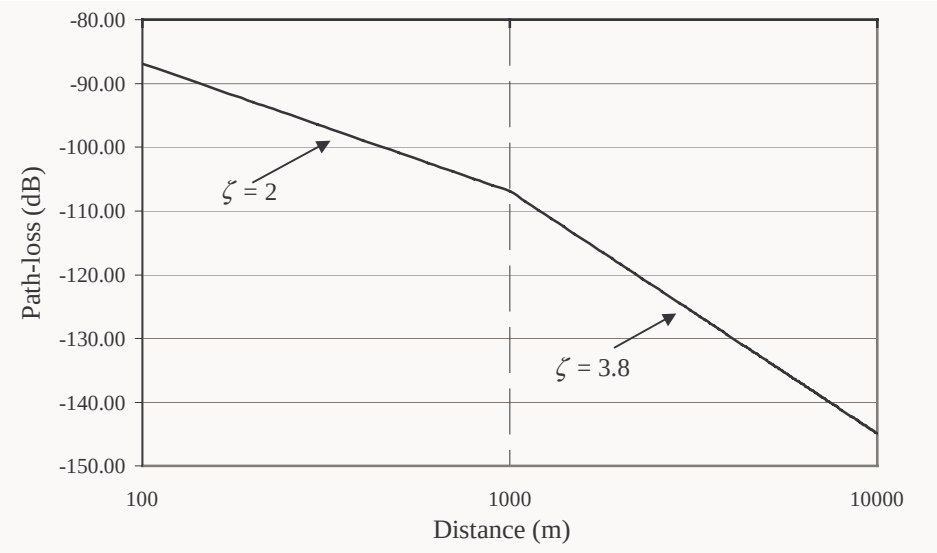

Figure 3.9: Random Height propagation model

3.2.3 Antennas

The SU uses a directional antenna with a 3 dB beamwidth of 20° and a front-to-back ratio of –23 dB and a maximum gain of 18 dBi. The SU is programmed to point directly to the AP antenna. The azimuth and elevation patterns of the SU antenna are identical. In OPNET, the antenna gain characteristic has a resolution of 5°. For example, if a user is facing the antenna at 13° from the main beam, its gain will remain the same while the direction is between 10° and 15°). The antenna pattern implemented in OPNET is shown in Figure 3.10.

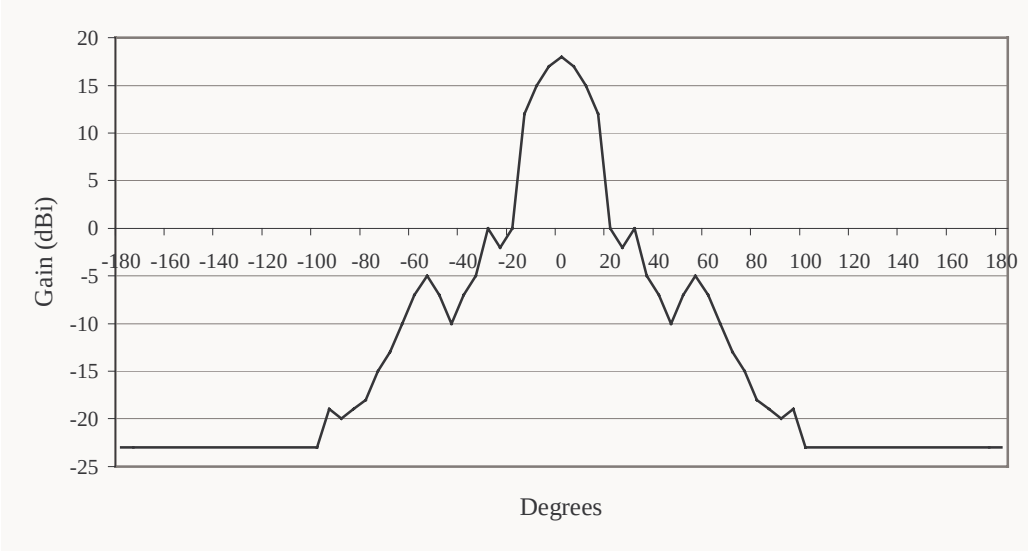

Figure 3.10: SU antenna pattern –the azimuth pattern is the same as the elevation pattern.

The AP uses a sectored antenna with a maximum gain of 17 dBi. The 3 dB beamwidth for the azimuth is 60° while the 3 dB beamwidth for the elevation is 7°. The front-to-back ratio is –23 dB. The main front lobe of the AP points to the middle of the cell at the average height of the SU antenna. The azimuth and elevation pattern implemented in OPNET are shown in Figure 3.11.

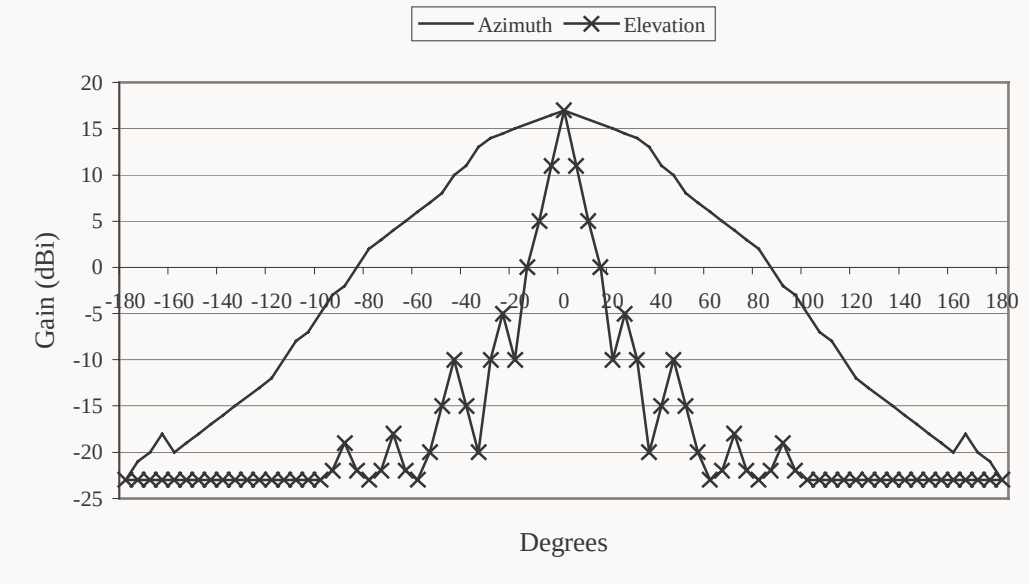

Figure 3.11: AP azimuth and elevation antenna patterns.

3.2.4 Traffic Model

In a circuit switched network providing a voice service, the traffic load is usually modelled as a Poisson process with an exponential service time [53] where the mean

call duration is 3 minutes [62]. However, it is shown in [79] (from Ethernet traffic measurements at different points in the network with a variety of applications from emails to voice calls) that Ethernet traffic exhibits a self-similar nature. A system exhibiting self-similar characteristics is scale invariant, where a similar pattern exists if observed using a different scale. An example of a self-similar characteristic can be found in tree branches, an example of which is shown in Figure 3.12, where the branches have similar patterns at different magnifications.

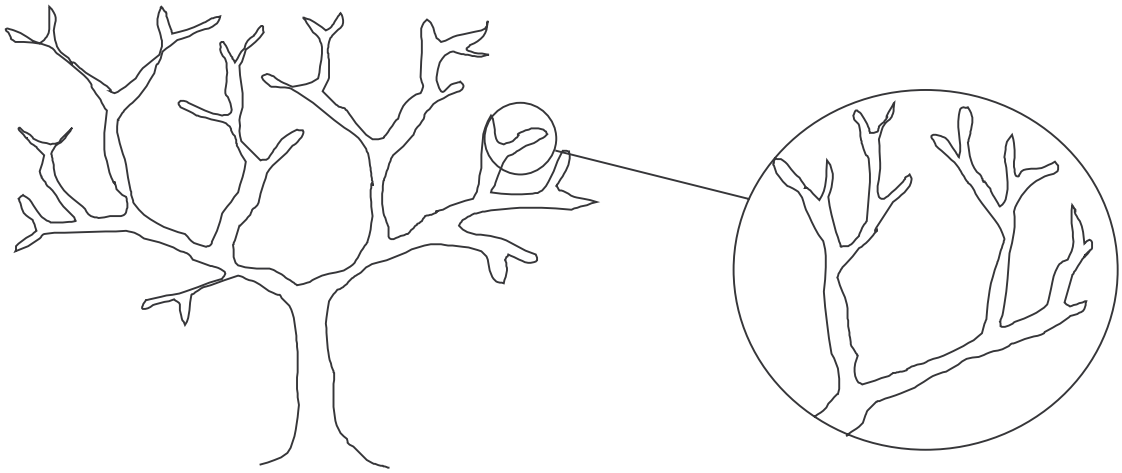

Figure 3.12: Self-similar characteristic – similar pattern found at different magnification.

The pattern of the Ethernet traffic found in [79] is "bursty" in nature. That is there is an active period of packets arrivals (a burst) followed by an inactive period with no packet arrivals. The Ethernet traffic becomes more "bursty" rather than smoother as the traffic becomes heavier or more users came onto the network. If a plot of packets/sec against time is plotted, the Ethernet traffic will exhibit self-similar characteristics i.e., it will have a similar bursty pattern when viewed using different time scales. On the other hand, the traffic for voice calls using the Poisson model does not exhibit a self-similar characteristic.

Measurements performed at the source level or source-destination pair (e.g. a user's workstation or router to workstation respectively) show that the traffic follows an alternating ON/OFF source where in the ON-period there is packet arrival and in the OFF-period there is no packet arrival [80]. The ON-period and OFF-period exhibit the Noah Effect, where the distribution has a high-variability or infinite variance. The

superposition of many independent and identically distributed ON/OFF sources produces the aggregated self-similar traffic observed in the measurements in [79].

The Pareto distribution is suggested for use in the ON and OFF periods in [79] and [61]. The Pareto probability density function is given as:

$$\boldsymbol{P}_{pareto}(t \mid t > \beta) = \frac{\alpha\beta^{\alpha}}{t^{\alpha+1}} \tag{3.3}$$

where the mean function is:

$$\boldsymbol{E}_{pareto}[t] = \frac{\alpha\beta}{\alpha - 1} \tag{3.4}$$

and the variance is:

$$\boldsymbol{Var}_{pareto}[t] = \frac{\alpha\beta^{2}}{(\alpha-1)^{2}(\alpha-2)} \tag{3.5}$$

The Pareto distribution exhibits the Noah Effect of high variability or infinite variance if $1 < \alpha < 2$. The values of α measured in typical source to host traffic in [80] are 1.7 for the ON-period and 1.2 for the OFF-period. The value of β for the ON-period is dependent upon the data rate and the average file size [61]. The average file size for web browsing application is 6.4 kbytes (for .html files) and 13.9 kbytes (for images) [81]. In this dissertation, time unit used is a normalised second (nmsec). Here, the time it takes to transmit one UCELLR (456 bits) is one nmsec therefore the data rate is hence 456 bits/nmsec. Using this rate and the larger average file size (13.9 kbytes) the average ON-period is about 243.9 nmsec, which corresponds to a β of 100.4 nmsec. In [80] the measured average ON-period and OFF-period of a typical source are 7.2 seconds and 10.5 seconds respectively. Using the same ratio (7.2 to 10.5) the corresponding average OFF-period is 355.6 nmsec, which corresponds to a β of 59.27 nmsec. During the ON-period, the packet arrival rate ranges from 0.4 to 0.8 packets/nmsec for the AP and 0.4 to 0.6 packets/nmsec for the SU. This gives an

average burst size of between 6.4 kbytes and 13.9 kbytes during the ON-period, and also reduces the number of packets transferred in OPNET.

3.2.5 Channel Measurement

In a measurement based DCA, the interference power of a channel is usually measured. As described previously, the autocorrelation of the measured interference power and the interference experienced by the packet is dependent upon the setup delay and the traffic model. The autocorrelation is lower in bursty traffic compared to voice traffic using Poisson arrival [59]. Hence, measurement may need to be performed very frequently to give an accurate picture of the interference environment. In [61] measurement is performed by the subscriber when it has transmitted all the packets in its queue, while in [59] it is performed after every ON-period. If measurements are performed too often, there will be less time for actual packet transmission and this will lower the packet throughput. The frequency with which channel measurement is performed is explored in later chapters in this dissertation.

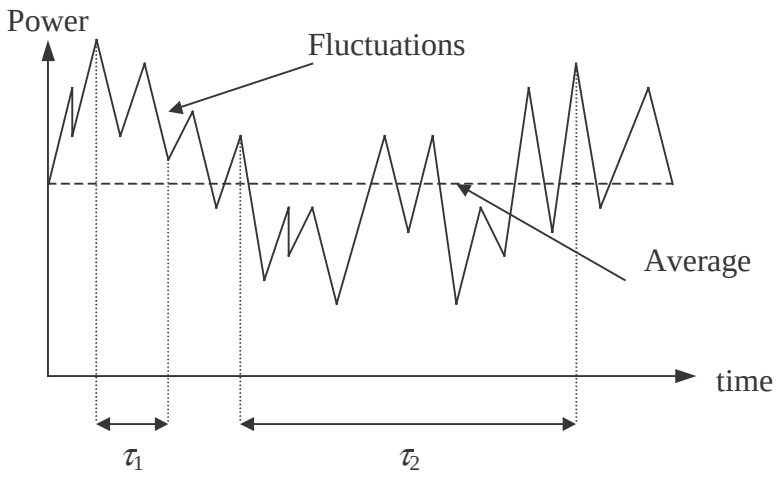

Figure 3.13: Fluctuations in measured power.

The duration of the measurement performed on each channel needs to be kept as short as possible in order to reduce setup delay (and hence improve the autocorrelation) and increase the packet throughput. However, due to multi-path fading and/or shadowing, fluctuations occur in the power measurement and if the measurement is performed too quickly, it may not give an accurate indication of the interference environment. For example, in Figure 3.13 if the measurement is performed within τ_1 as shown, the

average over this interval will over estimate the interference power. However, if the measurement is performed within τ_2 it will give a value closer to the average.

It is difficult to implement interference power measurement in OPNET. The Radio Pipeline Stages in OPNET Modeler are not designed to easily measure interference power. In OPNET, a receiver will continue to receive a packet even if the receiver's frequency changes during the packet reception because this packet has already passed the fourth stage (channel match) of the Radio Pipeline. For example, a receiver is set to Channel 1 and while a receiver is receiving a Valid Packet, a foreign packet (at Channel 1) that is longer than the Valid Packet arrives at the receiver and causes interference. After the first Valid Packet is received, the receiver changes to Channel 5 and begin receiving a second Valid Packet (at Channel 5). Theoretically, the foreign packet (on Channel 1) will not interfere with the second Valid Packet (on Channel 5). However, in OPNET, this foreign packet will continue to interfere with the second Valid Packet since the foreign packet has already passed the Channel Match (the fourth stage of the Radio Pipeline). Similarly, other foreign packets (on Channel 5) will not interfere with the second Valid Packet. To reduce this effect, the measurement needs to be longer than the longest packet. In reality, an uplink burst may consist of more than one UCELL and hence the resulting measurement time would be far too long causing a very low packet throughput. Alternatively, the length of the measurement can be fixed, while the packet size is limited (by breaking up a bigger packet into smaller ones) to be less than or equal to the measurement length. For example, if the measurement length can be fixed at 0.5 nmsec, the UCELL, which is 1 nmsec can be broken down into two packets each of 0.5 nmsec in length. This will increase the number of packets needed to be transmitted and will significantly lengthen the simulation time since each packet needs to pass through the 14 Pipeline Stages several times. Consequently, the measurement is set at one nmsec, which is the length of one data packet.

While performing a measurement, the AP is in a listening mode, where no packet is transmitted and the receiver is left to receive foreign packets where the power of these packets will be the interference power. In our model, the CDMA code in OPNET is used to distinguish Valid Packets from foreign packets (to reduce $N_{Tx\text{-}Rx}$ processing as described previously), the receiver will not be able to take in any foreign packet

unless there is a Valid Packet being received. However, the code can be changed during measurement (e.g. to a universal code) so that the receiver can process any foreign packets. Unfortunately, during the final channel measurement, if a valid wanted packet arrives at the end of the measurement, the receiver will continue to process the packet with the universal code (residue packet) and hence a true Valid Packet will be treated as noise (as the receiver can only receive one packet at a time).

To overcome this problem, we introduce a separate transmitter in each model AP solely for measurement purposes. The "imaginary" transmitter only transmits to the receiver of the AP containing this "imaginary" transmitter. The packet transmitted by the imaginary transmitter will not contribute any interference to other receivers and will not cause any delays in the simulation. The "measuring" packet size (packet transmitted by the "imaginary" transmitter) is made so that it occupies exactly one nmsec and the channel is set during the simulation to the destined receiver channel. Hence, the residue packet is eliminated and the interference power can be extracted directly from the "measuring" packet.

3.2.6 Cell Size

The interference power received by an AP is dependent upon the distance of the interferer. In a network with a regular cell size, the larger the cell radius, the further away the interferers are from an AP. If the transmit power is fixed and for cells in excess of a particular size, the receiver's noise will dominate the denominator part of the SNR. At this point, any improvements using DCA will be negligible.

A simulation with 50 APs and 277 SUs randomly distributed over 7 cells with a layout shown in Figure 3.14 is performed. All the APs uses the Random Channel Allocation (RND) with 15 channels and they transmit at fixed power. The cell size is varied from 250 m to 3000 m. The measurements are taken from the central cell shown shaded in Figure 3.14.

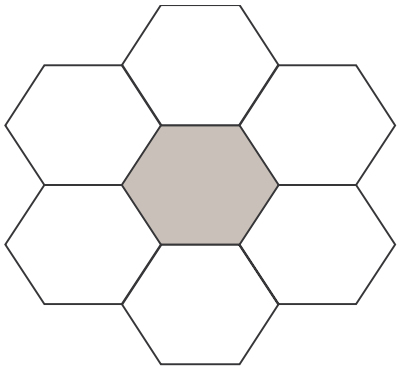

Figure 3.14: 7 cell layout for simulation.

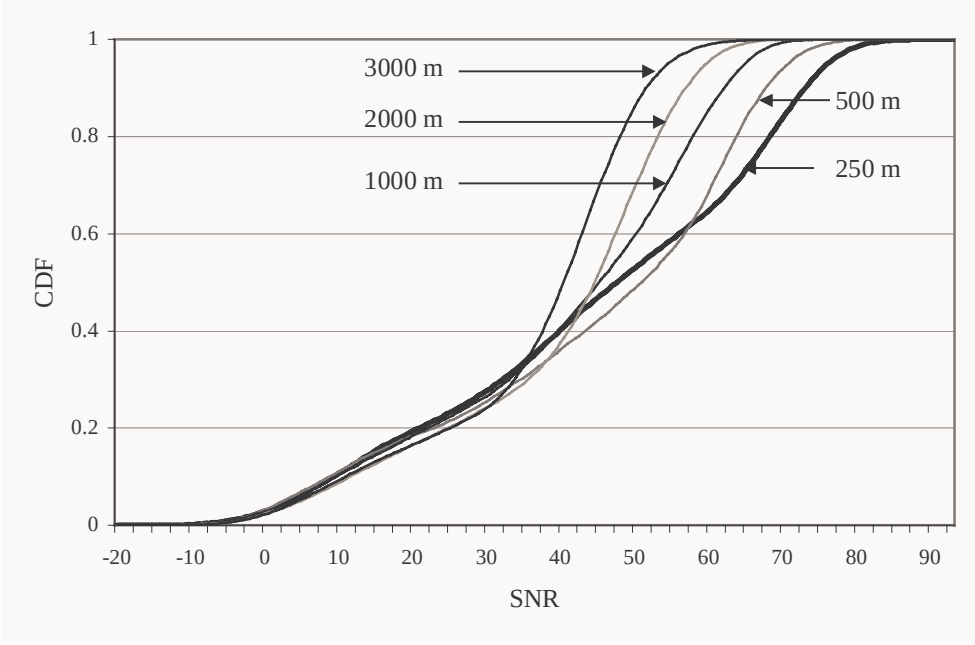

Figure 3.15: CDF - SNR performance for RND.

Figure 3.15 is the Cumulative Distribution Function (CDF) of the SNR performance. The larger cell size (e.g. 3000 m) has a narrower range of SNR values compared with that for a smaller cell size (e.g. 250 m). The SNR performance of a small cell size is interference limited. The SIR performance may be used instead but this will give a very large SIR range. It is also possible that no interference will be experienced at an AP over some short time intervals (since the traffic is of a bursty nature).

As the demand increases for broadband wireless access, microcells (i.e. cells with a radius of a few hundred metres) will be required. Consequently, microcells are employed in this dissertation, where specifically a cell radius of 500 m is used. A fixed transmit power of 30 dBm is also used for both the AP and SU. Using this cell

size, the SNR performance is interference limited and the performance of the various DCA schemes can be easily observed.

3.3 Area of Focus

The BFWA system under consideration operates in an unlicensed spectrum using TDD with asynchronous APs. The BFWA system is relatively simple since no synchronisation is required and simultaneous transmission and reception is not required. However, this BFWA is the hardest type of system in which to allocate channels. Most of the publications explored in Chapter 2 deal with voice services and none deal with this particular type of system.

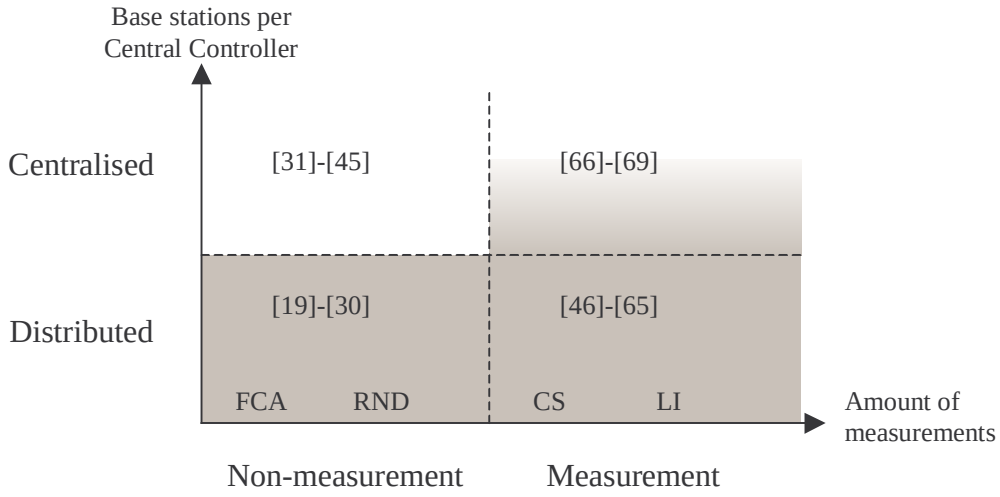

Figure 3.16: Channel Allocation Matrix – area of focus shaded in grey.

The DCA methods in the Centralised-Non Measurement quadrant of the Channel Allocation Matrix are not suitable for this BFWA system since it operates in an unlicensed band. A fully centralised system may not be scalable and is only practical in a small network. The area of focus in this dissertation are the lower two quadrants (Distributed DCA) of the Channel Allocation Matrix and a small part of the Centralised-Measurement quadrant. This relevant area is shown shaded in Figure 3.16.

4 DISTRIBUTED DCA BENCHMARKS

There are relatively few channel allocation methods designed specifically for a packet switched network. In this chapter, several channel allocation methods are implemented and applied to the BFWA system described previously.

As highlighted previously in the Channel Allocation Matrix, there are different methods of acquiring information regarding the interference environment. The information can be determined via current measurements or from a priori knowledge of the network. If we assume that each channel allocation method uses the information with the same efficiency, the performance (in terms of SNR and data throughput) of a channel allocation method is proportional to the amount of information available to the method [82].

The channel allocation methods to be implemented and applied to the BFWA system in this chapter are the Random Channel Allocation (RND), the Least Interfered (LI) method and a proposed FCA using a Genetic Algorithm (FCA-GA) [82]. The RND method uses the least information regarding the network and it is expected to have the worst performance, while the FCA-GA under the assumption of a static interference environment will have the most information concerning the entire network and so is expected to give the best performance.

4.1 Random Channel Allocation (RND)

In Random Channel Allocation, each channel is chosen at random from a uniform distribution. Each user has the same probability of using a good or bad channel and each channel will only be used for a short duration of time. Thus, in the long term, the interference experienced by each user is averaged out. This method is very simple to implement since it does not use a priori information nor obtain current information of the network during operation. The RND method is expected to give the worst performance and acts as a benchmark for the other channel allocation methods. This means that any proposed DCA that performs worse than RND should not be considered for use unless it is simpler to implement than RND.

For the BFWA system under consideration, the AP using the RND method will change its channel randomly at the start of every MAC frame. Since the size of each MAC frame is dependent on the traffic load (and hence is variable), the channel used by each AP will vary randomly. Each channel is selected randomly based on a uniform distribution and the AP sends this information to its SUs in the Header at the beginning of each MAC frame (i.e. the Header is sent using the same channel as the body of the previous MAC frame) so that the SUs will tune to the new channel. If a SU does not receive this information, it will lose its connection with the AP. The SU will need to re-acquire its connection with the AP perhaps using the broadcast channel or listen to all channels similar to that used in the Cellular Digital Packet Data (CDPD) system described in [55]. In this dissertation, it is assumed that the channel selected by the AP is correctly received by the SU.

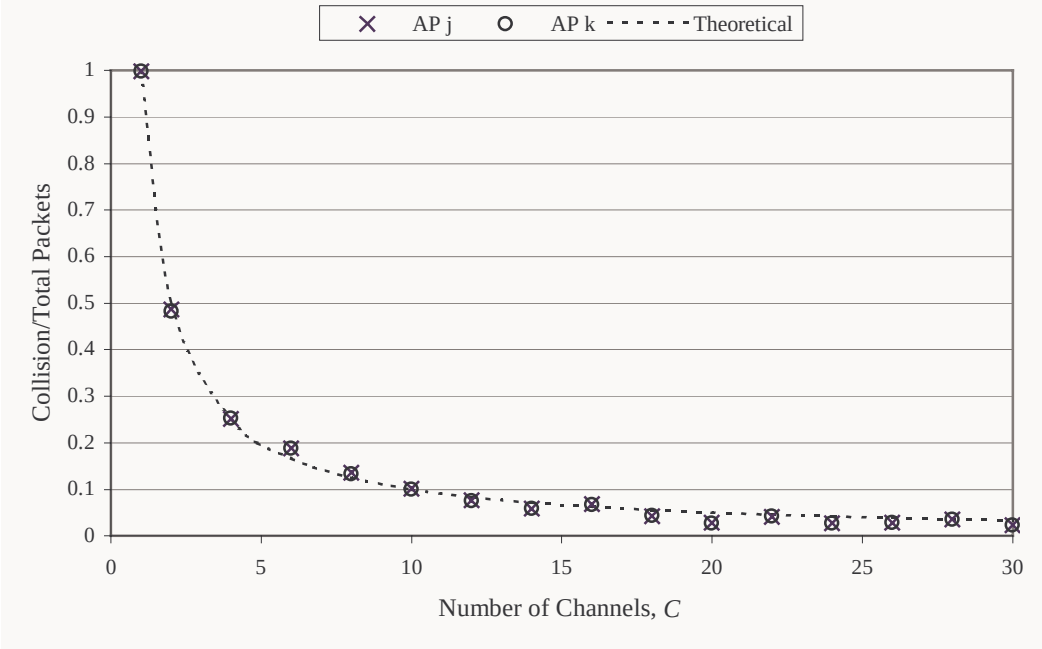

Figure 4.1: Probability of collision as a function of the number of channels, C for RND.

For the BFWA system under consideration, let C represents the total number of available channels. For two APs, AP j and AP k, the probability of AP k interfering with AP j's transmission and reception periods is plotted in Figure 4.1, where the probability of interference is given as the number of collisions over the total number of packets transmitted and received. In this scenario, there are only two APs, each having five SUs. A collision happens whenever a packet in AP j and any of its five SUs (belong to AP j) is interfered with AP k or any of AP k's SUs (i.e. when both APs

use the same channel). The probability of interference at AP k is obtained using the same method. The probability of interference (i.e. total packet collisions/total packets) for AP j and AP k are almost the same. Theoretically, at AP j, AP k has a probability of $1/C$ of interfering with AP j. Hence, the probability P_{I_RND} of an AP being interfered with another AP using RND is:

$$P_{I\_RND} = \frac{1}{C} \tag{4.1}$$

The theoretical value (4.1) is plotted in Figure 4.1 and it shows that the simulated results closely follow the theoretical value. For an AP, each of the other APs in the network has a probability of $1/C$ of interfering with it and the magnitude of the interference power is proportional to the position of the interferer. Therefore, the higher the number of channels the more effective RND becomes, however it is clear from Figure 4.1 that the effect of increasing C saturates when $C > 15$. For the same network layout, reducing P_{I_RND} will reduce the amount of interference faced by an AP.

4.2 Least Interfered (LI) Method

As described in Chapter 2, the Least Interfered (LI) method is the most popular DCA method in the Distributed-Measurement quadrant. This method measures the interference power of all the available channels and selects the one with the lowest interference power. However, this method is mainly used in circuit switched voice networks. For the BFWA system under consideration, a SCAN portion is introduced into the MAC frame as shown in Figure 4.2.

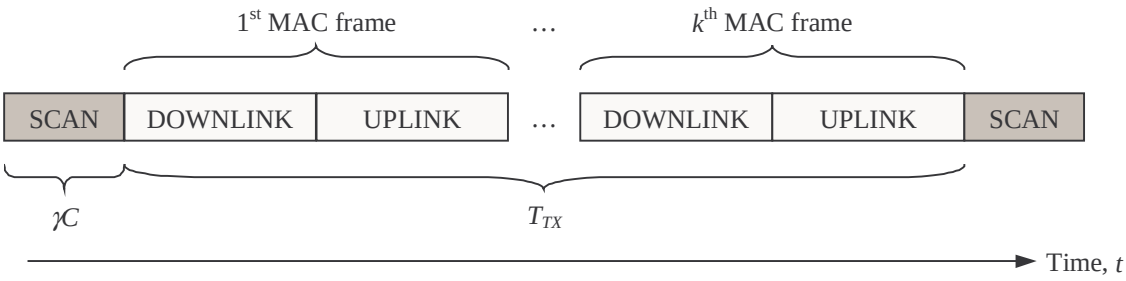

Figure 4.2: MAC frame with SCAN portion.

The measurement is performed by the AP since it is in a better position (i.e., higher) to perform the measurement. In the SCAN portion, the AP measures the interference power of all the C channels, where each channel requires γ nmsec to measure. The AP selects the channel with the lowest interference power and this channel is used for the next k MAC frames. The time between two SCANs is defined as the transmit portion T_{TX}, which is shown in Figure 4.2. If the AP performs measurements very frequently, the value of T_{TX} will be small and the packet throughput T_{MAC} of the system will be low since more time is spent on measuring channels rather than transmitting information. However, if the traffic load is low, a small value of T_{TX} may not affect T_{MAC} too much since most of the time, the AP will be broadcasting the header rather than sending DCELLs or receiving UCELLs. Since packet throughput T_{MAC} is dependent upon the traffic load, the data throughput G_{MAC} is used as the performance measurement of a system. The data throughput G_{MAC} is defined as the amount of time spent on data transmission per nmsec and it is expressed as:

$$G_{MAC} = \frac{T_{TX}}{\gamma C + T_{TX}} \tag{4.2}$$

Therefore, if an AP spent more time on measurements, it will have a lower value of G_{MAC}. The data throughput can be converted to bits per second. For example if the AP and SU in the system can transmit at 25 Mbps, and G_{MAC} = 0.8 (i.e. 20% of the time is spent on channel measurement), the overall bits per second is 20 Mbps. A 1% difference in data throughput is equivalent to 250 kbps in data rate. For a system that does not perform channel measurement (e.g. RND) G_{MAC} = 1. Hence a higher transmit portion T_{TX} leads to higher data throughput.

On the other hand, if T_{TX} is large, the network will take a longer time to react to any interference changes. To demonstrate this, a BFWA scenario utilising the LI method with 12 APs, each having 5 SUs is simulated. The total number of channels C is 15. At the beginning of the simulation, all the APs use the same channel and the network converges to an ideal channel assignment when each AP uses a different channel. Figure 4.3 is a plot of convergence time in nmsec as a function of T_{TX}/M_{MAC}, where M_{MAC} = 66.61 nmsec is the maximum MAC frame size. For each T_{TX}/M_{MAC} value, the simulation is run four times using different seed values and the average convergence

time is used. It is difficult to achieve an exact and constant value of T_{TX} since the size of the MAC frame is highly dependent upon the traffic load. Hence, an approximate value of T_{TX} is used. Each AP starts transmitting its first packet after a random delay so that the APs are asynchronous. It is shown in Figure 4.3 that the convergence time increases exponentially as T_{TX} increases.

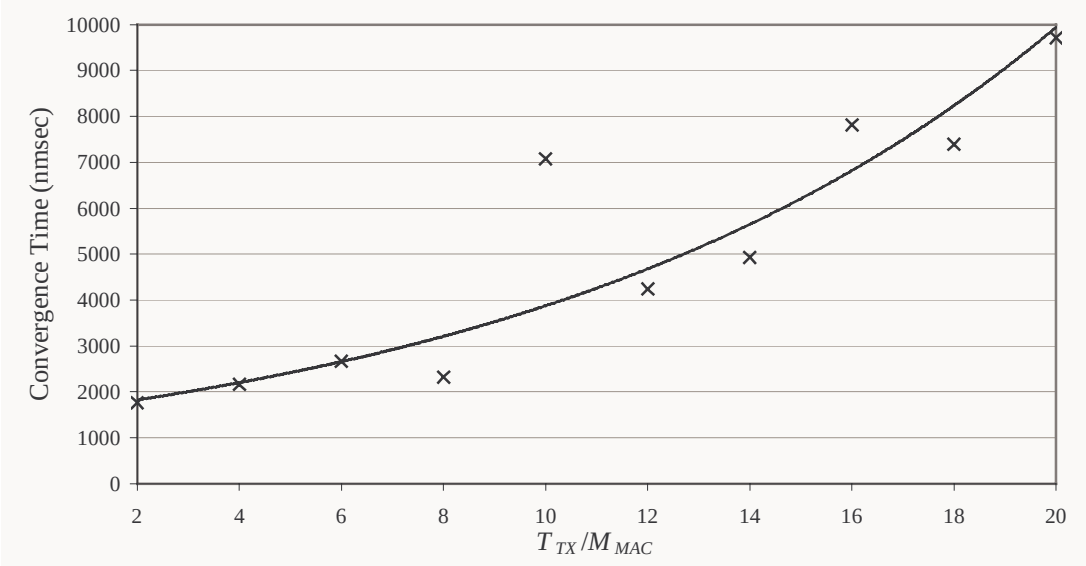

Figure 4.3: LI convergence time against T_{TX}/M_{MAC}.

The probability, P_{I_LI} that an AP is being interfered by another AP is dependent upon the measurements performed during the SCAN portion of the MAC. Let the time taken to measure one channel be γ, thereby giving a total SCAN time of γC as shown in Figure 4.2. For two APs, AP *j* and AP *k*, let the transmit periods of AP *j* and AP *k* be T_{TX_j} and T_{TX_k} respectively. The probability $D_{j,k}$ of AP *j* detecting the channel usage of AP *k* when AP *j* is scanning is:

$$D_{j,k} = \frac{T_{TX\_k}}{\gamma C + T_{TX\_k}} \tag{4.3}$$

Figure 4.4 shows a plot of $D_{j,k}$ against T_{TX_k} using (4.3). The simulated results using various values of T_{TX_j} (1, 5, 10 and 20) are also plotted in Figure 4.4. The simulation values of $D_{j,k}$ for the plotted combinations of T_{TX_k}/M_{MAC} and T_{TX_j}/M_{MAC} are obtained by dividing the number of successful detections by the number of channel scans performed. It can be observed that the curve is smoother when T_{TX_j}/M_{MAC} = 1

compared to that when $T_{TX\_j}/M_{MAC} = 20$. This is because the simulation is run for the same amount of time and at $T_{TX\_j}/M_{MAC} = 20$, less channel scans will be performed by AP j. Since the values of T_{TX_k} and T_{TX_j} are difficult to maintain at a constant level at all times, there is some deviation between the theoretical and simulated results.

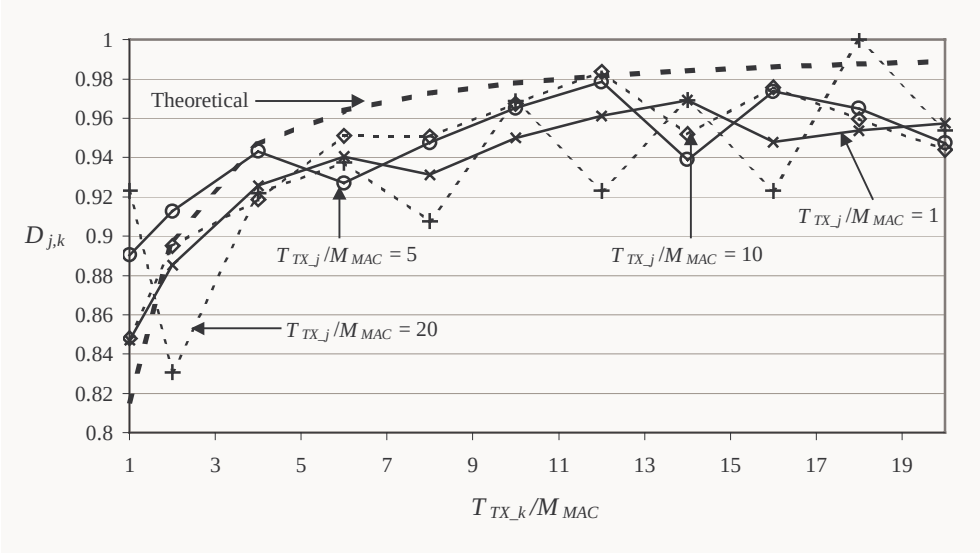

Figure 4.4: Probability of detection $D_{j,k}$ against T_{TX_k} for various T_{TX_j}.

If AP j detects the channel usage of AP k, AP j will avoid using the same channel as AP k (if that is not the lowest interfered channel). Even if AP j fails to detect the channel usage of AP k, they can also avoid using the same channel if AP k detects the channel usage of AP j. Therefore, as long as one of the APs successfully detects the channel usage of the other, the effect is the same. The probability D_{LI} that one or both APs detect the channel usage of each other using LI is given by:

$$D_{LI} = \left(\frac{\gamma C}{\gamma C + T_{TX\_j}}\right)\left(\frac{T_{TX\_k}}{\gamma C + T_{TX\_k}}\right) + \left(\frac{\gamma C}{\gamma C + T_{TX\_k}}\right)\left(\frac{T_{TX\_j}}{\gamma C + T_{TX\_j}}\right) \qquad (4.4)$$

The first term of (4.4) is the probability that AP j is listening while AP k is transmitting and the second term is the probability that AP k is listening while AP j is transmitting. If channel detection is successful, AP k will interfere with AP j only if both are using the channel with the lowest interference power. AP k has a probability of $1/C$ of using any one of the C channels, and it is assumed that there is a probability of $1/C$ that that channel has the lowest interference power. Hence, AP k and AP j will have a probability of $1/C^2$ of using the same channel, which is also the lowest

interfered channel. Note that this is a simplified expression of the probability since the probability of using the lowest interfered channel is influenced by the network layout and also the channel usage of all the APs in the network. On the other hand, if the channel detection fails, then the probability of AP *k* and AP *j* using the same channel is equivalent to that in RND. The probability P_{C_LI} of AP *k* and AP *j* using the same channel is given as:

$$P_{C\_LI} = \frac{1 - D_{LI}}{C} + \frac{D_{LI}}{C^2} \tag{4.5}$$

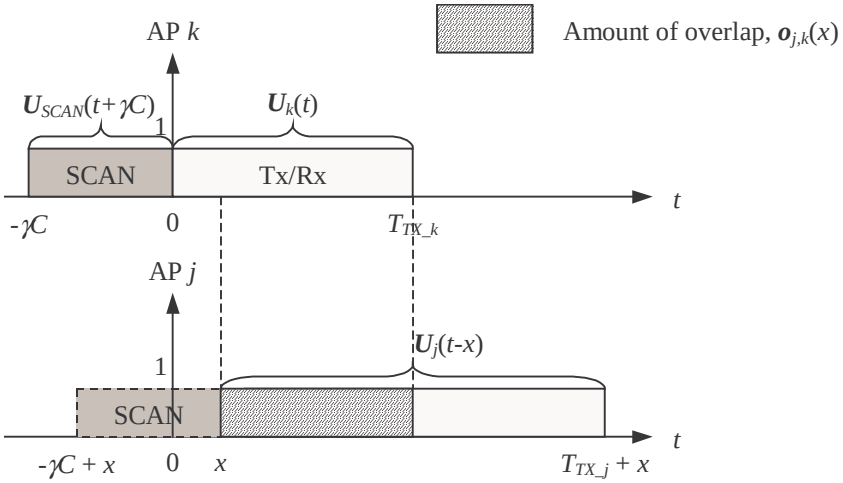

Figure 4.5: MAC frame modelled as functions $\boldsymbol{U}_j(t)$, $\boldsymbol{U}_k(t)$ and $\boldsymbol{U}_{SCAN}(t)$.

Although AP *j* and AP *k* may use the same channel, AP *k* will only interfere with AP *j* if the transmit portion of AP *j* (i.e. T_{TX_j}) coincides with that of AP *k* (i.e. T_{TX_k}). To find the average amount of overlap, the transmit portion of the MAC frame for AP *j* and AP *k* are modelled as functions $\boldsymbol{U}_j(t)$ and $\boldsymbol{U}_k(t)$ respectively as shown in Figure 4.5. The function $\boldsymbol{U}_j(t)$ is given as:

$$\boldsymbol{U}_j(t) = \boldsymbol{f}_u(t) - \boldsymbol{f}_u(t - T_{TX\_j}) \tag{4.6}$$

where $\boldsymbol{f}_u(t)$ is the unit step function. Function $\boldsymbol{U}_k(t)$ can be defined in a similar manner. Similarly the SCAN portion is $\boldsymbol{U}_{\mathrm{SCAN}}(t)$ is given by (which is the same for all APs in LI),

$$\boldsymbol{U}_{SCAN}(t) = \boldsymbol{f}_u(t) - \boldsymbol{f}_u(t - \gamma C) \tag{4.7}$$

As shown in Figure 4.5, the amount of overlap $\boldsymbol{o}_{j,k}(x)$ at AP j due to AP k is dependent upon the offset x. Consider just a pair of MAC frames and assume the distribution of x is uniform, $\boldsymbol{o}_{j,k}(x)$ can be found by convolving $\boldsymbol{U}_j(t)$ and $\boldsymbol{U}_k(t)$, that is:

$$\boldsymbol{o}_{j,k}(x) = \int_{-\infty}^{x} \boldsymbol{U}_j(t-x)\boldsymbol{U}_k(t)dx \tag{4.8}$$

The function $\boldsymbol{o}_{j,k}(x)$ can be represented graphically as a "trapezium" as shown in Figure 4.6.

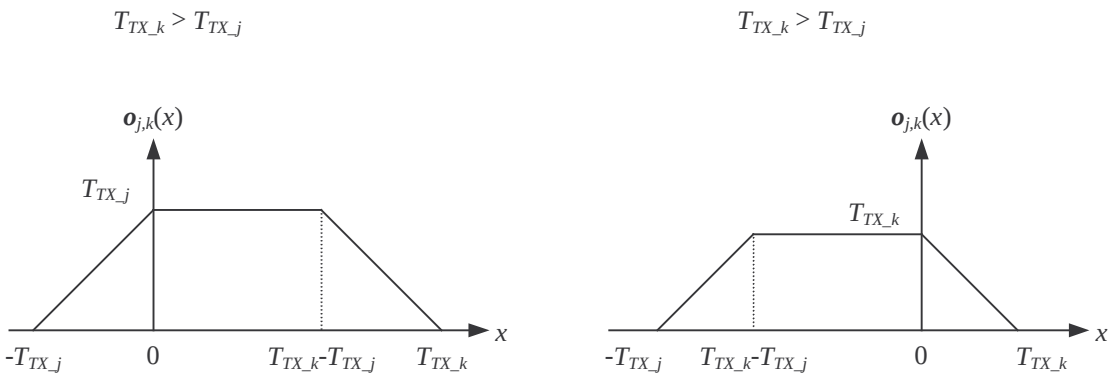

Figure 4.6: Graphical representation of $\boldsymbol{o}_{j,k}(x)$ for $T_{TX\_k} > T_{TX\_j}$ (left) and $T_{TX\_j} > T_{TX\_k}$ (right).

The MAC frames for each AP are continuous as shown in Figure 4.7 and therefore the amount of overlap includes the previous MAC frames.

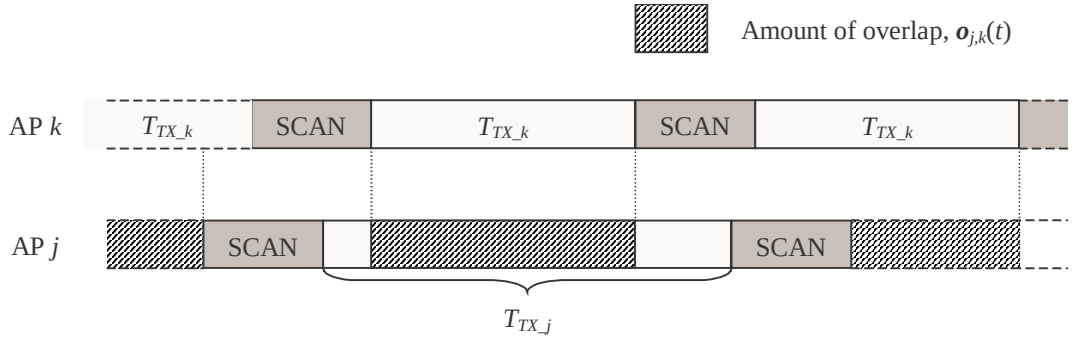

Figure 4.7: Amount of overlap $\boldsymbol{o}_{j,k}(t)$ considering previous MAC frames.

Taking into account the previous MAC frames the effective $\boldsymbol{o}_{j,k}(x)$ is represented as the summation of several "trapeziums" as shown in Figure 4.8 where each "trapezium" represents the amount of overlap for two single MAC frames (one for AP j and AP k).

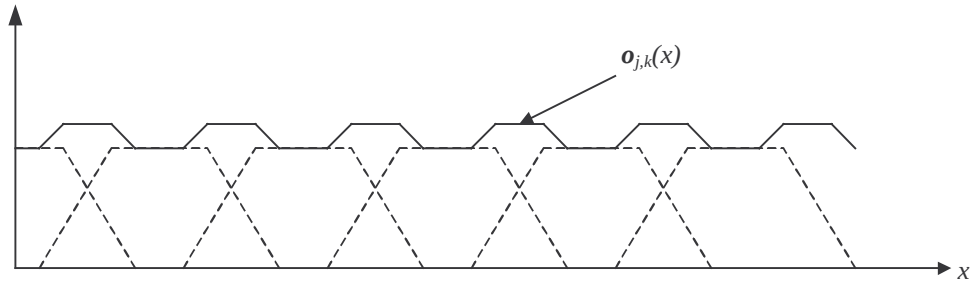

Figure 4.8: Effective $\boldsymbol{o}_{j,k}(x)$ taking previous MAC frames into consideration.

The average amount of overlap O_{AVG_j} for AP j due to AP k taking into account previous MAC frames is thus:

$$O_{AVG\_j} = \frac{T_{TX\_j} T_{TX\_k}}{\gamma C + T_{TX\_k}} \tag{4.9}$$

The average fraction $A_{j,k}$ of AP j's MAC frame that will overlap with AP k's transmit portion T_{TX_k} and causes interference if both APs use the same channel is thus:

$$A_{j,k} = \frac{O_{AVG\_j}}{\gamma C + T_{TX\_j}} = \frac{T_{TX\_j} T_{TX\_k}}{\left(\gamma C + T_{TX\_j}\right)\left(\gamma C + T_{TX\_k}\right)} \tag{4.10}$$

The value $A_{j,k}$ is effectively the probability that both APs are transmitting. Hence, the probability P_{I_LI} of AP k interfering AP j using LI is:

$$P_{I\_LI} = \left(\frac{1 - D_{LI}}{C} + \frac{D_{LI}}{C^2}\right) A_{j,k} \tag{4.11}$$

In a two AP scenario, the probability of interference in LI (P_{I_LI}) and RND (P_{I_RND}) are plotted in Figure 4.9. These curves demonstrate that using measured information, the probability of interference can be reduced.

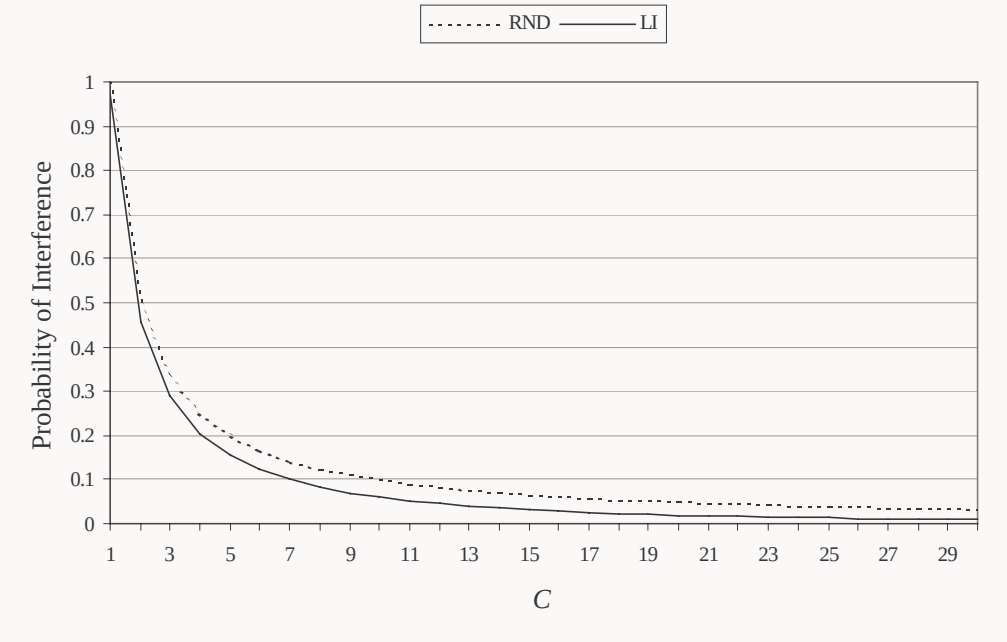

Figure 4.9: Probability of Interference as a function of C, where T_{TX}=1 in LI.

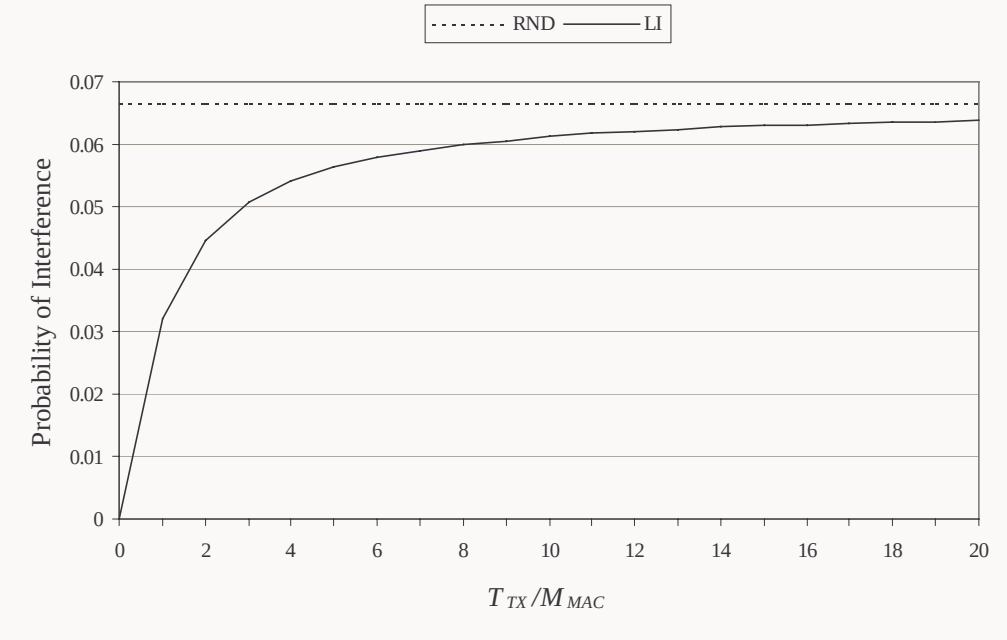

Figure 4.10: Probability of interference as a function of T_{TX}/M_{MAC} for C=15.

In LI, the transmit portions in AP *j* and AP *k* are set to the same value, that is $T_{TX\_j} = T_{TX\_k} = T_{TX}$. The probability of interference P_{I_LI} shown in Figure 4.10 is seen to increase as T_{TX}/M_{MAC} increases and at large values of T_{TX}/M_{MAC}, P_{I_LI} approaches P_{I_RND} since channel measurement is performed less frequently. Therefore, increasing

T_{TX} increases the data throughput G_{MAC} but a high T_{TX} will cause the system to react slowly to changes and result in a high level of interference. The value T_{TX} needs to be selected such that it gives a good balance between these factors.

4.3 FCA Using Genetic Algorithm

In this section, the channels are allocated to each AP using FCA. FCA uses only a priori information of the network configuration and does not utilise any current interference information. Consequently, it is unable to adapt to interference changes, which is critical particularly in this BFWA network, which operates in unlicensed spectrum. However, if we assume the interference environment is static (i.e. no additions to or changes of nodes and no foreign interferers), the a priori information used by FCA will not change. Thus, FCA will have global knowledge of the entire network and it will perform better than DCA methods that do not have access to the global network information. Under this condition, FCA will act as the upper (performance) benchmark for the DCA methods, while RND will act as the lower (performance) benchmark.

As described in Chapter 3, user satisfaction is proportional to the SNR experienced by the SU and hence the aim of channel allocation in this BFWA application is to try to maximise the overall SNR of all the SUs and APs in the network. For a fixed transmit power, the SNR can be increased by reducing interference. The interference power decreases when the distance from the interferer increases. If a sectored or directional antenna is used, the position of the interferer affects the amount of interference power that will be experienced by a user. Hence, by placing the co-channel base stations in strategic positions (e.g. such that the front lobe of the antennas do not face each other) and by increasing their separation distances the overall interference power will be reduced. However, this can be a large combinational problem and the size of the search space increases exponentially with the size of the problem in a similar manner to a NP-complete problem. Therefore, the optimum solution is difficult to compute in a large network. In a small network, the chance of finding a good sub-optimal solution is much higher compared to finding one in a large network (however it is difficult to determine whether a sub-optimal solution is the optimum solution).

As highlighted in Chapter 2, there are several intelligent search algorithms that are capable of escaping from a bad local optimum solution. Although each technique has

its group of supporters and each method claims (with modification) to be more effective than the other, the Genetic Algorithm is selected in this section. Firstly, the Genetic Algorithm is able to search different distinct points in the overall search space compared to Simulated Annealing and the Tabu Search (which search only in the vicinity of a solution). Secondly, the Genetic Algorithm has two methods (namely selection and mutation) to escape from a local optimum point compared to only one method of escape in Simulated Annealing and the Tabu Search.

4.3.1 Genetic Algorithm

A Genetic Algorithm is an iterative process that tries to find the optimal solution in a search space **S**. As described in Chapter 2, a Genetic Algorithm consists of a population $\mathbf{P}(t) \in \mathbf{S}$, where $\mathbf{P}(t)$ is a set of possible solutions at the t^{th} iteration (i.e. $\mathbf{P}(t) = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{|\mathbf{P}(t)|}\}$. The size of the population is $|\mathbf{P}(t)|$ and it is fixed in the proposed FCA scheme. Each solution **s** in the entire search space **S** is an individual and a fitness function $\boldsymbol{F}(\mathbf{s})$ is used to evaluate the gain (relative to other solutions) of individual **s**. Therefore, the fittest individual gives the optimum solution (there may be more than one individual with the maximum fitness value). The processes in the Genetic Algorithm consist of the Selection, Crossover and Mutation and these are shown in Figure 2.9 in Chapter 2.

The way Genetic Algorithm works can be explained using schema theory and the building block hypothesis [83]. A schema has fixed values at specific positions in a string and it represents a subset of strings that has these characteristics. For example for a string using binary encoding with a length of five elements, 10**1 is a schema that represents strings 10001, 10011, 10101 and 10111. The average fitness of a schema is the average fitness of all the strings that it represents. For a binary encoded string with length l, there are 2^l distinct schemata and these schemata compete against each other during the Genetic Algorithm processes. The optimal solution is assumed to be a concatenation of schemata with short lengths and high average fitness values. These schemata with short lengths and high average fitness values are called building blocks. If the optimal solution does not contain high fitness building blocks, the Genetic Algorithm may converge to only a local optimum solution.

In a population, the selection process favours building blocks that have high average fitness values and it increases the instances (i.e. strings that fall into this schema)

within the population. The crossover process will merge these high average fitness building blocks with the aim of forming the optimal solution. The mutation process produces radical new building blocks and is a way to escape from a local optimum solution. The Genetic Algorithm uses implicit parallelism, where it is able to process a large number of schemata using a small sample size. This is because, in a generation, the Genetic Algorithm by processing $K=|\mathbf{P}(t)|$ strings, implicitly evaluates approximately K^3 schemata [83].

4.3.2 Application of a Genetic Algorithm to FCA

In this section, a Genetic Algorithm is applied to the FCA problem. The objective of the FCA using Genetic Algorithm (FCA-GA) is to reduce interference by placing the channels such that the co-channel APs are at strategic locations (e.g. the main lobes of two co-channel APs' antenna do not face each other) and are as far as possible from each other. This approach is described in [82]. In common with other FCA methods, a priori information is required and in this case information concerning the number of APs in the network M_{AP}, the positions of these APs, their cell size, the antenna radiation patterns of the APs and SUs and the propagation model of the wireless channel are required.

Channel utilisation is defined as the number of APs using a particular channel. In general, if a channel is highly utilised, the APs using this channel will experience higher interference than the APs using a channel utilised less frequently. This is because, in a highly utilised channel, more APs transmit using this channel thereby raising the level of interference. To be fair to all APs, FCA-GA tries to achieve uniform channel utilisation across all channels. Hence, the number of APs is divided equally among the C channels in the system. This is represented in Figure 4.11 which shows a string where $C = 3$ and $M_{AP} = 9$. The elements in the string are the AP identities and in this example, 3 APs are assigned to each channel. In the example in Figure 4.11, AP 5, AP 2 and AP 8 are assigned to Channel 1, AP 3, AP 7 and AP 4 are assigned to Channel 2 and AP 1, AP 6 and AP 9 are assigned to Channel 3. The encoding used is non-binary and has a cardinality equals to the number of APs (i.e. the maximum value of an element is M_{AP}).

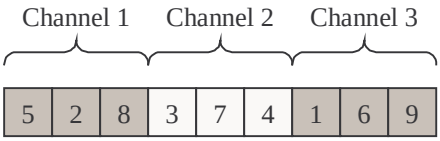

Figure 4.11: String representation of a channel assignment in FCA-GA.

Restricting the number of APs per channel reduces the total search space **S**, since FCA-GA does not consider non-uniform channel utilisation. Using non-binary encoding reduces the length of the string. Instead of using 3 codes (i.e. 101 in binary) to encode the value 5 (the AP identity), only one is used. This also avoids the Hamming cliffs problem reported in [83], where large Hamming distances are present between two adjacent integers. For example to get from 0111 to 1000, i.e. from 7 to 8, there is a Hamming distance of 4. Finally, it is easier to ensure a valid string (i.e. only one AP per channel) using integer encoding rather than binary encoding.

The fitness function $\boldsymbol{F}_{FCA}(t)$ is a measure of the overall interference experienced by each AP at the t^{th} iteration. It is given as:

$$\boldsymbol{F}_{FCA}(t) = \sum_{j=1}^{C} \boldsymbol{F}_j(t) \tag{4.12}$$

Where j is the channel number ranging from 1 to C and $\boldsymbol{F}_j(t)$ is the interference power experienced by APs and SUs using channel j. $\boldsymbol{F}_j(t)$ is given as:

$$\mathbf{F}_j(t) = \sum_{j \in \mathbf{M}_j(t)} \sum_{\substack{k \in \mathbf{M}_j(t) \\ k \neq j}} \left[\mathbf{P}_{Aj,Ak}(t) + \mathbf{P}_{Aj,Sk}(t) + \mathbf{P}_{Sj,Ak}(t) + \mathbf{P}_{Sj,Sk}(t) \right] \tag{4.13}$$

There are 4 ways interference can occur in a TDD system. The possible ways are AP to AP, AP to SU, SU to AP and SU to SU. Function $\boldsymbol{P}_{Aj,Ak}(t)$ in (4.13) is the received interference power at AP j from AP k's transmission at the t^{th} iteration. Function $\boldsymbol{P}_{Aj,Sk}(t)$ is the received interference power at AP j from the transmission of an SU belonging to AP k at the t^{th} iteration. Function $\boldsymbol{P}_{Sj,Ak}(t)$ is the received interference power at an SU belonging to AP j from AP k's transmission at the t^{th} iteration. Finally, function $\boldsymbol{P}_{Sj,Sk}(t)$ is the received interference power at an SU belonging to AP j from the transmission of an SU belonging to AP k at the t^{th} iteration. The MAC

structure defined in Chapter 3 allows only one SU to transmit at a time and we ignore the contribution due to the contention period since it is short compared to the transmission of data, i.e., UCELLs. To reduce the complexity in evaluating the fitness function, it is assumed that the interference contribution of SUs belonging to AP *k* are represented by one SU placed at the centre of the cell using the same sectored antenna as the AP. This is shown in Figure 4.12, where it is assumed that the overall antenna pattern due to the directional antennas of the SUs (that are placed randomly within the sector) is equivalent to a sectored antenna (similar to the one at the AP) placed at the centre of the cell facing the AP.

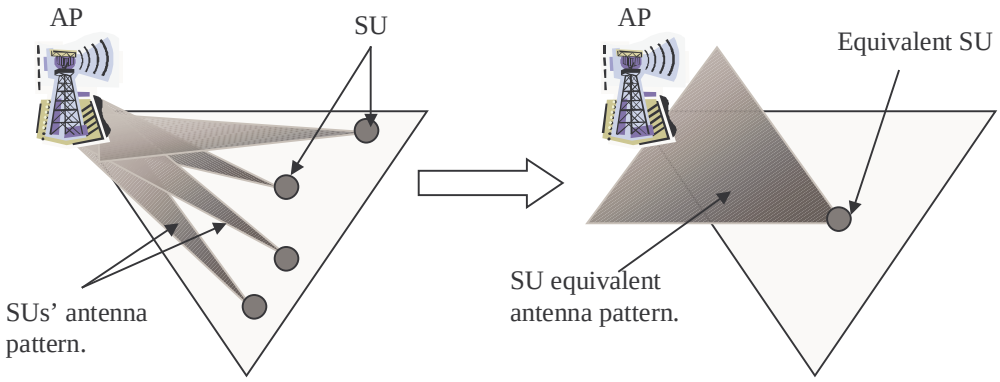

Figure 4.12: Equivalent SU antenna pattern.

$\mathbf{M}_j(t)$ is the set of APs using channel j and hence the received interference power $\boldsymbol{F}_j(t)$ is calculated by summation over all the APs and SUs using channel j. The overall fitness $\boldsymbol{F}_{FCA}(t)$ is the total received interference power by all APs and SUs in the network. This is a negative fitness function where the lower the fitness value the better is the solution. To avoid confusion, the word fitness for the rest of this section is replaced with the word cost, since a low cost value is generally considered "good."

The interference power is calculated based on the Random Height propagation model described in Chapter 3 excluding any shadowing effect to give a fixed cost (i.e. fitness) value to each channel assignment (since the shadow value is a random number). As well as having global knowledge of the entire network, the propagation model in the simulation will also give the FCA-GA partial (since shadow is not included in fitness calculation) knowledge of the interference environment thereby equipping it with sufficient information required for a channel allocation.

The crossover method employed by the Genetic Algorithm must ensure that it does not produce an invalid string (i.e. an AP can only appear once in the string) and the probability of crossover p_c is set to 1.0. A partially mapped crossover method (with modification) as employed for the Travelling Salesman Problem is used here [84]. In this arrangement when performing a crossover between two individuals, namely Individual 1 and Individual 2, a portion from Individual 1 is cut and placed into the identical position in Individual 2. An example is shown in Figure 4.13 for a string length of 10 elements, where the cut portion of Individual 1 is placed into the same section in Individual 2.

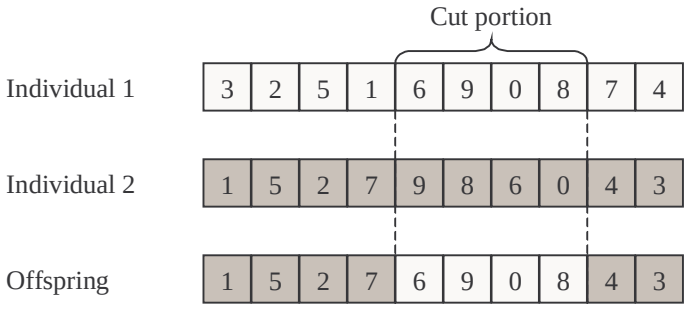

Figure 4.13: Crossover operation.

An invalid string occurs if one or more elements in the cut portion from Individual 1 have the same value as the section that is not being replaced in Individual 2. To see this, consider another two individuals, Individual A and Individual B as shown in Figure 4.14. If the cut portion of Individual A is directly placed into Individual B, the 5th and the 7th elements with values 6 and 0 respectively of the Offspring will conflict with its existing elements in positions 1 and 2.

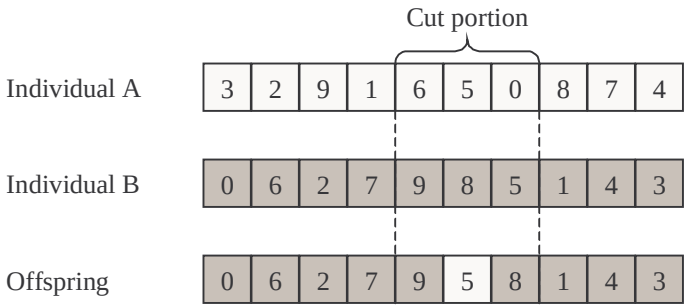

Figure 4.14: Partially mapped crossover that avoids having an invalid string.

In the partially mapped crossover, each element of the cut portion from Individual A is compared with the other portions of Individual B to ensure that there are no duplications. If duplications occur, the Offspring will select a value from an equal or different element from Individual B. For example, in Figure 4.14, the cut portion from Individual A consists of the 5^{th}, 6^{th} and 7^{th} elements with values 6, 5 and 0 respectively. The 5^{th} element of Individual A has the same value as the 2^{nd} element of Individual B. Instead of using the value from Individual A, the Offspring takes the value from Individual B. The 6^{th} element of Individual A can be placed directly into Offspring since it does not as yet cause any duplication. Finally, the 7^{th} element of Individual A will cause a conflict with the 1^{st} element of the Offspring if replaced directly. However, the Offspring cannot take the value from the 7^{th} element of Individual B (i.e. value 5) since it will cause a conflict with the 6^{th} element of the Offspring. In this case, the Offspring will take a value from the current element minus one (i.e. the 6^{th} element) of Individual B. If the 6^{th} element still causes a conflict, the Offspring will consider successively lower elements of Individual B until a valid one is found (within the cut portion) that does not cause any conflict. The value of the Offspring at the end of the crossover process is shown in Figure 4.14 where the elements that are shaded come from Individual B while the non-shaded elements come from Individual A.

The mutation process is performed by randomly selecting two elements in the string and swapping their values. Using this method, an invalid string will be avoided.

Three selection methods are considered namely the Roulette Wheel, Tournament and Elitism. Roulette Wheel selection has already been discussed in Chapter 2. In Tournament selection, two individuals are selected at random from the current population. The individual that has a higher cost will have a probability of 0.75 of being selected for crossover with another selected individual. In Elitism, the individuals that are the fittest fraction μ of the population are retained for the next generation while the remaining fraction (1-μ) of the population are selected for crossover using Tournament selection, (where individual are picked from the current entire population).

The choice of control parameters p_c, p_m and the population size is itself an optimisation problem. A network with 50 APs randomly distributed over 7 cells with

a layout the same as that in Figure 3.14 of Section 3.2.6 is used for the investigation of the control parameters. The transmit power of each AP is fixed at 30 dBm and the cell size is 500 m. In each selection method, various population sizes and mutation probabilities p_m are explored. The population size ranges from 10 to 5000 individuals while the values of p_m range from 0.01 to 0.5.

4.3.3 Population Size

The effect of population is investigated by fixing the mutation rate (p_m = 0.01) and changing the population size. The simulation is run for 10000 generations.

4.3.3.1 Average Lowest Cost, C_{ALC}

The lowest cost in each generation is recorded and the average lowest cost C_{ALC} is evaluated over the range from the 2000th generation until the 10000th generation. After 2000 generations, the system has converged in most cases.

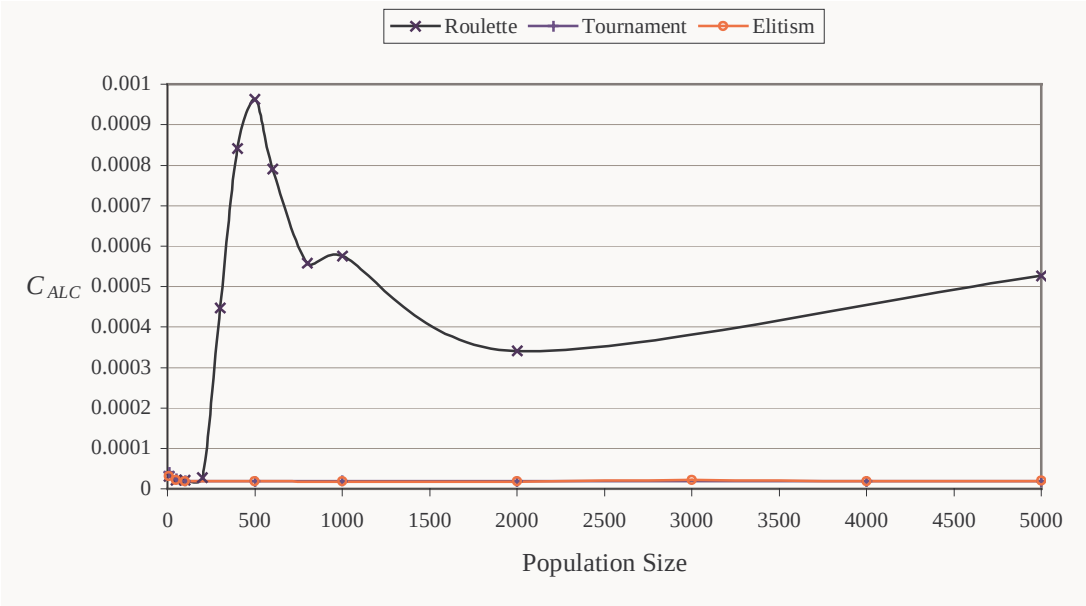

Figure 4.15: C_{ALC} as a function of population size.

Figure 4.15 plots C_{ALC} for the three different selection methods (with μ = 0.1 for Elitism). It shows that in Roulette Wheel selection, a population size in excess of 300 gives rise to significantly higher cost. This is because with a large population, the Roulette Wheel selection method finds it harder to choose the low cost individuals as the size of the sector corresponding to the low cost individuals in the Roulette Wheel becomes smaller. Consequently, the probability of a low cost individual being selected becomes low. Apart from this, with a large population size there are too

many sectors within the wheel making the probability of selecting each sector smaller. Figure 4.15 shows a maximum value of C_{ALC} when the population size is around 500 and that C_{ALC} remains fairly constant between population size of 1000 to 5000. A low cost solution is found when the population size is below 200.

Figure 4.15 is re-plotted in Figure 4.16, concentrating on a population size below 400. The results reinforce the view that a population size below 200 is appropriate for Roulette Wheel selection.

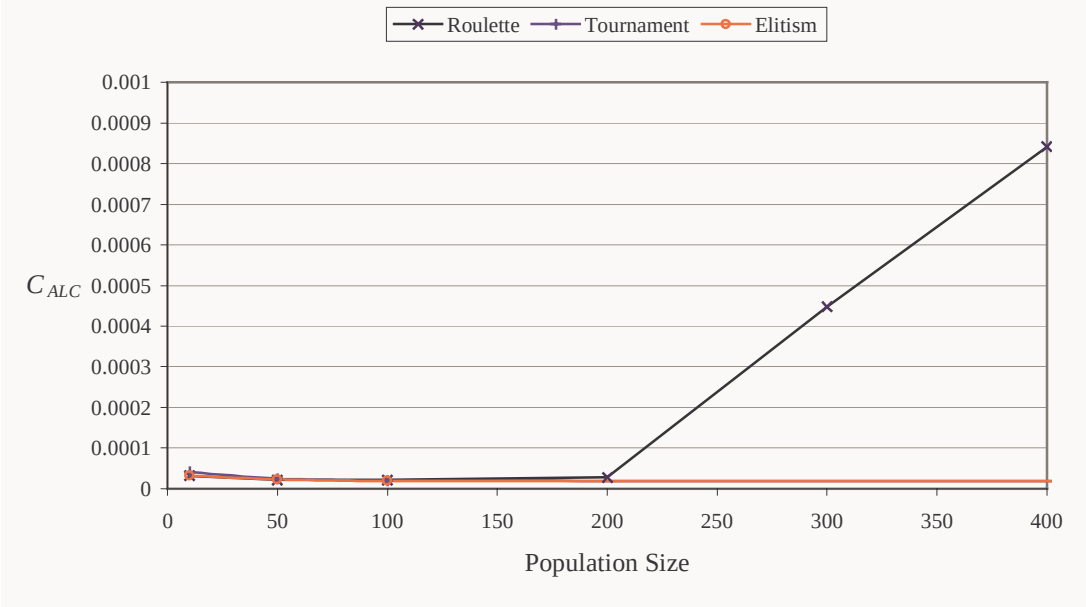

Figure 4.16: C_{ALC} as a function of population size – Roulette Wheel.

Figure 4.15 is replotted again in Figure 4.17, this time concentrating only on selection using Tournament and Elitism. C_{ALC} as a function of population size is similar for both selection techniques. In Tournament and Elitism selection, the increase in population size generally improves C_{ALC} in contrast to the situation when using Roulette Wheel selection. However, there is little improvement in C_{ALC} for population sizes in excess of 100.

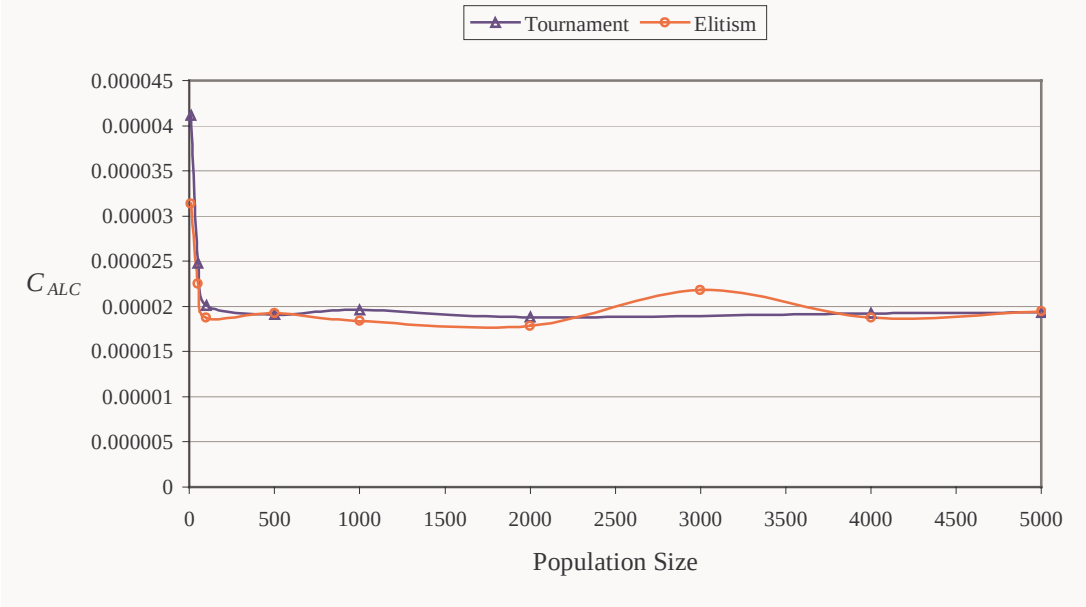

Figure 4.17: C_{ALC} as a function of population size – Tournament and Elitism (μ=0.1).

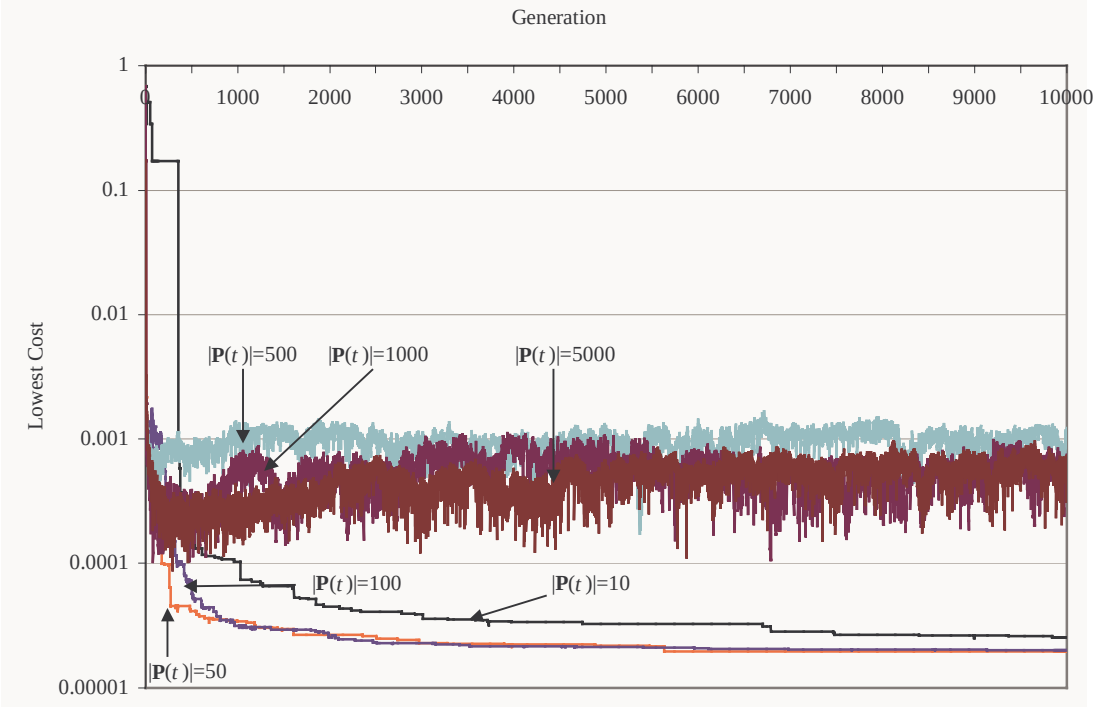

Figure 4.18: Lowest cost against generation - Roulette Wheel selection.

Figure 4.18 presents the lowest cost solution at each generation using Roulette Wheel Selection for various population sizes but with p_m fixed at 0.01. It shows that population sizes of 500, 1000 and 5000 converge to poor local optimums, while the smaller population sizes (i.e., 50 and 100) give a better solution. At a population size of 10, the system converges to a better solution than those with population sizes of

500 and above but it yields a solution that is worse than those with population sizes of 50 and 100. Although population size of 50 and 100 give similar results, a population size of 100 converges more slowly than when the population size is 50.

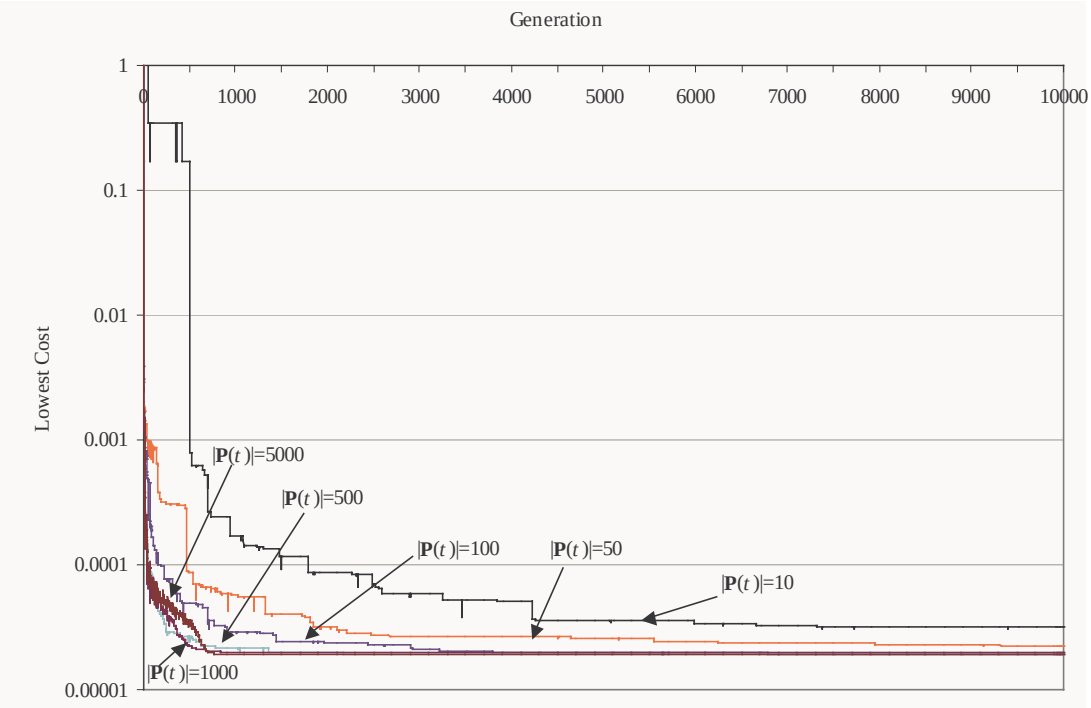

Figure 4.19: Lowest cost against generation - Tournament selection.

Figure 4.19 shows the lowest cost value as a function of the generation for Tournament selection, where the mutation rate is fixed at 0.01 for various population sizes. This figure also shows that a larger population size produces lower cost solutions. Apart from this, it is also observed that larger population sizes (500, 1000 and 5000) have a higher rate of convergence (less than 1000 generations) compared to that for a population size of 100 (which converges in excess of 3000 generations). However, the rate of convergence is about the same for population size in excess of 500. The use of a larger population requires more processing power and therefore it is not desirable.

4.3.3.2 Average Cost, C_{AVG}

In each generation, the average cost $\boldsymbol{C}_{AVG}(t)$ of the population is recorded. The average value of $\boldsymbol{C}_{AVG}(t)$ over the range from t = 2000 till 10000 for all three selection methods as a function of the population size is plotted in Figure 4.20. The average

value of $\boldsymbol{C}_{AVG}(t)$ is denoted as C_{AVG} and it measures the overall fitness (or cost) of the population.

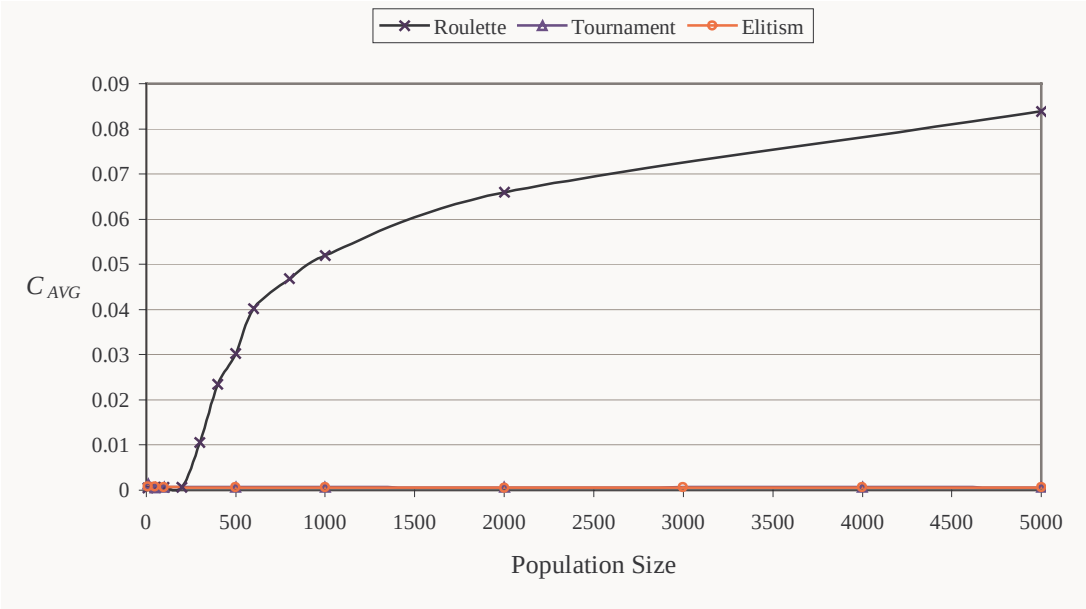

Figure 4.20: C_{AVG} against population size.

For Roulette Wheel selection, once the population size exceeds about 200, C_{AVG} increases as the population size increases as shown in Figure 4.20. As the population size increases, lower cost individuals are less likely to be selected for crossover thereby causing the overall population to have a high cost. Tournament and Elitism selections are generally better than that for Roulette Wheel selection as the population increases.

Figure 4.20 is re-plotted in Figure 4.21 concentrating only on Tournament and Elitism selection. It can be seen that for both selection methods, C_{AVG} saturates in excess of a population size of 100, following a similar pattern to the C_{ALC} curve presented in Figure 4.17. Elitism selection generally has a slightly lower C_{AVG} compared to that achieved by Tournament selection.

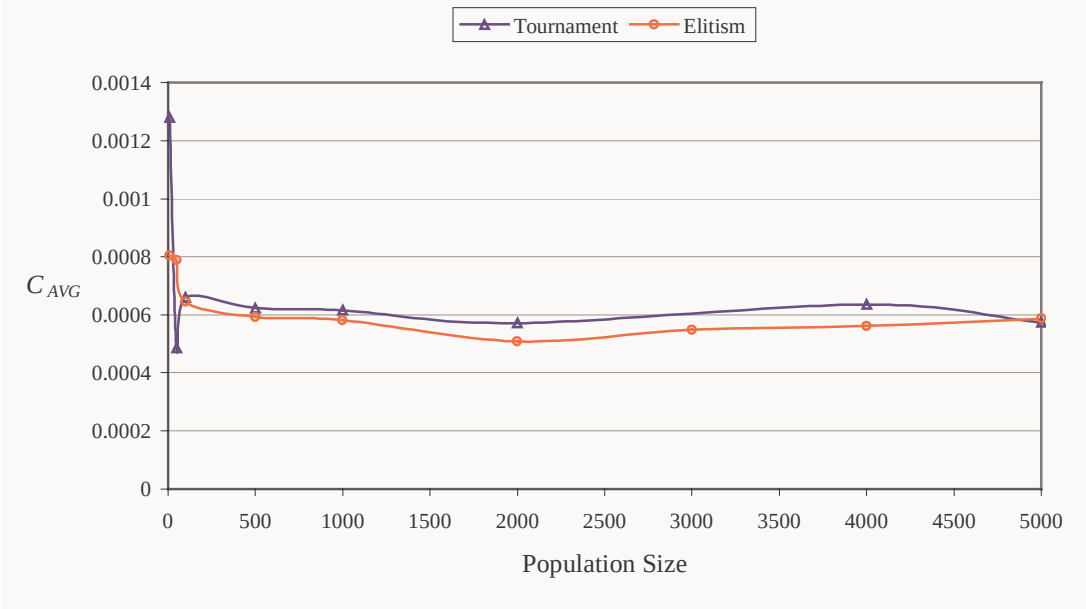

Figure 4.21: C_{AVG} against population size - Tournament and Elitism (μ=0.1).

4.3.3.3 *Diversity, D_{AVG}*

The standard deviation of the cost in the population in each generation is also recorded. The standard deviation is a measure of the diversity of the population. A population with a high diversity contains individuals that have larger differences. For each population size, the average diversity D_{AVG} over the range from the 2000th generation (where convergence is virtually complete) to the 10000th generation is plotted as a function of population in Figure 4.22.

Figure 4.22 shows that D_{AVG} for all three selection techniques increases with the population size. With more individuals, the diversity of the population is higher since there is a greater chance that each one will be different. A population with a high diversity has a higher chance of providing an individual with low cost than a population with low diversity. This is observed in the cases of Tournament and Elitism selection where greater population size giving a higher diversity produces a lower C_{ALC}. However, in Roulette Wheel selection the greater diversity owing to higher population size does not give a lower value of C_{ALC} since it is harder for these low cost individuals to be identified in a large population size.

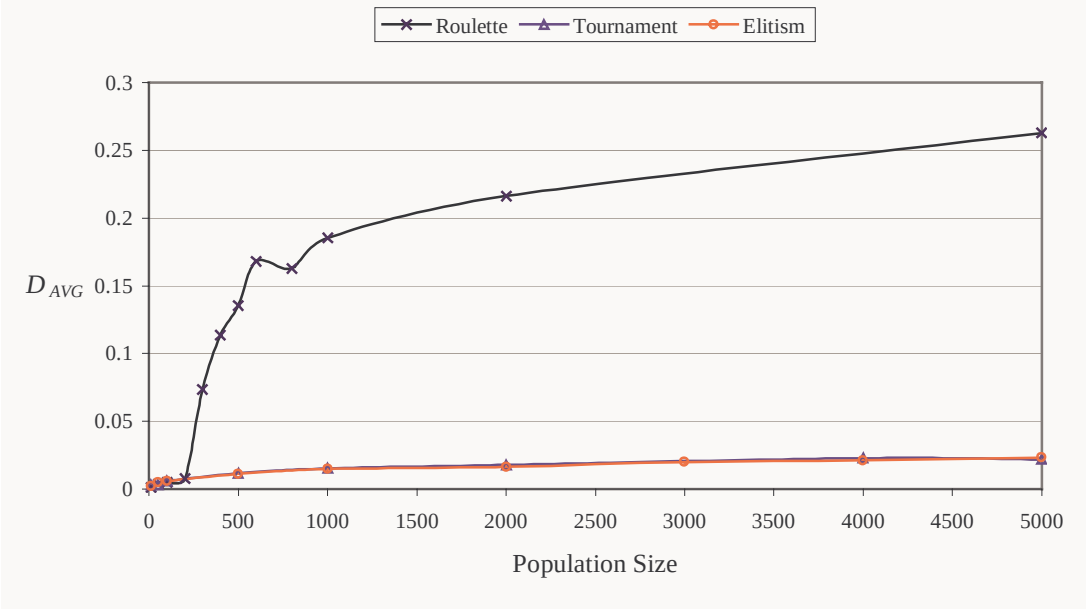

Figure 4.22: Average Diversity, D_{AVG} as a function of population size.

The population analysis performed suggests that Roulette Wheel should use a population size less than 200. Although higher populations give good solutions for Tournament and Elitism selection, a small population size (in the order of a few hundreds) is preferred since less processing is involved and the gain from using a higher population size is not significant.

4.3.4 Mutation Rate, p_m

The effect of mutation rate p_m has also been investigated. The population is fixed at 1000 individuals while p_m is varied from 0 to 0.5. For Elitism selection, the elite fraction μ = 0.1. In most scenarios investigated, it has been observed that convergence occurs after around 2000 generations.

4.3.4.1 Average Lowest Cost, C_{ALC}

The Average Lowest Cost, C_{ALC} over the range from the 2000th to 10000th generation as a function of the mutation rate is plotted in Figure 4.23. It is observed that using Roulette Wheel selection, increasing the mutation rate lowers C_{ALC}. However, this effect saturates when $p_m > 0.05$.

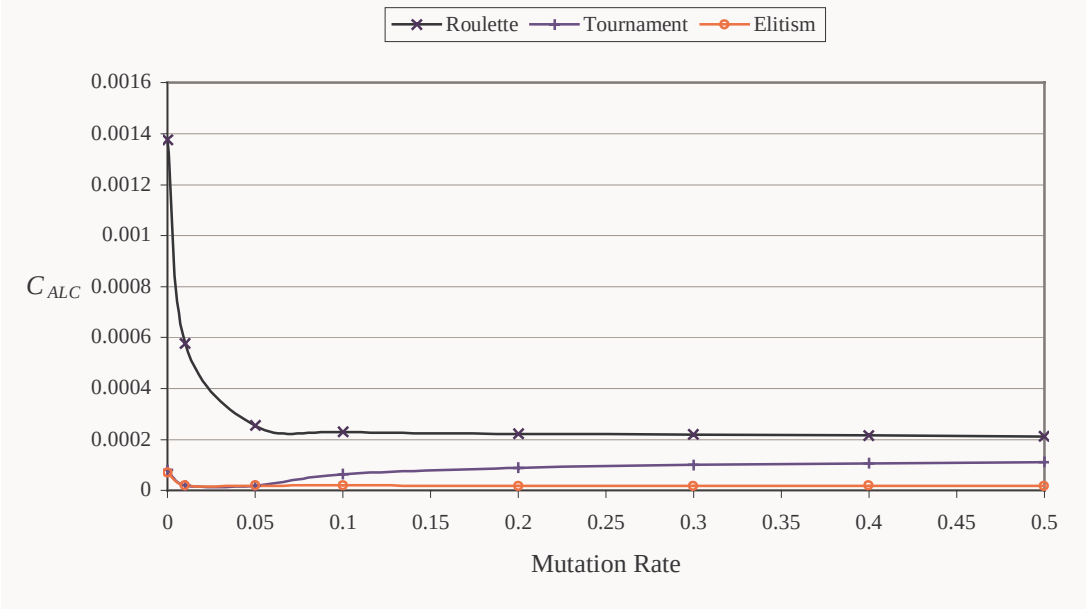

Figure 4.23: C_{ALC} as a function of mutation rate.

Figure 4.24: C_{ALC} as a function of mutation rate - Tournament and Elitism (μ=0.1).

The results of Figure 4.23 are re-plotted in Figure 4.24 without showing the Roulette Wheel selection results. Figure 4.24 shows that there is an optimum value for the mutation rate for Tournament selection, which lies between 0 and 0.1. Further increases in the mutation rate give rise to an increase in C_{ALC}. This is because in a population with high mutation rate, low cost individuals have a high probability of being mutated causing them to carry a high cost. However, when Elitism selection is used, the mutation rate does not affect the average lowest cost. This is because the

lowest cost individual is retained in the population and is not lost (due to high mutation rate) unlike the situation which occurs when Tournament and Roulette Wheel selection are used.

4.3.4.2 Average Cost, C_{AVG}

The results of a population analysis similar to that performed in Section 4.3.3.2 are shown in Figure 4.25 where C_{AVG} is plotted as a function of the mutation rate.

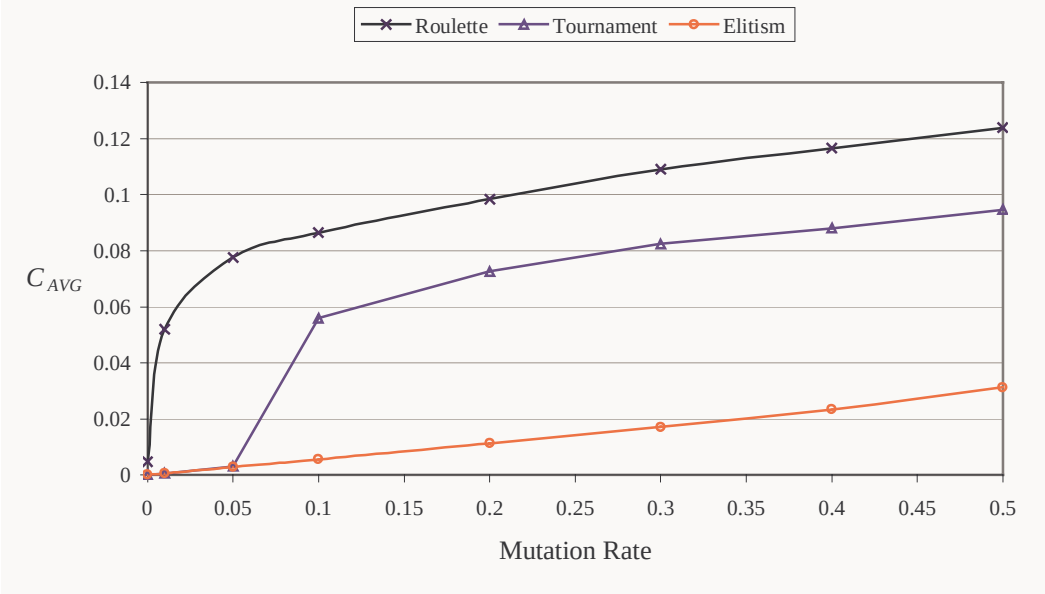

Figure 4.25: C_{AVG} as a function of mutation rate.

It is observed that C_{AVG} rises with an increasing mutation rate for all three selection methods. This is because a low cost individual is more likely to become a high cost individual under high mutation rate. Roulette Wheel selection has a higher value of C_{AVG} than that of Tournament selection while Elitism shows the lowest rise in C_{AVG} as a function of the mutation rate.

4.3.4.3 Diversity, D_{AVG}

The average diversity D_{AVG} over the range from the 2000th generation to the 5000th generation is plotted in Figure 4.26.

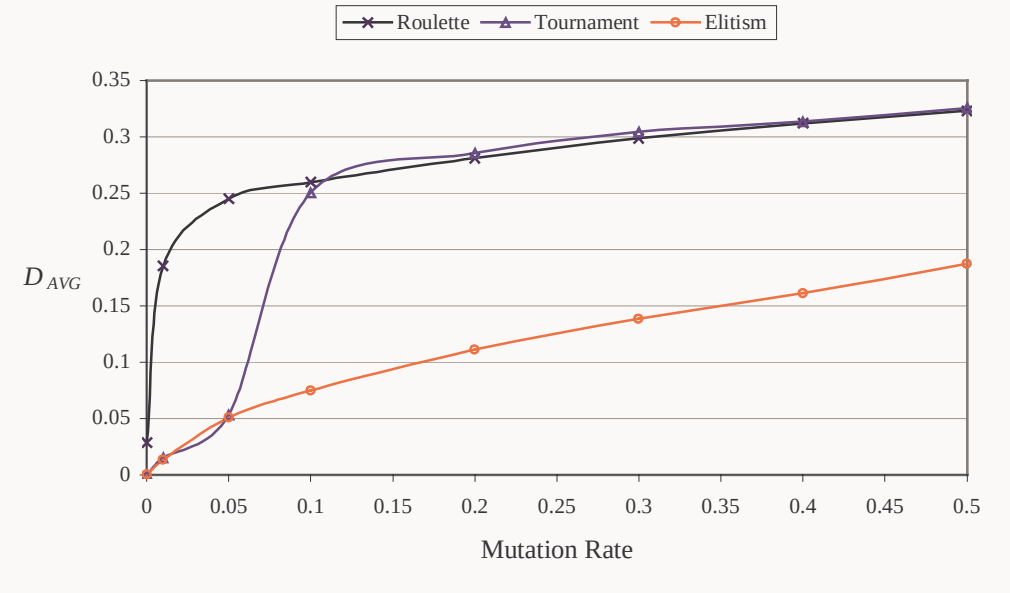

Figure 4.26: Average Diversity, D_{AVG} as a function of mutation rate.

An increase in the mutation rate D_{AVG} is observed for all selection methods inferring that mutation introduces diversity to the population. This widened diversity improves the chance of finding a good solution particularly in the case of Roulette Wheel selection as shown in Figure 4.23, though its effect is less significant when $p_m > 0.05$. Although diversity improves the chance of finding a good solution, a high mutation rate (that increases diversity) will destroy this good solution. Therefore, the gain from the increase in the mutation rate will reach a maximum after which mutation will cause more harm than good. The diversity in Roulette Wheel selection is more sensitive to the initial increase in the mutation rate compared to Tournament selection. As expected the increase in diversity brought about by Elitism selection is less than that in Tournament and Roulette Wheel selection.

The mutation rate analysis suggests that all three selection methods should use a small mutation rate of about 0.05.

4.3.5 Elitism – Elite Fraction μ

The effect of population size in Elitism selection is similar to that in Tournament selection since the remaining fraction (1- μ) following the application of Elitism uses Tournament selection. However, as we have seen Elitism selection is less sensitive to the increase in the mutation rate compared to Tournament selection. In this section, the effect of μ is analysed.

The mutation rate is fixed at 0.05 and the population size is fixed at 1000 while the elite percentage μ is varied. The Average Lowest Cost C_{ALC} (from the 2000th up to the 10000th generation) are plotted in Figure 4.27 for $\mu < 0.1$ and in Figure 4.28 for $\mu \geq 0.1$ respectively. These figures show that no clear pattern emerges in the variation of C_{ALC} as a function of μ.

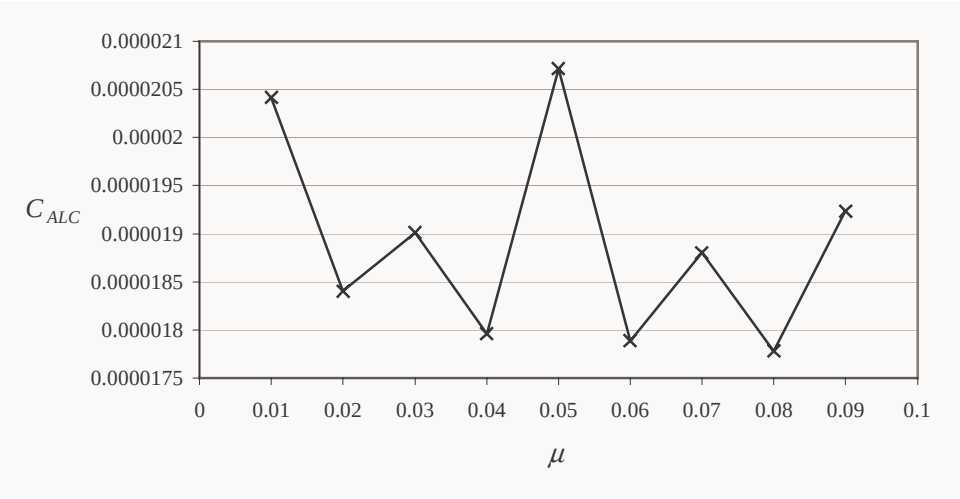

Figure 4.27: C_{ALC} as a function of Elite fraction (μ <0.1).

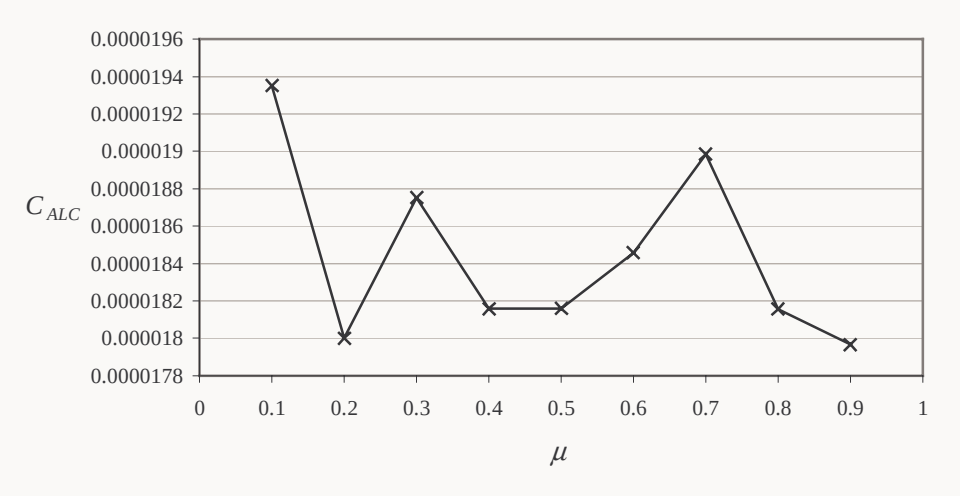

Figure 4.28: C_{ALC} as a function of Elite fraction ($\mu \geq 0.1$).

Figure 4.29 is a plot of the lowest cost up to the 1500th generation (note the system converges after 1500 generations).

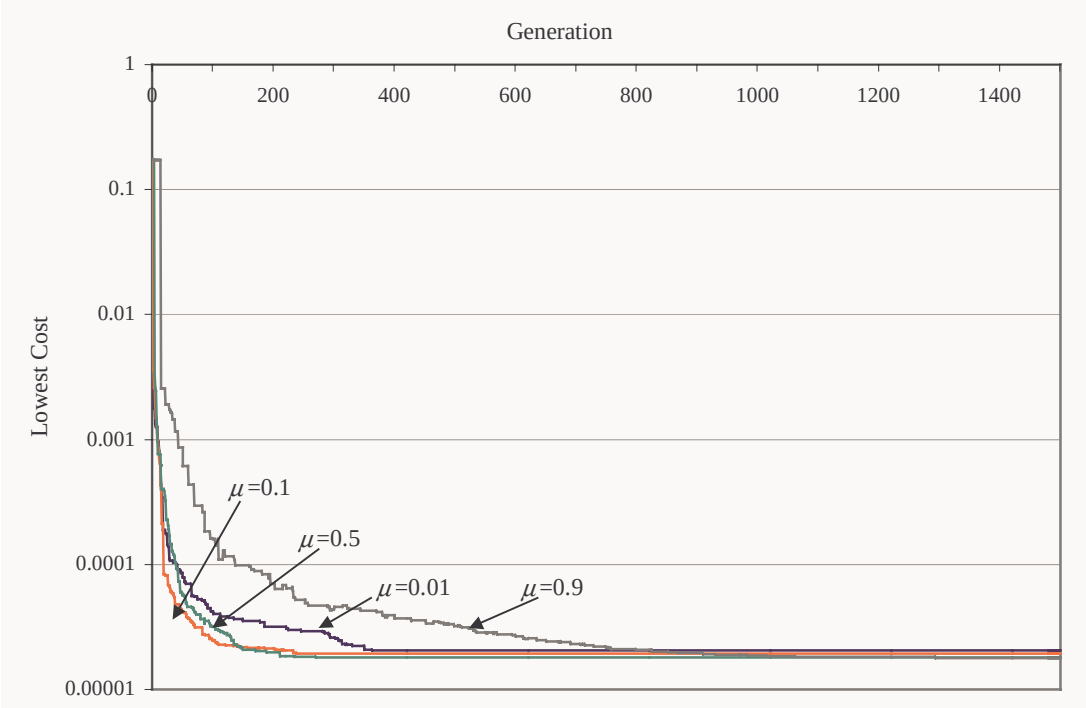

Figure 4.29: Lowest cost against generation – Elitism.

As observed in Figure 4.29, at very high values of μ (e.g. μ = 0.9) the system converges more slowly than when lower values of μ (e.g. μ = 0.1) are used. Figure 4.19 shows that Tournament selection converges after about 900 generations for a population size of 1000. This is higher than the results shown in Figure 4.29, which indicate a convergence time of about 400 generations for Elitism selection with μ = 0.01. Although μ has a less than obvious effect on C_{ALC}, it has a marked effect on the convergence time, where a minimum convergence time is obtained when $\mu \approx 0.1$.

Figure 4.30 and Figure 4.31 show C_{AVG} calculated over the range from the 2000th to the 10000th generation for $\mu < 0.1$ and $\mu \geq 0.1$ respectively. Although from the previous results it is observed that C_{ALC} is not affected by μ, it is seen the C_{AVG} improves as μ increases.

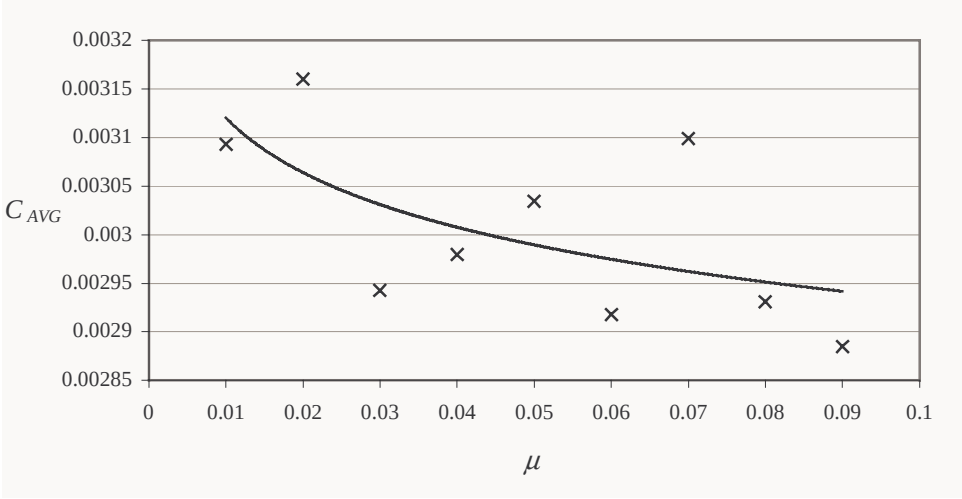

Figure 4.30: C_{AVG} as a function of Elite fraction (μ <0.1).

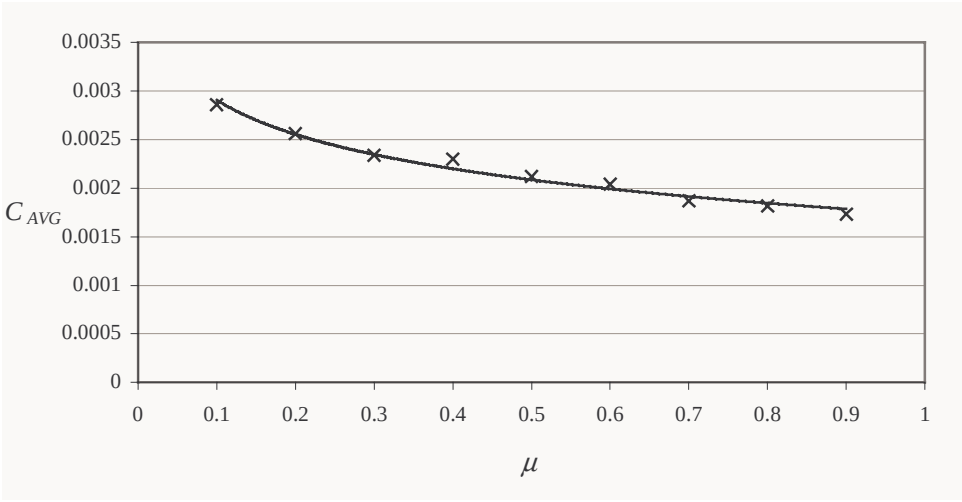

Figure 4.31: C_{AVG} as a function of Elite fraction ($\mu \geq 0.1$).

Figure 4.32 and Figure 4.33 are the average diversity D_{AVG} calculated over the range from the 2000th generation to the 10000th generation for $\mu < 0.1$ and $\mu \geq 0.1$ respectively. It can be seen that the diversity of the population decreases as μ increases but from the previous results the population tends towards lower cost individuals. Retaining a higher percentage of the fittest individuals from previous generations causes the system to change slowly while the diversity decreases.

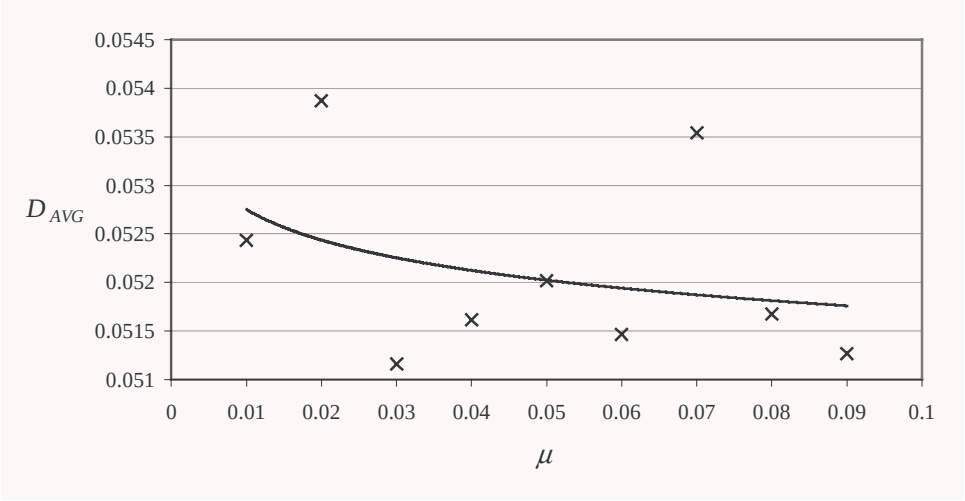

Figure 4.32: Average Diversity, D_{AVG} as a function of Elite fraction (μ <0.1).

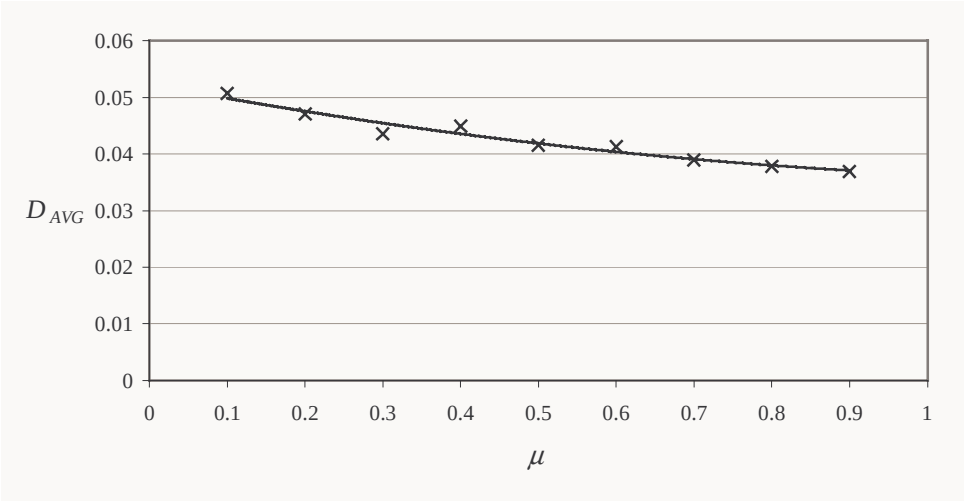

Figure 4.33: Average Diversity, D_{AVG} as a function of Elite fraction ($\mu \geq 0.1$).

From this analysis, the lowest cost value is largely independent of the Elite fraction μ. An Elite fraction μ = 0.1 is suggested from this analysis since it gives a fast rate of convergence.

4.4 Simulation and Results

The three algorithms introduced previously, namely RND, LI and FCA-GA are simulated using OPNET. The traffic and radio propagation models (with shadowing) utilise those detailed in Chapter 3. The scenario used consists of a network with 136 APs and 669 SUs non-uniformly distributed over 37 cells with the layout shown in Figure 4.34. To minimise boundary effects, results are taken from the three shaded cells that are surrounded by other cells and are closer to the centre of the layout. At

the start of the simulation, all the APs use the same channel (Channel 1) except for FCA-GA where the initial channel assignment is used throughout the simulation.

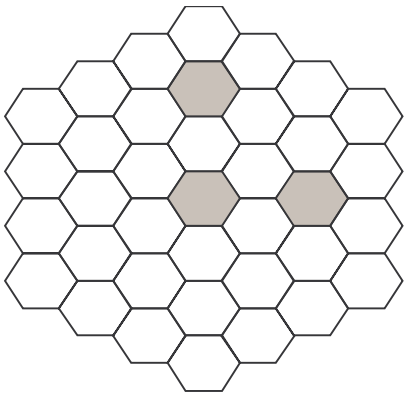

Figure 4.34: Simulation layout (37 Cells). Results are taken from shaded cells.

For the simulation using LI, the T_{TX}/M_{MAC} ratio is set as close as possible to 11.44. This value is used so that the data throughput is normalised with the DCA methods that will be introduced in the next chapter. Details concerning the selection of this value are explained in Section 5.4.4.1 of the next chapter.

The genetic algorithm parameter analysis performed in the previous sections showed that for the 7 cell network (with 50 APs), the Elitism selection method with p_m = 0.05, a population size of several hundred and μ = 0.1 gave a low cost solution and achieved superior convergence rates compared to the other combinations of selection method and parameters. Hence, the Elitism selection method with the same parameters will now be employed for the FCA-GA algorithm operating in a network with 37 cells. The lowest cost string (i.e. channel allocation) is found from a population of 200 and this string is used in the simulation. Each simulation is run for 36000 nmsec.

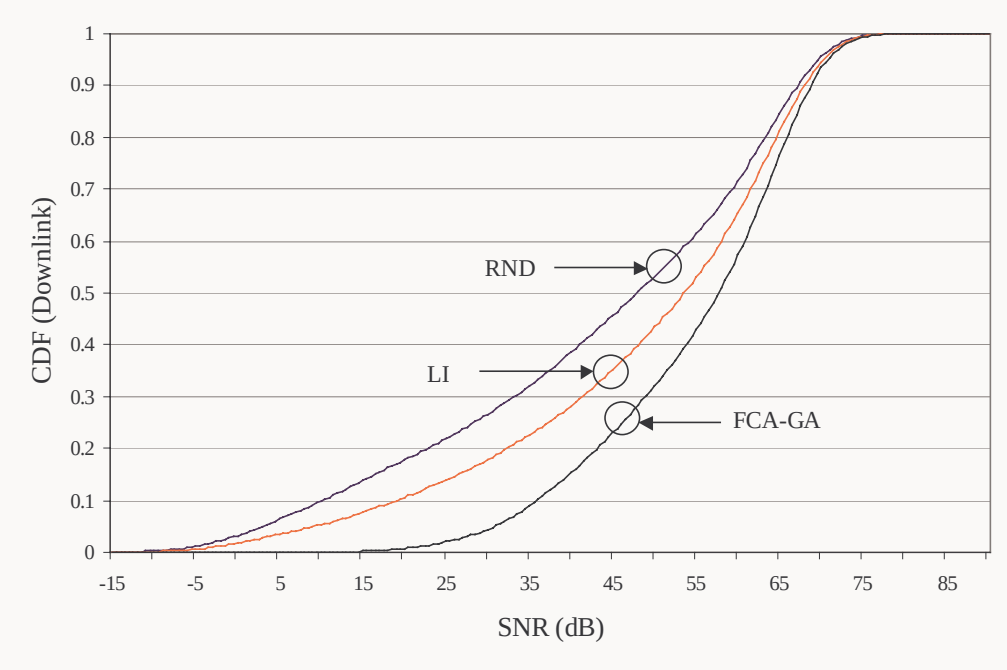

Figure 4.35: Downlink SNR (dB) performance for the shaded cells.

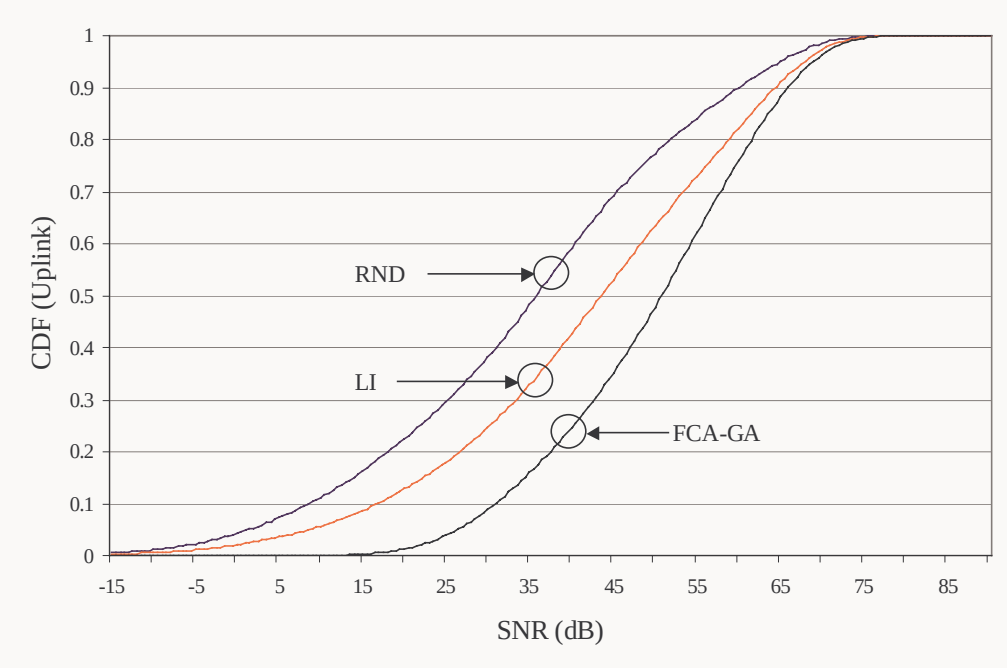

Figure 4.36: Uplink SNR (dB) performance for the shaded cells.

Cumulative distribution functions (CDF) giving the simulated received SNR performances for the downlink and uplink directions are presented in Figure 4.35 and in Figure 4.36 respectively for all the APs and SUs in the shaded cells of Figure 4.34. The presented results are obtained during the final 2 normalised hours (7200 nmsec NOT real time) of simulation to ensure the network has reached a steady state and that

the transient state that yields poor results has passed (since all AP uses the same channel at the start of the simulation).

In these figures, it can be observed that RND has the worst SNR performance, that LI has a better SNR performance than RND and that FCA-GA has the best SNR performance. The order of the results is consistent in the downlink and uplink directions. As expected, the performance is proportional to the amount of information available in each channel allocation method. RND does not use any network or prior information about the system and hence has the worst performance while FCA-GA has full knowledge of the entire network and the fact that the interference environment does not change much (expect for the shadowing) gives FCA-GA almost perfect information of the interference environment thereby giving it the best SNR performance. LI utilising measured information of the interference environment consistently gives a better SNR performance than RND. However, with only local information of the interference environment, the performance of LI is poorer than that of FCA-GA.

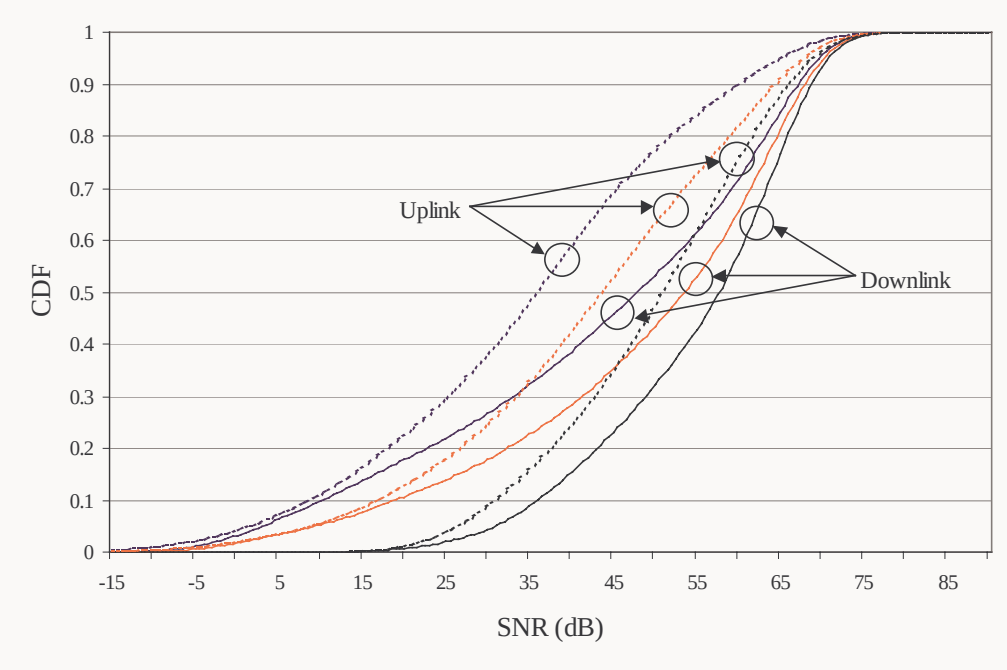

Figure 4.37: SNR (dB) downlink and uplink CDF performances for all shaded cells.

Although the measurement is performed by the AP (i.e. in the uplink direction), Figure 4.37 shows that the uplink received SNR performance in general poorer than that in the downlink direction. This is because the SUs use a highly directional (20°

beamwidth) antenna compared to the sectored (60° beamwidth) antenna used by the AP and hence the downlink direction is less susceptible to interference.

The first percentile SNR (denoted as 1% SNR) by definition means that there is a probability of 0.01 that the SNR of a received packet is below this value. The 1% SNR for the downlink and uplink taken from Figure 4.35 and Figure 4.36 respectively are listed in Table 4.1. Also in Table 4.1 are the 1% SNR values obtained when the overall downlink and uplink statistics are combined. In this dissertation, we assume that a packet is received successfully (without any error) if its received SNR is at least 21 dB. The probability of this occurring, namely $P_{\geq 21}$ is also shown in Table 4.1 for the various channel allocation methods. The value of $P_{\geq 21}$ is obtained from the combined statistics of the overall downlink and uplink performance. The effective data throughput for an AP is the fraction of G_{MAC} (data throughput) that has a received SNR of at least 21 dB. The effective data throughput is hence $G_{MAC} \times P_{\geq 21}$. The average effective data throughput π_{AVG} of all the APs in the shaded cells is recorded in Table 4.1 for various channel allocation methods. The average effective data throughput π_{AVG} is obtained from the average G_{MAC} of all the APs (in the shaded cells) multiplied by $P_{\geq 21}$. Since RND and FCA-GA do not perform any measurements, their value of G_{MAC} is unity since they will effectively transmit at all times.

	RND	LI	FCA-GA
1% SNR (downlink)	-5.0 dB	-2.5 dB	21.5 dB
1% SNR (uplink)	-10.5 dB	-5.5 dB	19.5 dB
1% SNR (overall)	-8.0 dB	-4.5 dB	20.0 dB
$P_{\geq 21}$	0.784	0.871	0.987
π_{AVG}	0.784	0.854	0.987

Table 4.1: 1% SNR and π_{AVG}.

LI which utilises measured information, has an overall 1% SNR gain of 3.5 dB over RND while FCA-GA which has almost full information of the network has a significant 1% SNR gain of 24.5 dB over LI. Using FCA-GA, 98.7% of all packets transmitted or received will be successful compared to 87.1% in LI and 78.4% in RND. LI needs to spend time measuring the interference power of each channel and hence loses efficiency compared with FCA-GA and RND that do not perform any measurements. However, due to the poor SNR performance of RND, LI has a higher average effective data throughput π_{AVG} compared to RND (even though RND does not

do any measurement). The difference in π_{AVG} between LI and RND is 0.07, which in a nominal 25 Mbps link corresponds to a gain of 1.75 Mbps.

The average channel utilisation is defined as the average number of APs using each channel during the simulation. The average channel utilisation is recorded after the simulation has passed the transient stage and it can be seen in Figure 4.38 that all 3 channel allocation methods achieve a uniform channel utilisation. FCA-GA channel utilisation is fixed and designed to be uniform across all channels. In RND, the channels are selected from a uniform distribution and therefore the average channel utilisation during the simulation is uniform. LI picks the least interfered channel for allocation and it is interesting to note that doing this also achieves uniform channel utilisation. This also validates our initial assumption (made previously in section 4.2) that the probability that a selected channel is also the least interfered channel is $1/C$ (where C is the total number of channels).

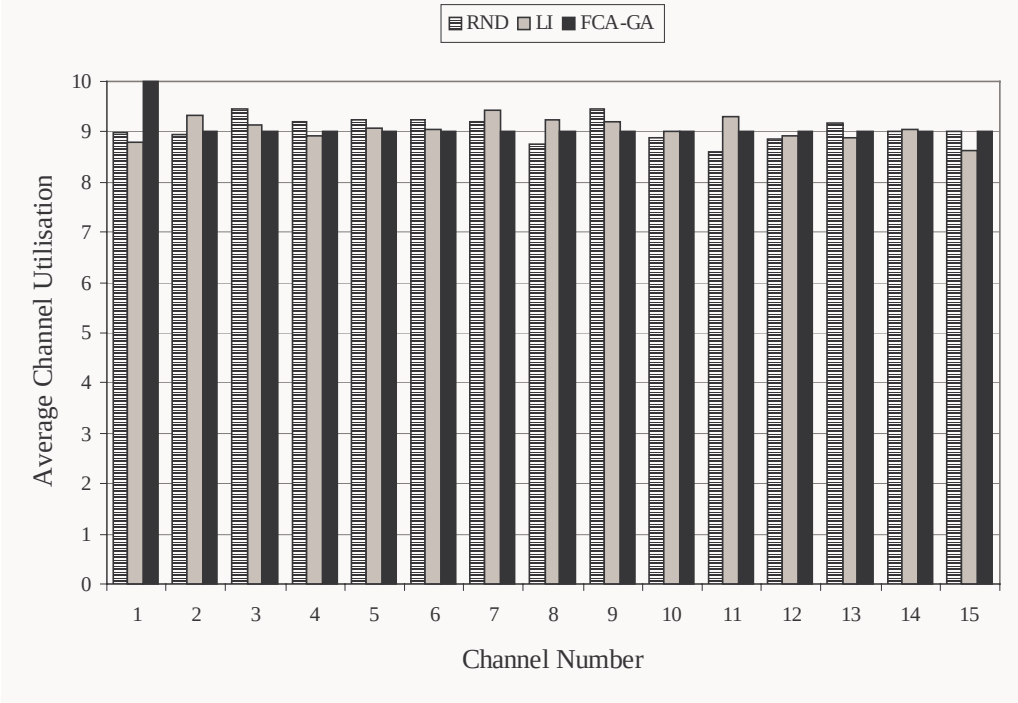

Figure 4.38: Average Channel Utilisation.

The channel fluctuation is a measure that describes how frequently an AP changes channel and this is defined as the standard deviation of the channel utilisation. In FCA-GA, the channel utilisation is fixed and the AP does not change channel, consequently there is no channel fluctuation in FCA-GA. The channel fluctuation

(recorded in the final 2 normalised hours or 7200 nmsec of simulation) for all channels with RND and LI are plotted in Figure 4.39.

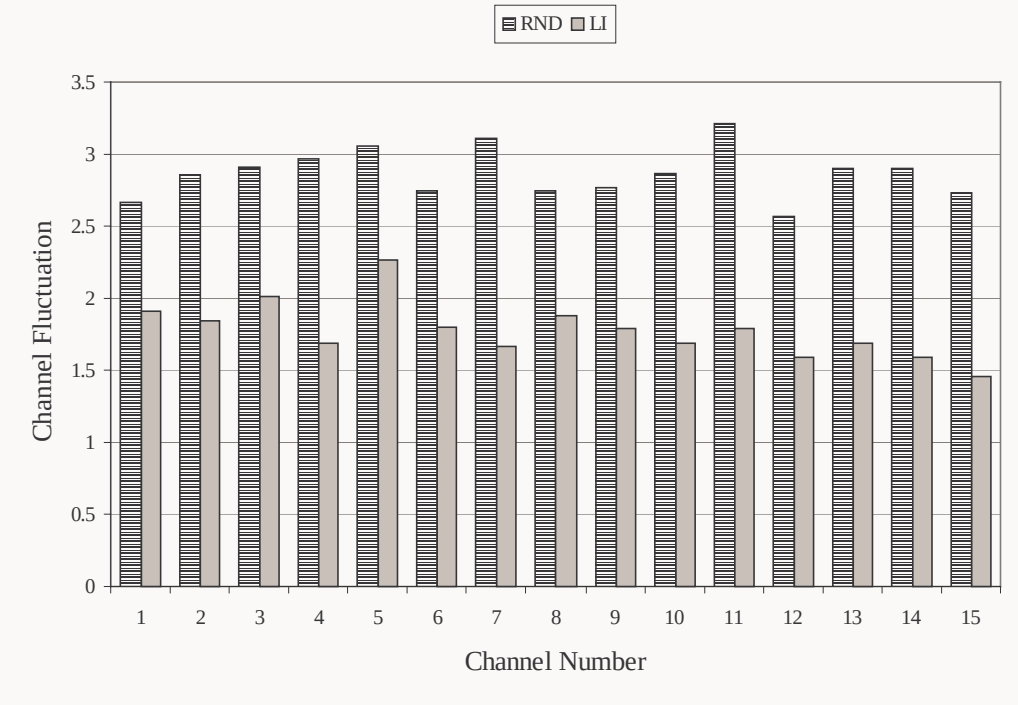

Figure 4.39: Channel fluctuation.

It can be seen that in general, RND has a higher channel fluctuation than LI. On average, RND has a channel fluctuation of 2.87 compared to a channel fluctuation of 1.78 in LI. The higher channel fluctuation in RND is to be expected since it changes channel at every MAC frame. Theoretically, if LI manages to correctly identify the least interfered channel (e.g. by coordinating channel measurement in a synchronised system) at all times it should be able to converge to a stable channel allocation pattern [52]. However, this is difficult to achieve in an asynchronous system. The convergence time can be found by running the simulation until no channel fluctuation occurs. Unfortunately, this is not practical since as described previously, OPNET consumes large amount of processing power and furthermore in an asynchronous system convergence may not be guaranteed. Despite this, LI is able to hold onto the least interfered channel for a longer period compared with RND. In LI, a high level of channel fluctuation in an AP makes it difficult for other APs to predict its channel usage, making it more difficult for the other APs to find the least interfered channel. This is because the interference environment measured in each channel during the

SCAN portion may change significantly during the transmit portion if the channel fluctuation is high.

4.5 Conclusion

In this chapter we have analysed both the RND and the LI methods and applied them to the BFWA system described in Chapter 3. A FCA-GA method is also proposed for BFWA [82] with the assumption of a static interference environment. Though FCA-GA does not perform any measurements, it does have a priori information of the entire network. The control parameters that can be set in a genetic algorithm when applied to FCA-GA are extensively investigated. The performance of a channel allocation technique is proportional to the amount of information that is available and also how effectively this information is used. RND uses the least information while FCA-GA uses the most information. Hence, the performances of RND and FCA-GA will act as benchmarks for the performance of other DCA methods.

The simulation results show that FCA-GA gives the best SNR performance while RND gives the worst SNR performance. LI using additional measured information has a better SNR performance than RND. Channel allocation is a NP-complete problem and hence the FCA-GA may not be practical for a very large network since the probability of it finding a good solution may be low while the complexity of the algorithm increases. Also the interference environment may not be fixed, particularly in a system that utilises unlicensed spectrum.

5 DISTRIBUTED DCA

In the previous chapter, it has been shown that the performance of a channel allocation method improves as the amount of information available increases. The effective use of information also affects the performance of the network. In this chapter, we will investigate the performance of a distributed DCA, which uses the available information more effectively.

LI is a fully distributed allocation method, where there is no communication among the APs. Therefore, the network information (e.g. channel usage of other APs) cannot be determined precisely using measurements. By improving the LI method, a new DCA using Game Theory (DCA-GT) is proposed. The quantity of measured information used in DCA-GT is identical to that in LI.

5.1 Game Theory

Interdependency among a group of individuals exists if the independent choices made by each individual within the group affect the outcome of the entire group. Game theory studies the interdependence strategies that present in such groups. In a game, there are rules that describe the following four items [85]:

- The players in the game – the individuals within the group,
- The set of strategies each player can select – the available choices,
- The order each player gets to play (e.g. simultaneously or by taking turns)
- The amount of payoff gained (or lost) by each player due to the choices made.

A popular example of Game Theory is the Prisoners' Dilemma. The players in this game are two prisoners, Prisoner 1 and Prisoner 2. The set of strategies is **S**={"Confess", "Don't Confess"}, that is each prisoner can either confess or not confess to the crime they are charged with. The two prisoners are locked in different cells such that they cannot communicate with each other. The prisoners must therefore make their choices independently and simultaneously. The payoff each prisoner gets is the length in years of the prison sentence. The game can be represented in two formats: the strategic (or normal) form or the extensive form. The

strategic form of the game is represented in a matrix while the extensive form is represented using a tree diagram. In the Prisoners' Dilemma game, the strategic form of the game is shown in Figure 5.1. The matrix shows the payoff for each player, where the non-shaded areas correspond to the payoff for Prisoner 1 and the shaded areas correspond to the payoff for Prisoner 2.

Payoff for Prisoner 1

Payoff for Prisoner 2

		Prisoner 2			
		Don't Confess		Confess	
Prisoner 1	Don't Confess	1 year	1 year	10 years	0 years
	Confess	0 years	10 years	6 years	6 years

Figure 5.1: Strategic form of the Prisoners' Dilemma.

The payoff for each prisoner is dependent on the combined strategies of both prisoners. Let strategy *D* be "Don't Confess" and strategy *C* be "Confess." The payoff for Prisoner *j* is $\pi_j(s_j, s_k)$ if Prisoner *j* selects strategy s_j while the other prisoner selects s_k, where s_j and s_k can either be *D* or *C*. The extensive form of the Prisoners' Dilemma is shown in Figure 5.2.

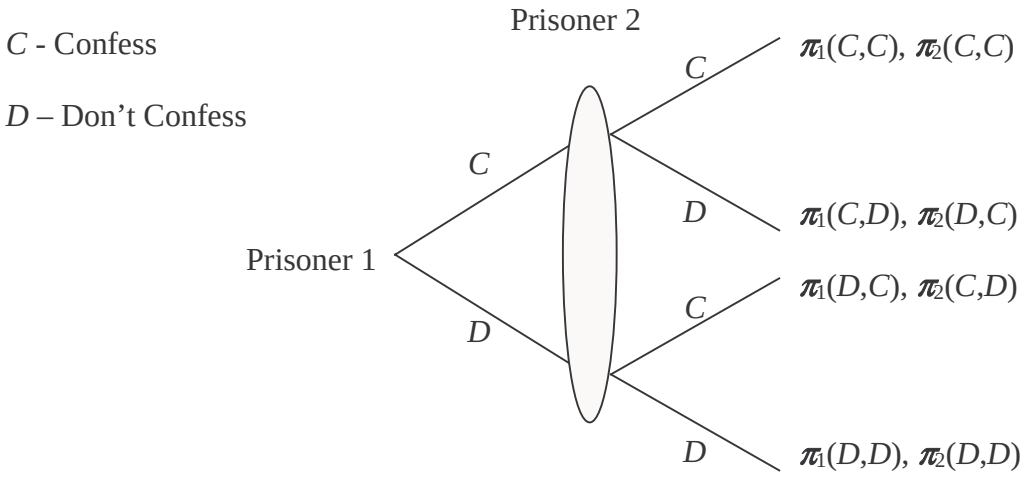

Figure 5.2: Extensive form of the Prisoners' Dilemma.

In the extensive form, each branch corresponds to the strategy selected by the player and the end of the tree gives the payoff for each player. The oval in Figure 5.2 means that the strategies are selected by both players simultaneously.

The best payoff for Prisoner j is $\pi_j(C, D)$, that is to walk away free (i.e. 0 years sentence) but this will cost the other prisoner the heaviest sentence of 10 years. Let $\mathbf{S}$ be a set of available strategies. Note in this game, the aim is to minimise the sentences, so a shorter sentence has a higher payoff. The best response $\boldsymbol{R}(s) \in \mathbf{S}$ is the strategy that will give a player the highest payoff if the other player selects strategy $s \in \mathbf{S}$. The best response to "Don't Confess" is thus to "Confess" (i.e. $\boldsymbol{R}(D) = C$). The Nash Equilibrium is defined as a point (i.e. strategy vector $\mathbf{s}_N$) that is stable where no players in the game have any incentive to change their strategies. For an N player game, $\mathbf{s}_N=(s^*_1, s^*_2, \ldots, s^*_N)$ is the Nash Equilibrium if the best response for player j is $\boldsymbol{R}_j(s^*_1, s^*_2, \ldots, s^*_{j-1}, s^*_{j+1}, \ldots, s^*_N)=s^*_j$ for $j = 1$ to N. In the Prisoners' Dilemma, Nash Equilibrium is achieved when both prisoners confess, i.e. the strategy vector is (C, C) since $\boldsymbol{R}(C)=C$ (confessing gives a prison sentence of 6 years compared to don't confess which has a prison sentence of 10 years). This strategy vector (C, C) is also the fixed point (i.e. $\boldsymbol{R}(C)=C$) of the best response function.

If the Prisoner Dilemma can be played several times or an infinite number of times then the pure strategy with strategy vector (C, C) will not give the maximum payoff for each player. Rather than playing a pure strategy (C, C), the players can use a mixed strategy where the player uses a combination of different strategies. For example, each time the game is played, if each player selects the opposing strategy (i.e. Prisoner 1 confesses while Prisoner 2 does not confess) and alternate their choices in the subsequent games, they in turn would obtain a higher payoff in the long term.

Game Theory has been used since the 19th century and its uses have ranged from economic analysis to the study of the arms race in the Cold War. Game Theory is also used in communications engineering for example in the design of a random access MAC schemes [86] and in power control [74]. In the next section, we will use Game Theory in a distributed DCA.

5.2 Game Theory in Distributed DCA

The interdependency characteristic of Game Theory is present in a distributed DCA. In a distributed DCA, a channel selected by each AP independently of other APs will change the interference environment of the network and this in turn will affect the received SNR performance of each AP. In the following sections, we analyse game theory in a distributed DCA and proposed a new DCA method, namely DCA Game Theory (DCA-GT) [87], [88], which is in the Distributed Measurement quadrant of the Channel Allocation Matrix.

In a general distributed DCA, the game is the BFWA network and the players are the APs in the network. In the BFWA network being considered, the APs operate asynchronously and therefore the APs make their decisions simultaneously. Channel allocation is repeated at each SCAN portion and hence this game is an infinitely repeated game (i.e. is played indefinitely or until the simulation is completed). We analyse a single play of the game. The players in the game, the payoff function and the strategies are described in the following sections.

5.2.1 The Players

As described in Chapter 3, for a data network it is the SNR which is important from a user's perspective. Power control is not implemented in the BFWA model and all APs and SUs transmit at a fixed power. Hence, the signal power is a function of the SU's distance from its AP. However, we can improve the received SNR by reducing the level of interference and noise. Since the BFWA uses a TDD method, the sources that interfere with an AP can also interfere with its corresponding SUs. For example, if AP k and AP j use the same channel, the SUs belonging to AP k will also interfere with AP j and AP j's SUs. Therefore, we will focus on the channel usage of the AP since it also represents the channel usage of its corresponding SUs.

The noise power $\boldsymbol{N}_j(t)$ received at AP j at time t is given as:

$$\boldsymbol{N}_j(t) = \sum_{k \in \mathbf{B}_j(t)} \boldsymbol{I}_k(t) + n_{RX} \tag{5.1}$$

where, $\mathbf{B}_j(t)$ is the set of APs in the BFWA network that use the same channel as AP j at time t and n_{RX} is the receiver noise of AP j, which will be assumed to be constant

and identical for all APs. $\boldsymbol{I}_k(t)$ is the interference power received at AP j due to AP k and/or its SUs at time t. The average total noise power $\overline{N_j}$ for AP j over time τ, is thus:

$$\overline{N}_j = \frac{1}{\tau}\int_0^{\tau} \sum_{\substack{k\in\mathbf{B}\\ k\neq j}} \boldsymbol{I}_k(t)\boldsymbol{P}_{k,j}(t)\,dt + n_{RX} \tag{5.2}$$

Where, $\boldsymbol{P}_{k,j}(t)$ is the probability of AP k and AP j using the same channel and **B** is the set of all APs in the BFWA network. The average noise power is therefore dependent upon $\boldsymbol{P}_{k,j}(t)$ and since the APs operate independently, we can focus on a pair of APs. The players in this game are two APs in the BFWA network namely AP j and AP k.

5.2.2 Payoff Function

The payoff is something that the players in the game value and consequently they will try to maximise it. In the DCA game, the payoff is related to the data throughput that each player achieves. For the two player scenario, AP j and AP k, we define the payoff function $\boldsymbol{\pi}_{j,k}(t)$ for AP j at time t as the percentage of total bits transmitted and received per nmsec by AP j that are interference free from AP k. This is expressed as:

$$\boldsymbol{\pi}_{j,k}(t) = \boldsymbol{G}_j(t)\left((1-\boldsymbol{P}_C(t))\boldsymbol{O}_{j,k}(t) + \boldsymbol{S}_{j,k}(t)\right) \tag{5.3}$$

where, $\boldsymbol{G}_j(t)$ is defined as the data throughput in a similar manner to G_{MAC} in (4.2) but is time dependent since the transmit portion of AP j, namely $\boldsymbol{T}_j(t)$ can have different lengths. The data throughput $\boldsymbol{G}_j(t)$ of AP j at time t is given by:

$$\boldsymbol{G}_j(t) = \frac{\boldsymbol{T}_j(t)}{\gamma C + \boldsymbol{T}_j(t)} \tag{5.4}$$

where, γ is the scan period for one channel and C is the number of channels measured during the SCAN portion of the MAC. $\boldsymbol{T}_j(t)$ is the transmit period (the time between two SCANs) for AP j at time t. This is the same definition as for T_{TX_j} in Chapter 4 but here it is generalised since the transmit portion need not be the same for different APs as is the case in LI.

$\boldsymbol{P}_C(t)$ is the probability that AP j and AP k use the same channel and is given in (4.5) for LI. $\boldsymbol{O}_{j,k}(t)$ and $\boldsymbol{S}_{j,k}(t)$ are the overlap functions. $\boldsymbol{O}_{j,k}(t)$ is the average fraction of $\boldsymbol{T}_j(t)$ that would coincide with $\boldsymbol{T}_k(t)$ and $\boldsymbol{S}_{j,k}(t)$ is the average fraction of $\boldsymbol{T}_j(t)$ that coincides with the SCAN portion of AP k's MAC. This is shown in Figure 5.3 where $\boldsymbol{o}_{j,k}(t)$ is the actual amount (in nmsec) of $\boldsymbol{T}_j(t)$ that coincides with $\boldsymbol{T}_k(t)$ while $\boldsymbol{s}_{j,k}(t)$ is the actual amount (in nmsec) of $\boldsymbol{T}_j(t)$ that coincides with the SCAN portion of AP k.

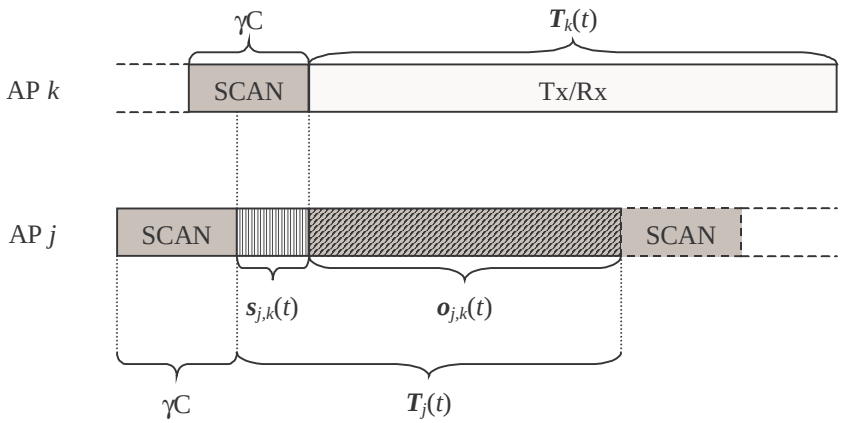

Figure 5.3: Overlap functions.

The amount of overlap $\boldsymbol{o}_{j,k}(t)$ is found using (4.8) and using a similar approach to that presented previously in Section 4.2, the overlap functions $\boldsymbol{s}_{j,k}(x)$ and $\boldsymbol{o}_{j,k}(x)$ for a single transmit portion of AP j can be represented graphically in Figure 5.4 for the situation where $\boldsymbol{T}_j(t) \leq \gamma C + \boldsymbol{T}_k(t)$.

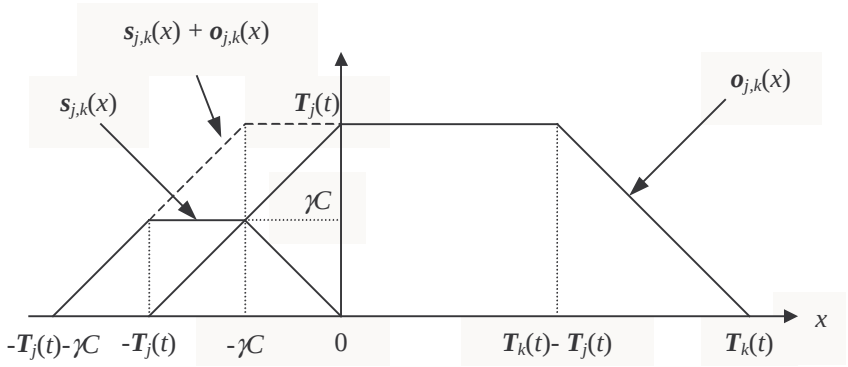

Figure 5.4: Overlap functions $\boldsymbol{s}_{j,k}(t)$ and $\boldsymbol{o}_{j,k}(t)$.

In the case where $\boldsymbol{T}_j(t) > \gamma C + \boldsymbol{T}_k(t)$, that is within the transmit portion of AP j, there are several consecutive SCANs and transmit portions of AP k, the overlap functions can be calculated by consolidating the transmit portions of AP k into one large

transmit portion as shown in Figure 5.5. If $\boldsymbol{T}_j(t)$ is larger than $\boldsymbol{T}_k(t)$, then AP j will benefit from the measurements performed by AP k.

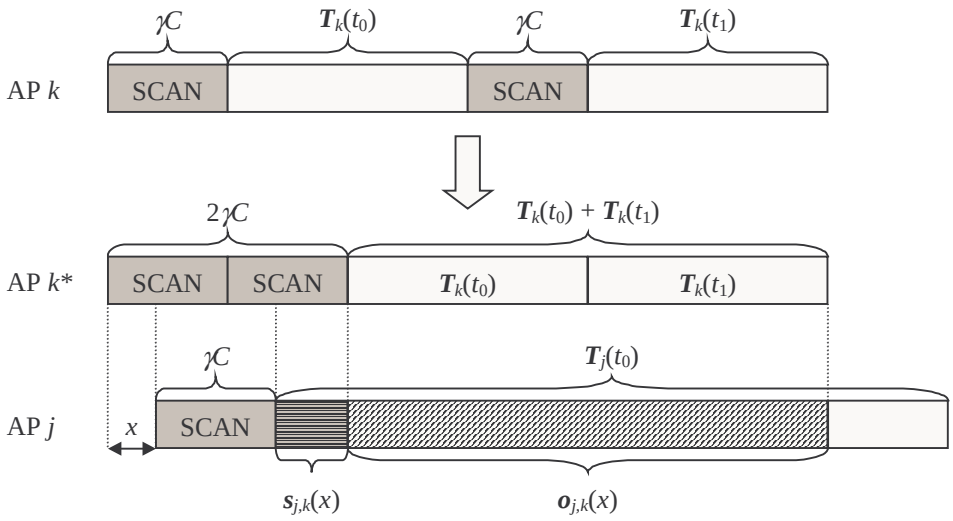

Figure 5.5: Overlap functions when $\boldsymbol{T}_j(t) > \gamma C + \boldsymbol{T}_k(t)$.

In the example shown in Figure 5.5, $\boldsymbol{T}_j(t)$ is about twice the size of $\gamma C + \boldsymbol{T}_k(t)$. The two consecutive transmit portions of AP k are therefore consolidated into one transmit portion to form an imaginary AP k*, where $\boldsymbol{T}_{k^*}(t_0) = \boldsymbol{T}_k(t_0) + \boldsymbol{T}_k(t_1)$ and the SCAN portion of AP k* is twice that of AP k. Using this approach and assuming that $\boldsymbol{T}_k(t_0) = \boldsymbol{T}_k(t_1) = \boldsymbol{T}_k(t)$, $\boldsymbol{O}_{j,k}(t)$ can be expressed as:

$$\boldsymbol{O}_{j,k}(t) = \begin{cases} \dfrac{\boldsymbol{T}_k(t)}{\gamma C + \boldsymbol{T}_k(t) + \boldsymbol{T}_j(t)} & , \boldsymbol{T}_j(t) \le \gamma C + \boldsymbol{T}_k(t) \\ \dfrac{\boldsymbol{T}_k(t)}{2(\gamma C + \boldsymbol{T}_k(t))} & , \boldsymbol{T}_j(t) \le \gamma C + \boldsymbol{T}_k(t) \end{cases} \tag{5.5}$$

and $\boldsymbol{S}_{j,k}(t)$ can be expressed as:

$$\boldsymbol{S}_{j,k}(t) = \begin{cases} \dfrac{\gamma C}{\gamma C + \boldsymbol{T}_k(t) + \boldsymbol{T}_j(t)} & , \boldsymbol{T}_j(t) \le \gamma C + \boldsymbol{T}_k(t) \\ \dfrac{\gamma C}{2(\gamma C + \boldsymbol{T}_k(t))} & , \boldsymbol{T}_j(t) \le \gamma C + \boldsymbol{T}_k(t) \end{cases} \tag{5.6}$$

The payoff $\boldsymbol{\pi}_{j,k}(t)$ for a single play of the game in (5.3) can be described as follows: $\boldsymbol{G}_j(t)$ is the amount of data transmitted per nmsec and among these data, $\boldsymbol{S}_{j,k}(t)$ is the

portion that will be interference free from AP k while the portion $\boldsymbol{O}_{j,k}(t)$ that coincides with $\boldsymbol{T}_k(t)$ has a probability $\boldsymbol{P}_C(t)$ of being interfered with AP k. The payoff function $\boldsymbol{\pi}_{j,k}(t)$ is plotted in Figure 5.6 where the transmit portion of AP k is fixed at $\boldsymbol{T}_k(t)/M_{MAC}$ = 2 (where M_{MAC} is the maximum MAC frame size). The payoff $\boldsymbol{\pi}_{LI}(t)$ using the LI method (i.e. $\boldsymbol{T}_k(t) = \boldsymbol{T}_j(t)$) is also plotted in Figure 5.6.

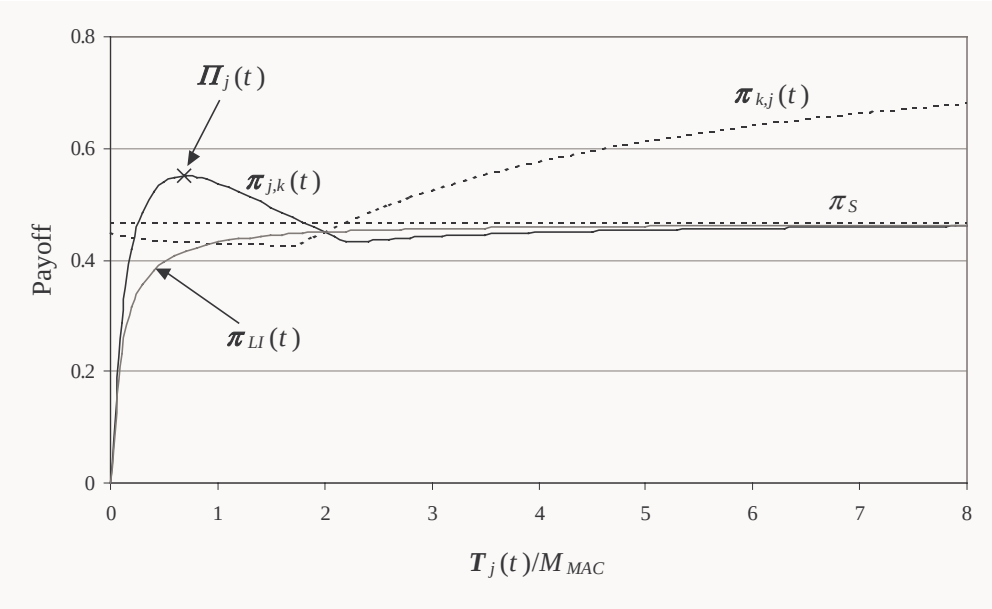

Figure 5.6: Payoff function ($\boldsymbol{T}_k(t)/M_{MAC}$ = 2).

5.2.3 Strategy

The payoff is a function of γ, C, $\boldsymbol{T}_k(t)$ and $\boldsymbol{T}_j(t)$. The total number of channels C and the scan period γ are fixed in this dissertation. Each AP can control its own transmit period by controlling the number of MAC frames F_{MAC} between each successive channel measurement (i.e. SCAN portion of the MAC). In this game, the set of strategies **S** is defined as the number of MAC frames, F_{MAC} for which an AP will use a selected channel after a channel measurement. The set of strategies **S** is thus:

$$\mathbf{S} = \{F_{MAC} \in \mathbf{I} \mid 0 < F_{MAC} < \infty\} \tag{5.7}$$

Where **I** is the set of integer numbers.

5.3 Proposed DCA Using Game Theory (DCA-GT)

In this section, we will use the game defined for a general distributed DCA to introduce the proposed DCA-GT. The payoff for both AP j and AP k are shown in

Figure 5.6 and are identical when $\boldsymbol{T}_k(t) = \boldsymbol{T}_j(t)$, that is when the LI method is used. Using the LI method, the payoff saturates at a value of π_S (shown in Figure 5.6) as the transmit portion increases to infinity. As described in Chapter 4 using a large transmit portion causes the network to react slowly to interference changes and hence it is not desirable. As shown in Figure 5.6, for a fixed value of $\boldsymbol{T}_k(t)$, there is a peak payoff $\boldsymbol{\Pi}_j(t)$ for AP j (i.e. $\boldsymbol{\pi}_{j,k}(t) = \boldsymbol{\Pi}_j(t)$) when $\boldsymbol{T}_j(t) < \boldsymbol{T}_k(t)$. This peak payoff is obtained at the expense of AP k's payoff, i.e. AP k will experience a lower payoff. For every fixed value of $\boldsymbol{T}_k(t) = T_T$, there is a value $\boldsymbol{T}_j(t) = T_L$ such that $\boldsymbol{\pi}_{j,k}(t) = \boldsymbol{\Pi}_j(t)$. Let $\boldsymbol{\Gamma}(T_T)$ be a function that given the length of the transmit portion T_T of an AP (e.g. AP k), will find the corresponding transmit portion T_L of the other AP (e.g. AP j) that will give this AP (e.g. AP j) the peak payoff (i.e. $\boldsymbol{\Gamma}(T_T) = T_L$). The function $\boldsymbol{\Gamma}(T_T)$ can be found by optimising the payoff function in (5.3) for $\boldsymbol{T}_j(t)$. Refer to APPENDIX A for a detailed description of $\boldsymbol{\Gamma}(T_T)$. The function $\boldsymbol{\Gamma}(T_T)$ is shown in Figure 5.7.

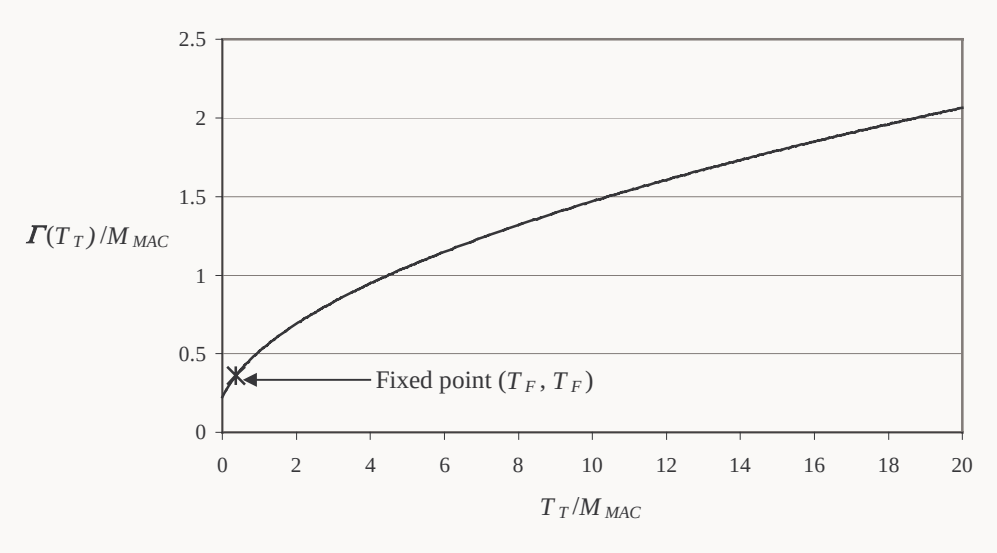

Figure 5.7: Function $\boldsymbol{\Gamma}(T_T)$.

Let $\mathbf{s}_{TX}(x)$ be a strategy where an AP will select $F_{MAC} \in \mathbf{S}$ such that its transmit portion is x nmsec. In a pure strategy, if AP k uses $\mathbf{s}_{TX}(x_1)$, AP j will select strategy $\mathbf{s}_{TX}(x_2)$, where $\boldsymbol{\Gamma}(x_1) = x_2$ (i.e. change $\boldsymbol{T}_j(t)$ to x_2) so that it can achieve the peak payoff. At this strategy vector $(\mathbf{s}_{TX}(x_1), \mathbf{s}_{TX}(x_2))$, AP k will have a lower payoff than AP j and if the game continues and AP k is allowed to change its strategy, it will reduce $\boldsymbol{T}_k(t)$ to $\boldsymbol{\Gamma}(x_2)$ so that AP k now achieves the peak payoff and this will cause AP j to have a lower payoff. The function $\boldsymbol{\Gamma}(x)$ has a fixed point (T_F, T_F) as shown in Figure 5.7 such that

$\Gamma(T_F) = T_F$. Refer to APPENDIX A for a detailed description of the fixed point. Although a fixed point exists, there is no Nash Equilibrium in using a pure strategy. This is because $\Gamma(x)$ serves as the best response $\mathbf{R}(x)$ for strategy $\mathbf{s}_{TX}(x)$ until x is less than a threshold T_S. To see this consider the peak payoff $\Pi_j(t)$ that is achieved when $\mathbf{T}_k(t) = T_T$ and $\mathbf{T}_j(t) = \Gamma(T_T)$ as shown in Figure 5.8. The peak payoff falls below the saturation payoff π_S when T_T decreases below a threshold T_S. Hence $\Gamma(T_T)$ is not the best response when $T_T < T_S$ because at this point the AP can achieve a higher payoff by increasing its transmit portion to a large value to reach π_S. The best response function $\mathbf{R}(T_T) = \Gamma(T_T)$ for $T_T > T_S$ and $\mathbf{R}(T_T) = \infty$ for $T_T < T_S$ for this single game. Since the fixed point T_F is less than T_S, the APs will not settle upon this fixed point and hence there is no Nash Equilibrium in playing a pure strategy.

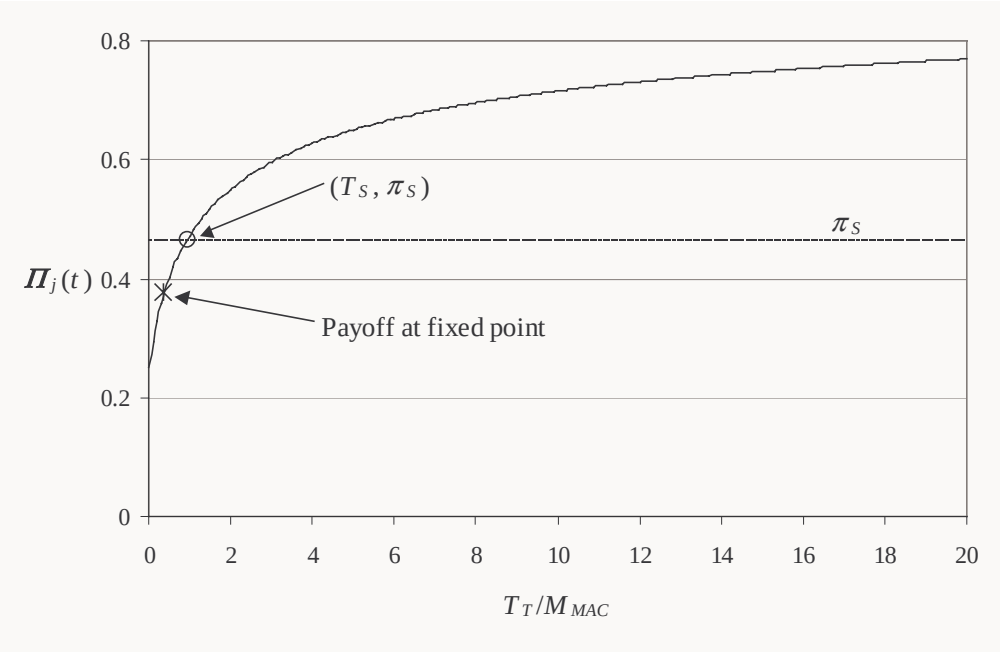

Figure 5.8: Peak payoff $\Pi_j(t)$, $\mathbf{T}_j(t) = T_L$ and $\mathbf{T}_k(t) = T_T$.

5.3.1 Mixed Strategy

The saturation payoff π_S is obtained when the transmit portion of both APs are very large. The LI method is used when both AP's transmit portions are the same (i.e. $\mathbf{T}_j(t) = \mathbf{T}_k(t)$) and the maximum payoff for LI $\pi_{LI}(t)$ is less than π_S as shown in Figure 5.6. Hence, to improve on LI we will focus on the region in Figure 5.8 where $T_T > T_S$, that is where the payoff is larger than $\pi_{LI}(t)$. In this region, the peak payoff is achieved by one AP at the expense of the other AP's payoff. Since there is no stable strategy

vector or Nash Equilibrium in a pure strategy, a mixed strategy is employed. A Nash Equilibrium can exist in a mixed strategy even if none is found in a pure strategy [85].

Instead of a fixed strategy vector (T_T, $\Pi(T_T)$) where $T_T > T_S$, a mixed strategy is introduced so that both APs can take turns to reap the peak payoff. In the proposed DCT-GT method, each AP can select from two strategies s_T and s_L in each play, where strategy $s_T = \mathbf{s}_{TX}(T_T)$ and strategy $s_L = \mathbf{s}_{TX}(T_L)$. Here, $T_L = \Pi(T_T)$ and $T_L < T_T$. In this mixed strategy, each AP will select s_L with probability p_{MIX} and s_T with probability $(1 - p_{MIX})$. Since the strategy vector only has a pair of elements, there are only two strategies for each AP. The payoff function is now just a function of these two strategies and we define $\boldsymbol{U}(m, n)$ as the payoff of an AP (e.g. AP j) when this AP (i.e. AP j) plays strategy m while the other AP (e.g. AP k) plays strategy n, where $m, n \in \{s_L, s_T\}$.

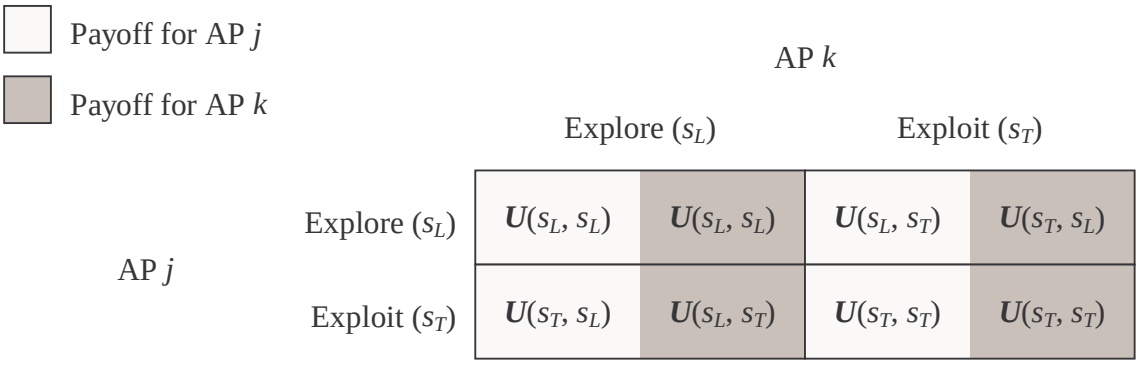

Figure 5.9: Strategic form of game (DCA-GT).

The APs select one of these two strategies independently of each other. In strategy s_L, an AP will spend more time measuring interference power and hence will explore the possible usage of different channels. While in strategy s_T, an AP will spend more time exploiting the channel that it selected in the last measurement period. The degree of channel exploration versus exploitation is set by the probability p_{MIX}. This probability p_{MIX} is derived by first considering the strategic form of the DCA-GT game shown in Figure 5.9.

Each AP in the proposed DCA-GT will use a mixed strategy and the expected payoff Π_{MIX} for each AP playing this mixed strategy is:

$$\begin{aligned} \Pi_{MIX} &= p_{MIX}^2\, \boldsymbol{U}(s_L, s_L) + p_{MIX}(1-p_{MIX})\boldsymbol{U}(s_L, s_T) \\ &+ (1-p_{MIX})p_{MIX}\, \boldsymbol{U}(s_T, s_L) + (1-p_{MIX})^2\, \boldsymbol{U}(s_T, s_T) \end{aligned} \tag{5.8}$$

Optimising (5.8) to find the probability $p_{MIX} = p^*_{MIX}$ that maximises Π_{MIX} leads to:

$$p^*_{MIX} = \frac{2\boldsymbol{U}(s_T, s_T) - \boldsymbol{U}(s_L, s_T) - \boldsymbol{U}(s_T, s_L)}{2\left(\boldsymbol{U}(s_L, s_L) + \boldsymbol{U}(s_T, s_T) - \boldsymbol{U}(s_L, s_T) - \boldsymbol{U}(s_T, s_L)\right)} \tag{5.9}$$

5.4 Simulations and Results

In this section, we will explore the performance of the proposed DCA-GT using simulations modelled in OPNET. The convergence characteristics, the effect of the probability p_{MIX} and the payoff using different sets of (T_T, T_L) are presented. In addition, the SNR performance of DCA-GT is compared with both LI and RND.

5.4.1 Convergence

A scenario similar to that in Section 4.2 is set-up where there are 12 APs all transmitting using the same channel at the beginning of the simulation. The network is said to have converged if all 12 APs use different channels. In LI the transmit portion is fixed, while in DCA-GT it can take two values. The effective transmit portion T_{EFF} in DCA-GT is defined as:

$$T_{EFF} = p^*_{MIX} T_L + \left(1 - p^*_{MIX}\right) T_T \tag{5.10}$$

The time to converge for LI and DCA-GT are plotted as a function of T_{EFF}/M_{MAC} in Figure 5.10. Each point in Figure 5.10 is simulated four times with different seed values and the average result is plotted. In LI, each AP uses a pure strategy, therefore p^*_{MIX} has no meaning and T_{EFF} is the same as the transmit portion of the AP (i.e. $T_{EFF} = T_{TX}$). In Figure 5.10, the LI results are those presented previously in Figure 4.3 but are repeated here for convenience. It is evident from Figure 5.10 that DCA-GT has a much faster rate of convergence compared with LI for the same effective length of transmit portion. This is because the interference detection process in DCA-GT is superior to that in LI.

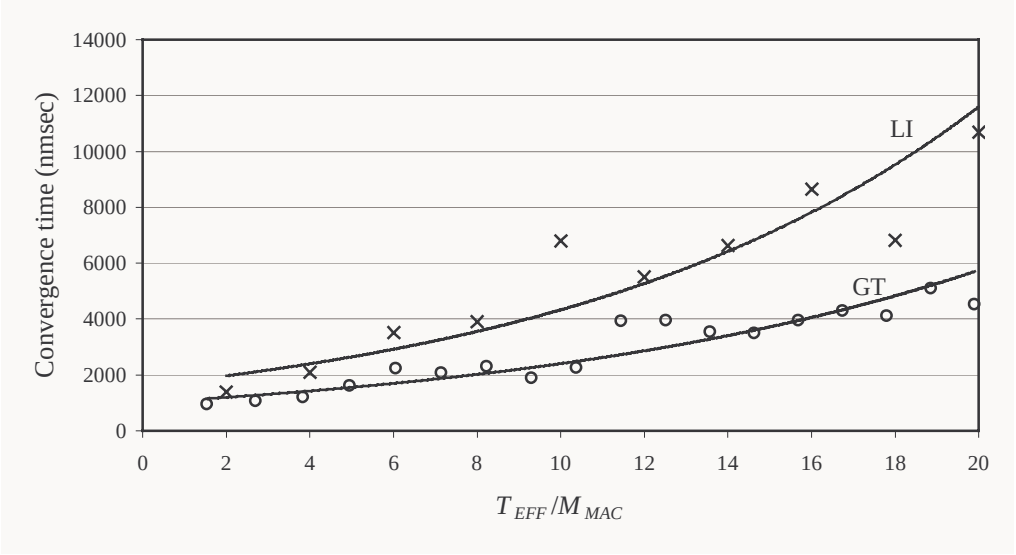

Figure 5.10: Convergence for DCA-GT and LI.

5.4.2 *Probability p_{MIX}*

In this section, we will investigate the effect of varying the probability p_{MIX}. For a fixed set of (T_L, T_T) values, varying the probability p_{MIX} changes the expected payoff Π_{MIX}. For this section, we set T_T/M_{MAC} to 20 and the corresponding value of T_L/M_{MAC} is 2.065, which is obtained from Figure 5.7.

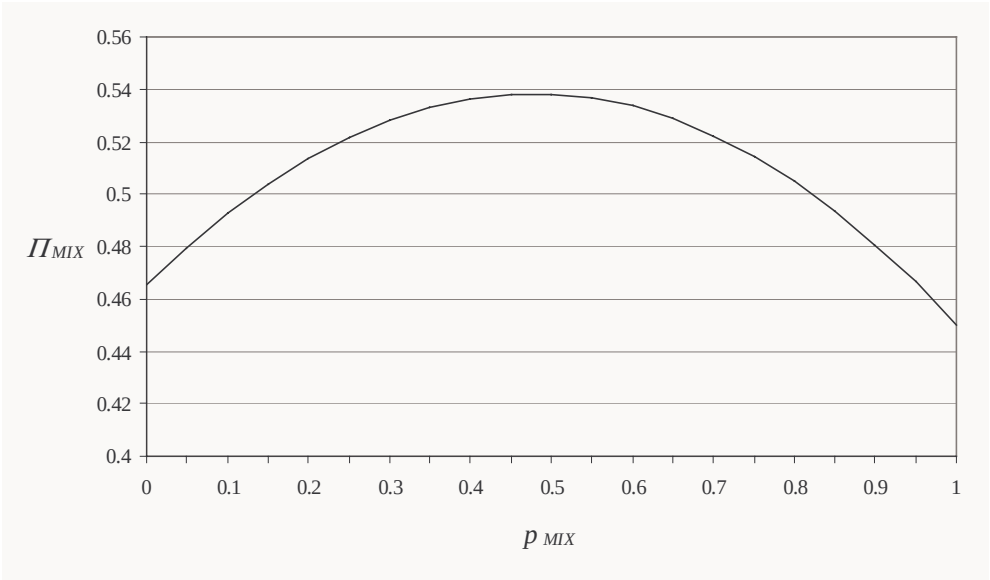

Figure 5.11: Expected payoff Π_{MIX} against p_{MIX} (T_T/M_{MAC} = 20 and T_L/M_{MAC} = 2.065).

The expected payoff Π_{MIX} against p_{MIX} is plotted Figure 5.11, where Π_{MIX} is calculated using (5.8). There is a maximum Π_{MIX} as expected and the two end points where p_{MIX}

= 0 and p_{MIX} = 1.0 are equivalent to the payoff if the LI method is used with the transmit portion set to 20 and 2.065 respectively.

A BFWA network with 50 APs and 277 SUs non-uniformly distributed over 7 cells with the layout shown in Figure 5.12 is constructed in OPNET. The same Random Height propagation model and traffic model as detailed previously in Chapter 3 are used here. At the start of the simulation, all the AP use the same channel and results are taken from the 3 shaded cells shown Figure 5.12. The simulation is run for 36000 nmsec (10 normalised hours). The APs in this simulation use the DCA-GT method where T_L/M_{MAC} = 2.065 and T_T/M_{MAC} = 20, i.e. similar to those values used previously to yield Figure 5.11.

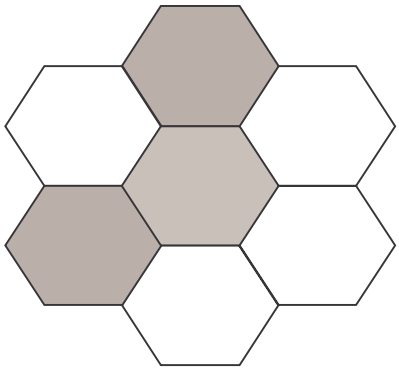

Figure 5.12: 7 cells layout.

This network is simulated for different values of p_{MIX}. The average effective data throughput π_{AVG} as defined in Section 4.4 is used to represent the payoff function.

The average effective data throughput π_{AVG} from the shaded cells in Figure 5.12 is calculated from the results taken during the last two hours (7200 nmsec) of the simulation (to avoid the transient start-up state of the simulation) and it is plotted against p_{MIX} in Figure 5.13. The effective transmit portion T_{EFF} for this DCA-GT scenario when $p_{MIX} = p^*_{MIX}$ = 0.477 is found using (5.10) to be T_{EFF}/M_{MAC} = 11.44. A similar scenario is also simulated where the APs use the LI method with the transmit portion T_{TX} set such that $T_{TX}/M_{MAC} = T_{EFF}/M_{MAC}$ = 11.44 so that data throughput for LI is similar to that in DCA-GT when $p_{MIX} = p^*_{MIX}$. As shown in Figure 5.13, a polynomial fit (the solid line) to the plotted points (i.e. the crosses) shows that the relationship between p_{MIX} and π_{AVG} is similar to that between p_{MIX} and Π_{MIX} shown previously in Figure 5.11. Indeed both curves show a maximum value when p_{MIX} is

about 0.5. LI using T_{TX}/M_{MAC} = 11.44 can be seen to have a lower π_{AVG} compared to DCA-GT when p_{MIX} between 0.1 and 0.7.

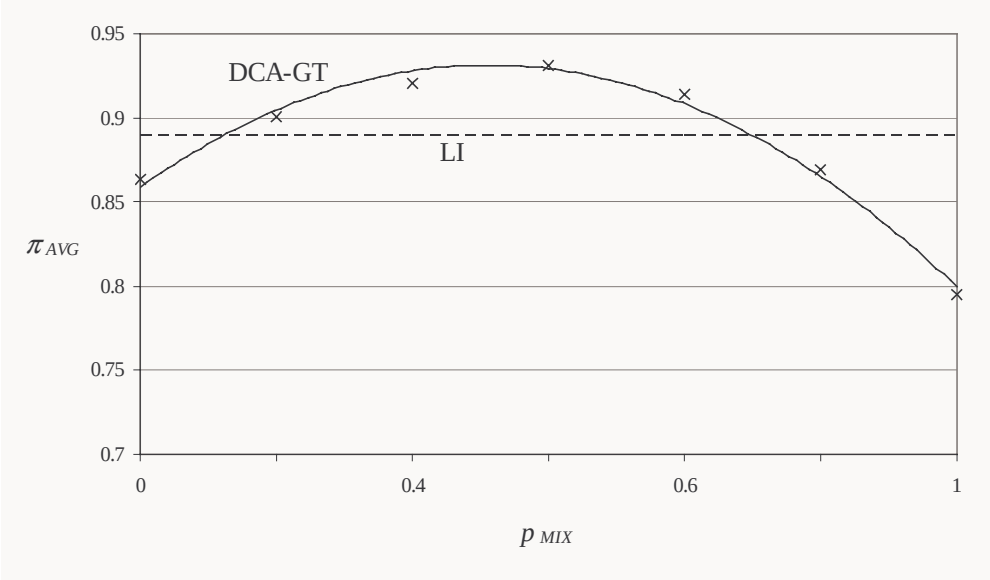

Figure 5.13: π_{AVG} for DCA-GT (T_T/M_{MAC} = 20 and T_L/M_{MAC} = 2.065) and LI (T_{TX}/M_{MAC} = 11.44).

5.4.3 Other Best Reponses

In this section, the relationship between the payoff and different (T_L, T_T) pairs are investigated. As described previously in Section 5.3, each pair (T_L, T_T) corresponds to the peak payoff for the AP with its transmit portion equal to T_L. The expected payoff Π_{MIX} against T_T/M_{MAC} is plotted in Figure 5.14.

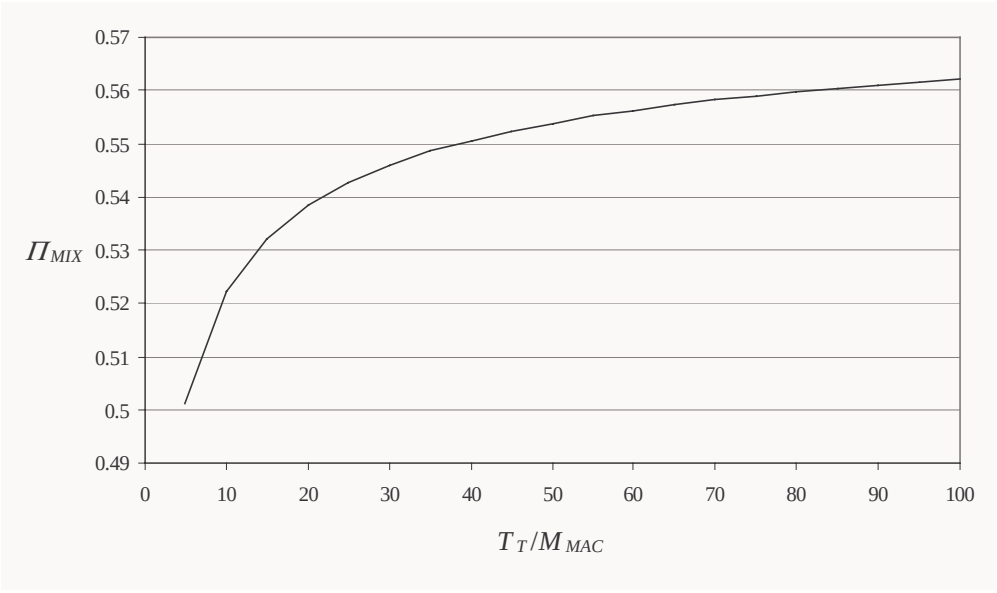

Figure 5.14: Π_{MIX} against T_T/M_{MAC}, where corresponding $T_L = \Gamma(T_T)$.

The value of Π_{MIX} increases as T_T increases but the rate of increase of Π_{MIX} slows markedly at larger values of T_T. The same simulation model used previously in Section 5.4.2 (7 cells, 50 APs and 277 SUs) will again be employed in this section. The AP uses DCA-GT where in each scenario a different value of T_T is used in conjunction with the corresponding value of $T_L = \Gamma(T_T)$ (refer to APPENDIX A) and p^*_{MIX} (obtained using (5.9)). The average effective data throughput π_{AVG} (from the shaded cells in Figure 5.12) is extracted from each simulated scenario.

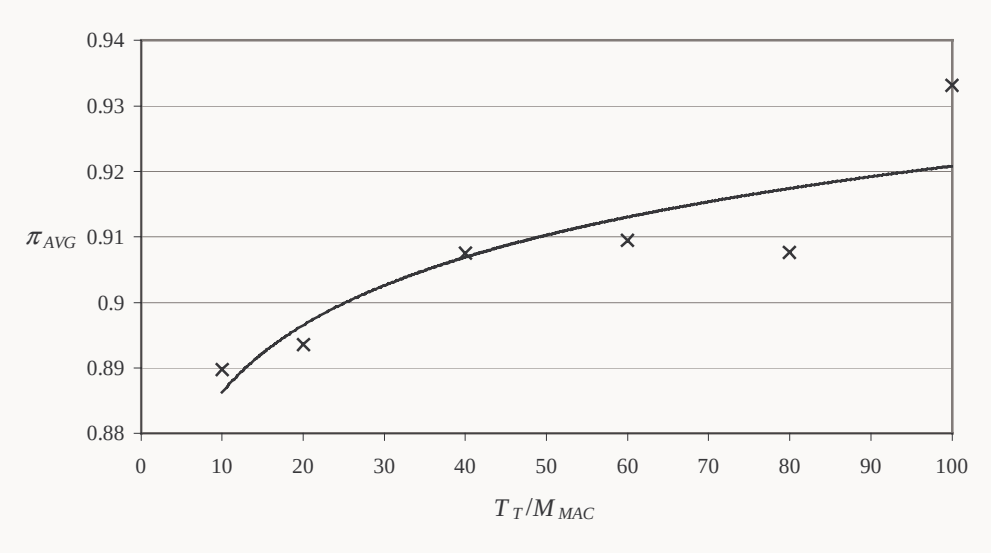

Figure 5.15: π_{AVG} for DCA-GT against different set of (T_L, T_T).

Figure 5.15 is a plot of the simulated results showing π_{AVG} against T_T/M_{MAC}. It can be seen that a similar relationship between π_{AVG} and T_T exists as that shown previously between Π_{MIX} and T_T in Figure 5.14. Again, the rate of increase of π_{AVG} decreases as T_T increases. Hence, in DCA-GT, the payoff obtained by each AP increases with higher values of T_T when the corresponding values of $T_L = \Gamma(T_T)$ and p^*_{MIX} are used. However, the rate of increase of the payoff decreases as T_T increases.

5.4.4 Performance Comparison

In this section, the performance of DCA-GT is compared with that of RND and LI. The same network as that used previously in Section 4.4 is used in this simulation, where 136 APs and 669 SUs are randomly distributed over 37 cells with the layout shown in Figure 4.34. To minimise boundary effects, results are taken from the three shaded cells that are surrounded by other cells and are closer to the centre of the layout. The traffic and propagation models described in Chapter 3 are again used in

this simulation. The DCA-GT method uses T_L/M_{MAC} = 2.065, T_T/M_{MAC} = 20 and p_{MIX} = p^*_{MIX} = 0.477. At the start of each simulation, all the APs use the same channel (i.e. Channel 1). To avoid the unrepresentative poor performance during the transient start up period, the results are taken from the final 2 normalised hours (7200 nmsec) of the simulation.

5.4.4.1 Transmit Portion T_{TX} of LI

As shown previously in Figure 5.6, the payoff for APs using the LI method increases as the transmit portion T_{TX} increases and also that the payoff saturates for T_{TX}/M_{MAC} > 1 (i.e. the rate of increase of the payoff decreases as T_{TX} increases). Although larger values of T_{TX} give a higher payoff, it also causes the network to react slowly to interference changes as shown in the convergence results presented in Figure 5.10 and consequently the use of large values of T_{TX} is not desirable. The probability of interference P_{I_LI} plotted in Figure 4.10 has also been shown to increase as T_{TX} increases. Therefore, the choice of T_{TX} is a balance between achieving high payoff and also delivering a sufficiently rapid response to interference changes.

As described in Chapter 4, it is desirable to have a uniform channel utilisation across the network. There are a total of N = 136 APs in the simulated network. Therefore, for uniform channel utilisation each channel should be in use by an average of $X = N/C$ APs. In this system, C = 15 and this gives $X \approx 9$. For an AP say AP j, the probability $\boldsymbol{Q}_{K,L}(t)$ of exactly K other APs out of L APs (where L is equal to the number of remaining APs i.e., $N - 1$) in the network using the same channel as AP j is:

$$\boldsymbol{Q}_{K,L}(t) = \frac{L!}{K!(L-K)!}\boldsymbol{P}_C(t)^K\left(1-\boldsymbol{P}_C(t)\right)^{(L-K)} \qquad (5.11)$$

Where, $\boldsymbol{P}_C(t)$ is given in (4.5) and "*!*" represents the factorial operation. For a particular channel, the probability that a uniform channel utilisation (i.e. exactly $X = N/C$ APs are using the same channel out of total N APs in the network) is not achieved is $\boldsymbol{P}_{X,N}(t) = (1 - \boldsymbol{Q}_{X\text{-}1,N\text{-}1}(t))$.

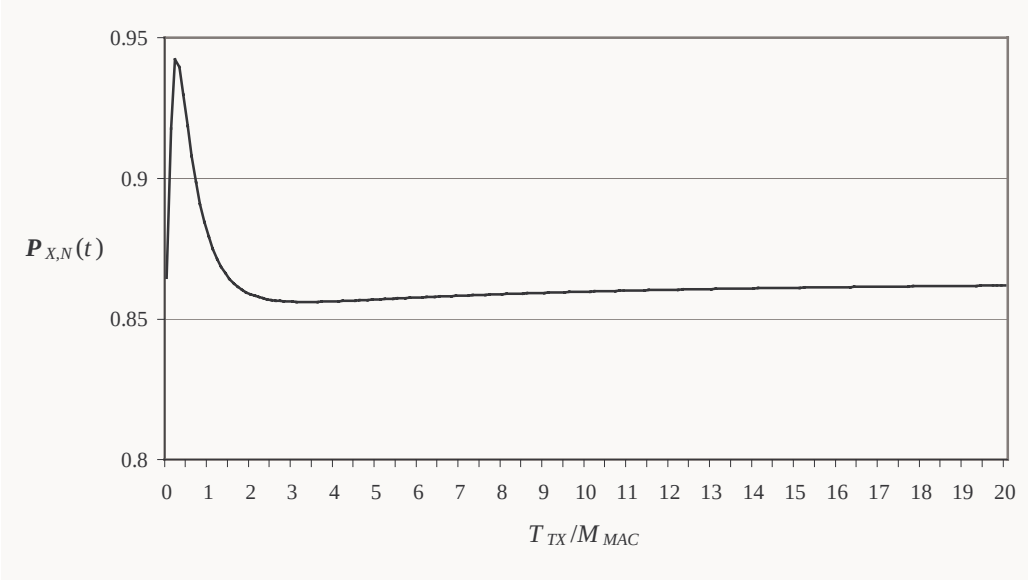

Figure 5.16: Probability of non-uniform channel utilisation (X=9, N=136).

The probability $\boldsymbol{P}_{X,N}(t)$ against T_{TX}/M_{MAC} is plotted in Figure 5.16. The probability $\boldsymbol{P}_{X,N}(t)$ has a minimum when $T_{TX}/M_{MAC} \approx 2$ as shown in Figure 5.16 and saturates in excess of $T_{TX}/M_{MAC} = 2$. In DCA-GT, $T_{EFF}/M_{MAC} = 11.44$ (using $T_L/M_{MAC} = 2.065$, $T_T/M_{MAC} = 20$ and $p^*_{MIX} = 0.477$ in (5.10)) and since the probability $\boldsymbol{P}_{X,N}(t)$ doesn't increase significantly when $T_{TX} = T_{EFF}$ compared to the minimum value of $\boldsymbol{P}_{X,N}(t)$, in this simulation $T_{TX}/M_{MAC} = T_{EFF}/M_{MAC} = 11.44$ is used. Also, when $T_{TX} = T_{EFF}$, the data throughput of DCA-GT and LI are the same. The value of T_{TX} used here for LI is also the same value as that used previously in Section 4.4.

5.4.4.2 Results

The downlink and uplink SNR performances for RND, LI, DCA-GT and FCA-GA are shown in Figure 5.17 and Figure 5.18 respectively. RND has the worst SNR performance in both downlink and uplink directions since RND does not take advantage of any interference or network information. As expected, FCA-GA with global information of the network has the best SNR performance in both downlink and uplink directions. Although LI and DCA-GT both use similar quantities of information, DCA-GT has a better SNR performance than LI in both downlink and uplink directions.

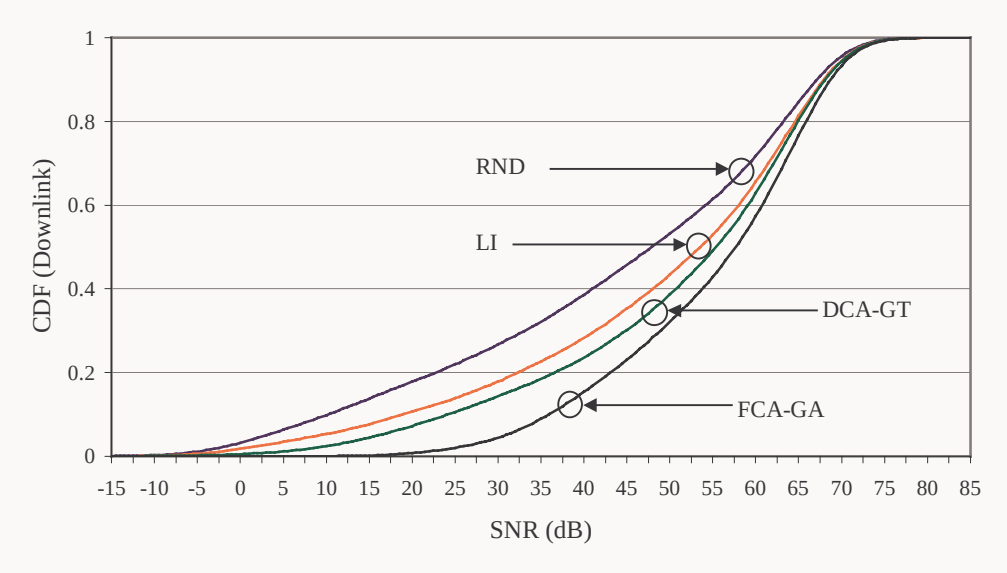

Figure 5.17: Downlink SNR (dB) performance for all shaded cells.

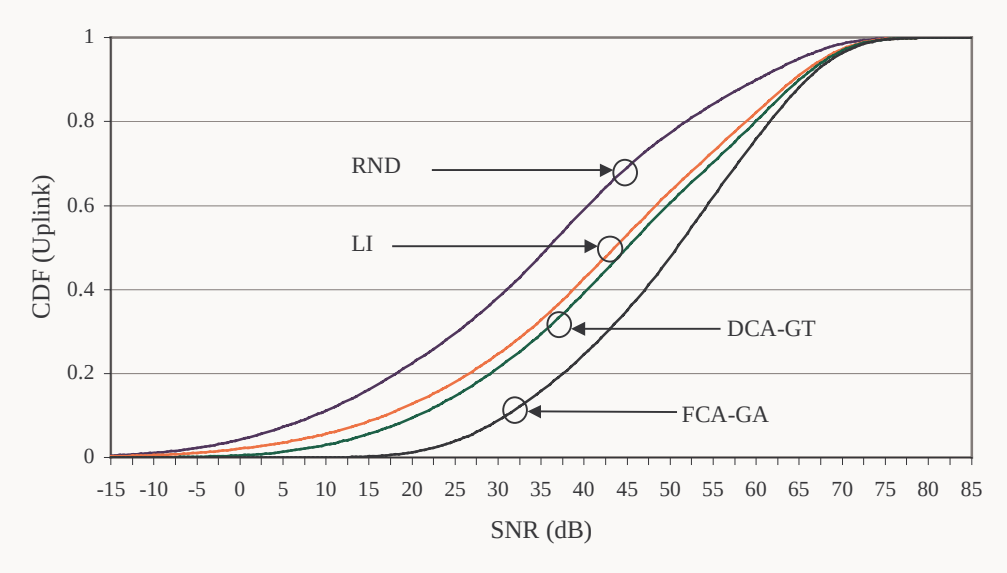

Figure 5.18: Uplink SNR (dB) performance for all shaded cells.

As noted in the results presented previously in Figure 4.37 of Section 4.4, once again the uplink has a poorer SNR performance than the downlink as shown in Figure 5.19. Again, this is because the directional antenna used by the SU is less susceptible to interference compared with the sectored antenna used by the AP.

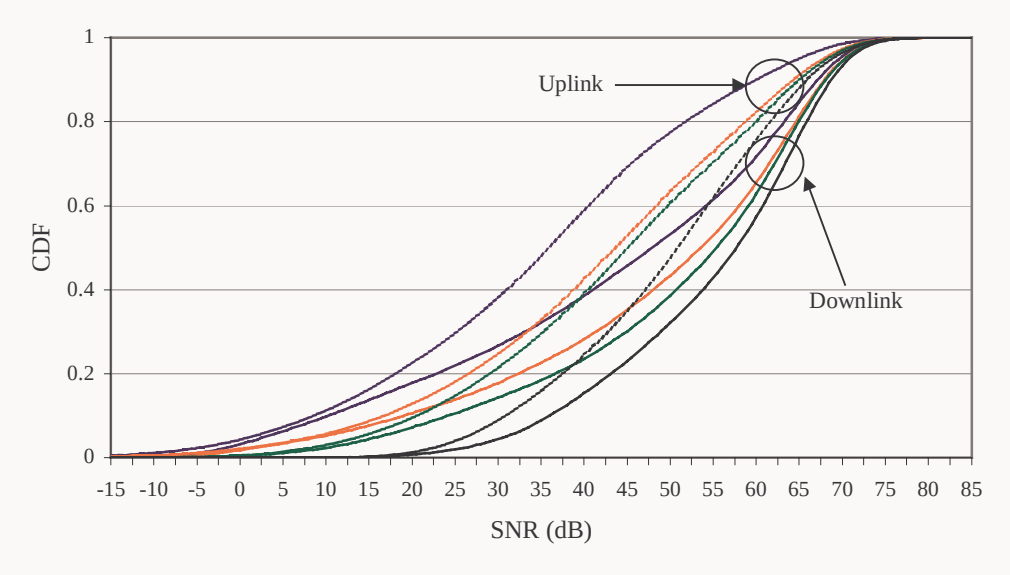

Figure 5.19: Downlink and Uplink SNR (dB) performance.

The first percentile SNR (1% SNR), probability $P_{\geq 21}$ and π_{AVG} are listed in Table 5.1.

	RND	LI	DCA-GT	FCA-GA
1% SNR (downlink)	-5.0 dB	-2.5 dB	4.5 dB	21.5 dB
1% SNR (uplink)	-10.5 dB	-5.5 dB	3.5 dB	19.5 dB
1% SNR (overall)	-8.0 dB	-4.5 dB	4.0 dB	20.0 dB
$P_{\geq 21}$	0.784	0.871	0.905	0.987
π_{AVG}	0.784	0.854	0.891	0.987

Table 5.1: 1% SNR and π_{AVG}.

From Table 5.1, it can be seen that LI has an overall 1% SNR gain of 3.5 dB over RND. DCA-GT using the same amount of measured information as LI has an overall 1% SNR gain of 8.5 dB over LI and 12 dB gain over RND. The best performance is again given by FCA-GA, which has an overall 1% SNR gain of 16 dB over DCA-GT. Although RND does not spend time on measurements, owing to its poor SNR performance it has a lower π_{AVG} compared to both DCA-GT and LI. Although, DCA-GT and LI have the same data throughput, π_{AVG} is higher for DCA-GT than for LI since DCA-GT has a better SNR performance. In a nominal 25 Mbps link, this amounts to a gain of 925 kbps for DCA-GT compared with LI.

The average channel utilisation is plotted in Figure 5.20 for RND, LI and DCA-GT. The average channel utilisation for FCA-GA is fixed and has already been shown previously in Figure 4.38. All three channel allocation methods have a broadly uniform channel utilisation particularly RND. Some channels (e.g. Channel 2 and

Channel 13) in DCA-GT exhibit a slight deviation from uniform channel utilisation. This is because, as the system converges, the AP will settle upon a channel and the channel utilisation is therefore an integer number. Since there are 136 APs in the network and 15 channels, the channel utilisation cannot be divided into equal integers and therefore the optimum channel utilisation is to have 14 channels with 9 APs and 1 channel with 10 APs (to give a total of 136 APs), which is the same channel utilisation used in FCA-GA. Therefore in DCA-GT, Channel 13 is converging towards 10 APs while the remainder of the channels are converging towards 9 APs.

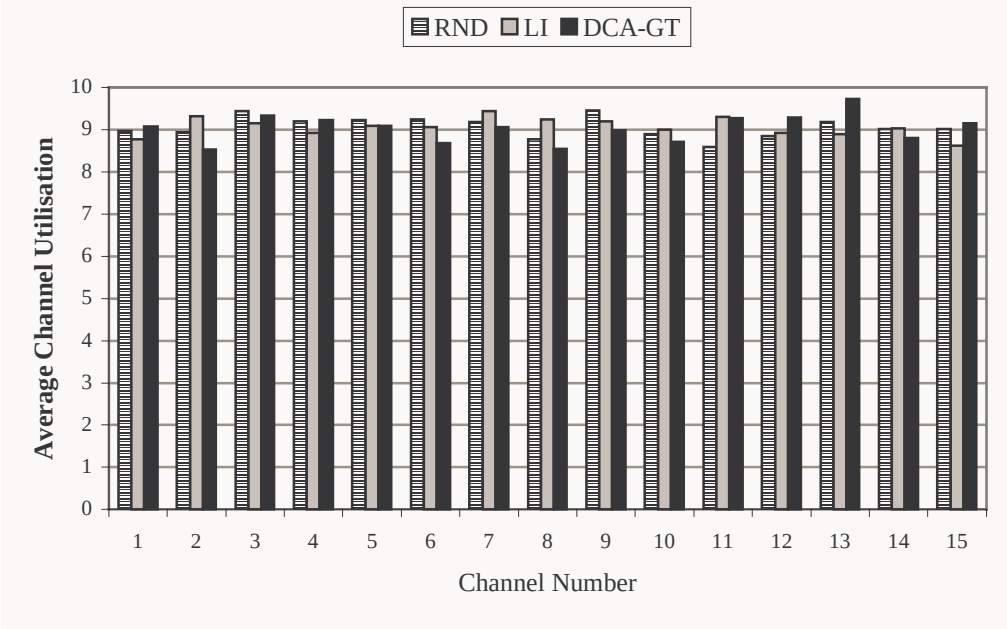

Figure 5.20: Average channel utilisation

The channel fluctuation is plotted in Figure 5.21. In general, RND has a higher channel fluctuation than both LI and DCA-GT since RND changes channel at every MAC frame. The average fluctuation in RND is 2.87 compared to 1.78 and 1.55 in LI and DCA-GT respectively. DCA-GT has a lower average channel fluctuation than LI since DCA-GT is able to settle upon a channel for a longer period. The lower fluctuation in DCA-GT makes channel usage more predictable by the APs and thereby improves the probability of detecting channels with interference.

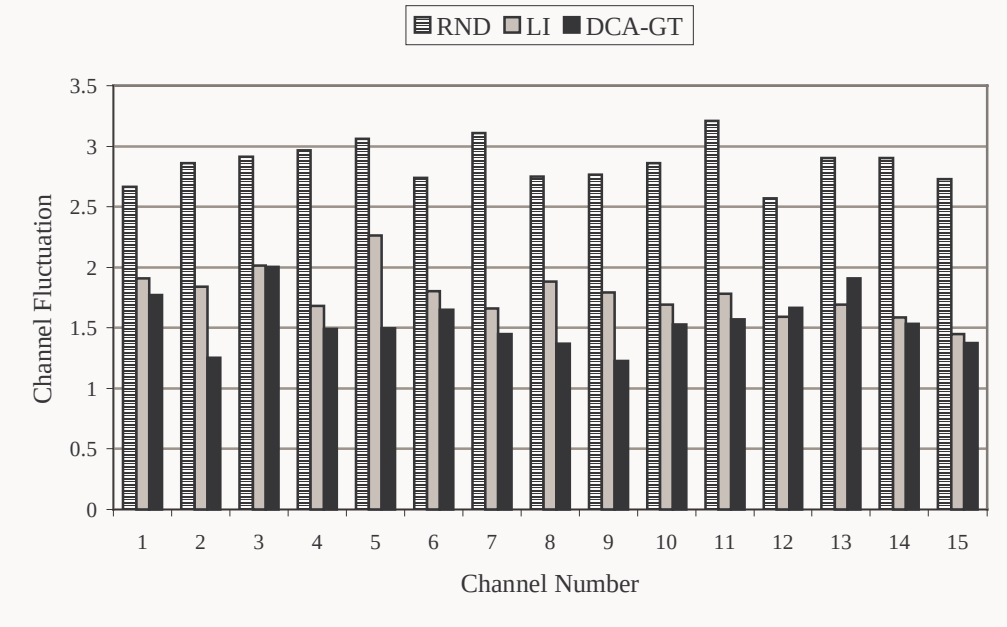

Figure 5.21: Channel fluctuation.

5.5 Conclusion

The interdependency characteristic of Game Theory is exhibited in a BFWA network employing a distributed DCA and consequently the distributed DCA problem can be transformed into a game. A payoff function is defined for a distributed DCA game where the payoff is a function of the AP's data throughput and interference. The payoff defined for this game does not have a Nash Equilibrium if a pure strategy is used.

A new DCA, namely DCA-GT is proposed which uses a mixed strategy where each AP takes turn to obtain the peak payoff such that each AP gets an expected payoff Π_{MIX}. By maximising Π_{MIX}, DCA-GT achieves a balance between exploring different channel usage and exploiting a selected channel. When an AP uses the exploring strategy, it tries to look for a better channel thereby improving SNR. In the exploiting strategy, the AP uses the channel for a longer period thereby achieving higher data throughput and lowering the amount of channel fluctuation so other APs can better predict its channel usage. The average effective data throughput π_{AVG} previously defined in Chapter 4 is found to follow a similar characteristic as the defined expected payoff Π_{MIX}.

The simulation results show that DCA-GT has a 1% SNR performance gain of 8.5 dB over LI even though an identical quantity of measured information is used in both cases. DCA-GT also has a lower average channel fluctuation than does LI. These results show that DCA-GT makes more effective use of information than LI, since DCA-GT achieves a better SNR performance than LI.

6 THRESHOLD AND PRIORITY BASED DCA

In Chapter 5, the LI method and the proposed DCA-GT method have been applied to a BFWA network, however there are two disadvantages with these methods. Firstly, even though these methods select the lowest interfered channel, they do not differentiate whether a selected channel is good or bad. For example, the selected channel may not meet the minimum required SNR threshold. Secondly, these two methods require the AP to measure the interference power of all channels. If there are a large number of channels, these methods may require a significant amount of time to perform the measurements and so considerably reduce the potential data throughput. If the time spent performing measurements is too long, the measured interference and the actual interference experienced by an AP may be significantly different since the channel usage of other APs may have changed during the measurement period.

A threshold based DCA which lies in the Distributed Measurement quadrant of the Channel Allocation Matrix can be used to solve the first problem by ensuring that a selected channel meets a minimum required interference or SNR threshold. To solve the problem of excessive channel measurement, one possibility is for the AP to stop further channel measurements when the first channel that meets the interference threshold requirement is found (e.g. as in First Available DCA [59]). However, as described in Section 2.2.3.3, First Available DCA performs redundant measurements on channels that are highly utilised. This may still cause the AP to measure an excessive number of channels and hence lower the data throughput.

The priority based Distributed Measurement DCA described in 2.2.3.3 can also be used to solve the second problem of having to perform an excessive number of channel measurements. The priority based DCA uses a priority ordered list of channels where each channel is assigned a priority value. The measurement starts from the channel at the top of the list and stops when a channel that meets the required threshold is found. This will reduce the number of measurements that need to be performed when the system has converged to yield a stable list of channels.

This chapter will investigate a Distributed Measurement DCA that uses both a threshold and a priority list for the BFWA system under consideration. A new DCA scheme that utilises a Genetic Algorithm, namely DCA-GA is proposed.

6.1 Channel Segregation for PRMA

Channel Segregation (CS) [60] is one of the most popular Distributed Measurement DCA schemes and uses both a threshold and a priority list. However, it has been investigated previously in the context of voice service in a circuit switched network. For use in the BFWA system under consideration, each AP in the network will maintain a priority ordered list. The AP measures and considers each channel in this list starting with the highest priority channel. The measurement stops when the first channel in the list is found which has an interference power below a predefined threshold T_{INT}. This channel will be used for T_{TX} nmsec (i.e. the transmit portion of the MAC). The priority value for each channel is updated using (2.12) and the channels in the list are resorted according to the updated priority values. In a voice service, if no channel is found that has an interference power less than T_{INT}, the call is blocked. In this BFWA system, if none of the channels in the list has an interference power less than T_{INT}, the lowest interfered channel is selected for use.

The interference threshold T_{INT} is dependent upon the required SNR. In this dissertation, we have defined a packet as being received successfully if its SNR is at least 21 dB. The AP performs the measurement and it will try to ensure that all the uplinks in its cell have a received SNR that meets the required SNR, namely SNR^*. The interference threshold is set such that the SU experiencing the worst SNR is able to meet SNR^*. The worst performing SU is assumed to reside at the edge of the cell since it is the furthest away from its AP. Since the transmit power P_{TX} of the SU is fixed, the received signal power P_{RX} from this SU at the AP can be estimated using the propagation model described in Chapter 3 (ignoring the shadow loss). The receiver's noise n_{RX} at the AP is also fixed and thus the interference threshold T_{INT} can be determined using the following:

$$T_{INT} = \frac{P_{RX}}{SNR^*} - n_{RX} \qquad (6.1)$$

A simulation model with the same arrangement as that described previously in Figure 5.12 (i.e. 7 cells, 50 APs and 277 SUs) is employed to investigate the performance of the CS algorithm. The traffic and propagation models used are again those described previously in Chapter 3. Four scenarios are simulated where in each scenario all the APs use CS for a different value of T_{INT}. The T_{INT} values are selected such that SNR^* is 10 dB, 20 dB, 30 dB and 40 dB and are denoted as CS-10dB, CS-20dB, CS-30dB and CS-40dB respectively. The transmit portion T_{TX} is set at $T_{TX}/M_{MAC} = 2$. At the start of each simulation, the channels are randomly ordered in the priority list for each AP so that the APs will have a different list and will consider different channels at the start of the simulation. RND and FCA-GA are also simulated using the same network layout, where the channel assignment for FCA-GA is taken to be the lowest cost solution (i.e. string) generated from running a Genetic Algorithm with Elitism selection for $p_m = 0.05$, $\mu = 0.1$ and with population size of 200. To be consistent with the previous simulations, all the APs use the same channel (Channel 1) at the beginning of the simulation (except for FCA-GA), which is run for 3600 nmsec. The results are recorded from the APs in the shaded cells of Figure 5.12 for the final 2 normalised hours (7200 nmsec) of the simulation to avoid the start-up transient stage.

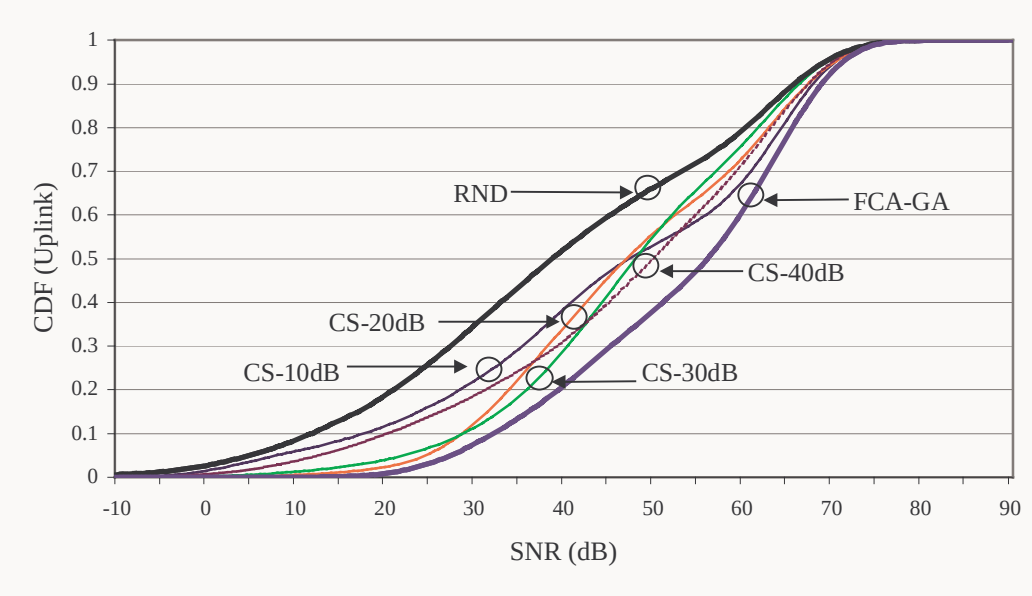

Figure 6.1: Uplink SNR (dB) performance for CS with various thresholds.

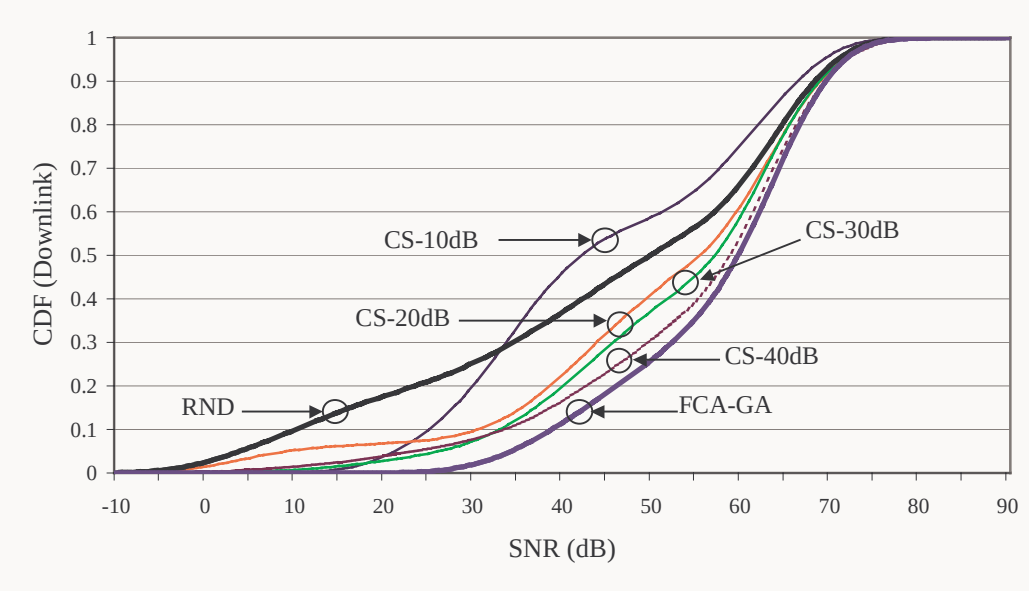

Figure 6.2: Downlink SNR (dB) performance for CS with various thresholds.

The SNR performance of all the APs in the shaded cells of Figure 5.12 are shown in Figure 6.1 and Figure 6.2 for the uplink and downlink directions respectively. The benchmark DCAs, specifically RND and FCA-GA, are also plotted here to permit comparison. In the uplink direction, CS-10dB (i.e. CS with T_{INT} such that SNR^* = 10 dB) has the worst performance for SNR values below 45 dB. Since the SNR^* requirement is low in CS-10dB, most of the channels will be able to satisfy this requirement and consequently the packets received will have a lower SNR. A high SNR^* requirement does not necessarily produce a better SNR performance as shown in Figure 6.1. Hence, CS-40dB has a worse performance than that of CS-20dB and CS-30dB. However, for SNR values above 40 dB, CS-40dB has better SNR performance than that of CS-20dB and CS-30dB. This is because, CS-40dB has a higher concentration of packets with a SNR above 40 dB (about 69%) owing to its high SNR^* requirement compared to those in CS-20dB and CS-30dB. However, owing to its high SNR^* requirement, many channels may not be able to satisfy this requirement and consequently the lowest interfered channel will be selected, which may give a poorer SNR performance since the quality of the channel is now not guaranteed. Similarly, CS-30dB has a poorer SNR performance than CS-20dB at low values of SNR and at SNR values around 30 dB, CS-30dB has a better performance than that of CS-20dB.

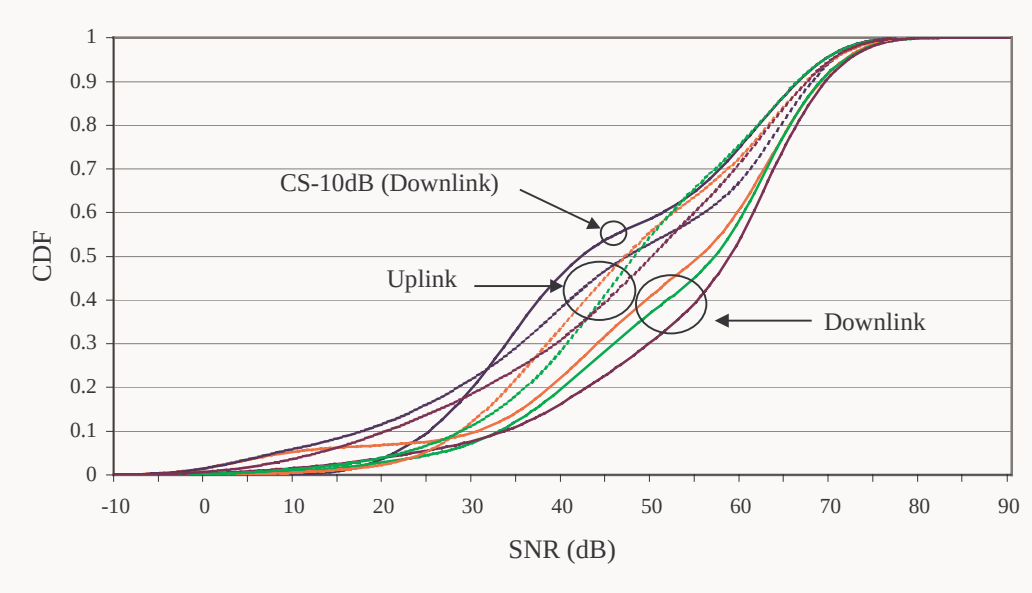

Figure 6.3: Uplink and downlink SNR (dB) performances for CS with various thresholds.

The uplink and downlink SNR performance for CS shown in Figure 6.1 and Figure 6.2 are re-plotted in Figure 6.3. For CS with a higher SNR^* requirement (CS-30dB and CS-40dB), the SNR performance in the downlink direction is better than the SNR performance in the uplink direction, which is consistent with previously observed SNR performances using other channel allocation methods (e.g. Figure 4.37 and Figure 5.19). However, for lower SNR^* requirements (CS-10dB and CS-20dB), the SNR performance in downlink and uplink direction are not consistent. For example, in Figure 6.3 the downlink SNR performance for CS-10dB is worse than that of the uplink for SNR values above 30dB. In the uplink as shown in Figure 6.1, CS-10dB has the worst SNR performance in the mid SNR region (below 45 dB) but it has the best SNR performance for a SNR greater than 55 dB. Meanwhile, in the downlink shown in Figure 6.2, CS-10dB has the best performance for low SNR values (below 19 dB) but for SNRs greater than 30 dB, CS-10dB performance is worse than that of RND. This inconsistency is also found in CS-20dB where as shown in Figure 6.3, the downlink SNR performance at lower SNR values (below 27 dB) is worse than that of the uplink. In the uplink, as shown in Figure 6.1, CS-20dB has the best SNR performance for a SNR below 27 dB. However, in the downlink as shown in Figure 6.2, CS-20dB operating at around the same SNR (below 24 dB), has the worst SNR performance. The unbalanced uplink and downlink SNR performances of CS (specifically CS-20dB) are seen more clearly in the temporal SNR plot in Figure 6.4

(which plots SNR against time) corresponding to all the APs and SUs in the middle cell of Figure 5.12. The AP is able to settle upon a stable channel assignment that gives a good uplink SNR performance. However, the same channel assignment gives a poor downlink SNR performance.

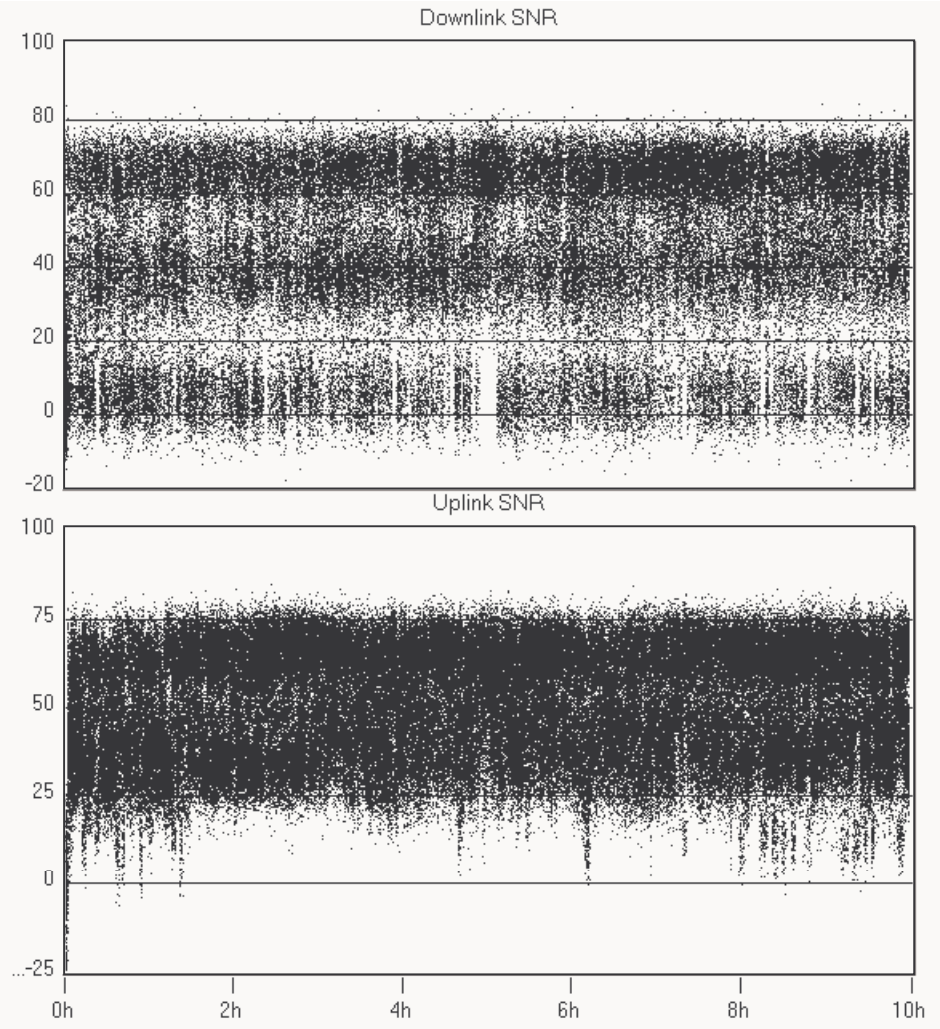

Figure 6.4: Downlink and uplink SNR (dB) temporal plot of CS-20dB for all APs in middle cell.

For CS with a lower SNR^* requirement, the AP needs to perform less channel measurements since it is easier to find a channel that satisfies SNR^*. The decrease in the number of channel measurements reduces the information available on the interference environment. Since the measurement is performed by the AP, the uplink has a closer correlation with the SNR^* (i.e. has a higher concentration of packets with SNR above SNR^*). However, the interference experienced by the AP may not be the same as that experienced by the SU and the lack of information on the interference environment experienced by CS with a lower SNR^* causes the unbalanced uplink and

downlink SNR performance. At higher SNR^* requirements, more channels are measured (making it more like LI) giving it more information on the interference environment thereby leading to a consistent and balanced uplink and downlink performance.

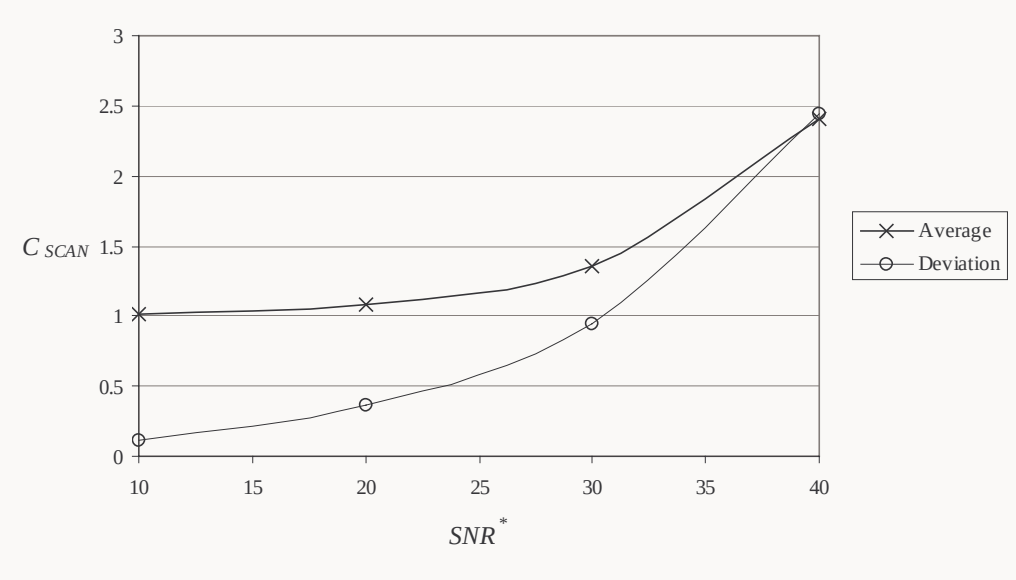

Figure 6.5: Average and standard deviation for C_{SCAN}.

Unlike the situation in LI and DCA-GT, in CS, the number of channels C_{SCAN} measured during each SCAN portion varies. When the system has converged, the AP needs only to measure one channel (C_{SCAN} = 1) and at this point, the channel measurement is performed simply to check (with a period of T_{TX} nmsec) that the channel in use still satisfies the T_{INT} threshold. The average and standard deviation of C_{SCAN} (averaged over all APs in the shaded cells of Figure 5.12) for each scenario are plotted together in Figure 6.5. These values are taken in the second half of the simulation (where the system has passed the initial transient stage). At low values of SNR^* (e.g. CS-10dB), most channels are able to satisfy the SNR requirement and hence the AP will need to perform only one channel measurement most of the time as shown by the corresponding low standard deviation (i.e. most APs have settled upon a channel priority list). As SNR^* increases, the average value of C_{SCAN} also increases as does the standard deviation of C_{SCAN}. This is because at higher SNR requirements, it is harder for the channels to satisfy this requirement and so more channels need to be considered. Also with a higher SNR requirement, it is harder for the AP to settle upon a channel priority list, thereby increasing the standard deviation. The CS algorithm

with SNR^* at infinity i.e. CS-∞dB effectively becomes LI and using the trend in Figure 6.5, it is harder for LI (or equivalently CS-∞dB) to converge to a stable channel assignment than it is for CS with lower values of SNR^*.

For the simulation conducted, it can be seen from Figure 6.5 that the number of channels measured is very low (with a highest average C_{SCAN} value of about 2.5 channels and a deviation of about 2.5 channels) compared to that in LI and DCA-GT (where C_{SCAN} = 15). Therefore, CS will have a much higher data throughput than LI and DCA-GT. However, in CS, particularly CS-20dB, only the AP enjoys the benefits of high effective data throughput while the SUs suffer from poor SNR performance owing to the unbalanced (downlink and uplink) SNR performance. The average effective data throughput π_{AVG}, defined as the fraction of the average data throughput (for all APs in shaded cells) with a received SNR above 21 dB (i.e., SNR_{RX} = 21 dB) per nmsec per AP is plotted in Figure 6.6. It shows that there is an optimum π_{AVG} for CS at a $SNR^* \approx 24$ dB. The optimum SNR^* will change as shown in Figure 6.6 if the required SNR (i.e. SNR_{RX}), for the successful reception of a packet is changed e.g., to 30 dB or 40 dB. This is so because CS-20dB has a lower percentage of packets above 30 dB and 40 dB. Also note that even though CS-40dB has a higher percentage of packets above 40 dB compared to CS-30dB, it has a lower π_{AVG} for SNR_{RX} = 40 dB. This is because CS-40dB has a higher average C_{SCAN} (leading to lower data throughput) compared to that of CS-30dB.

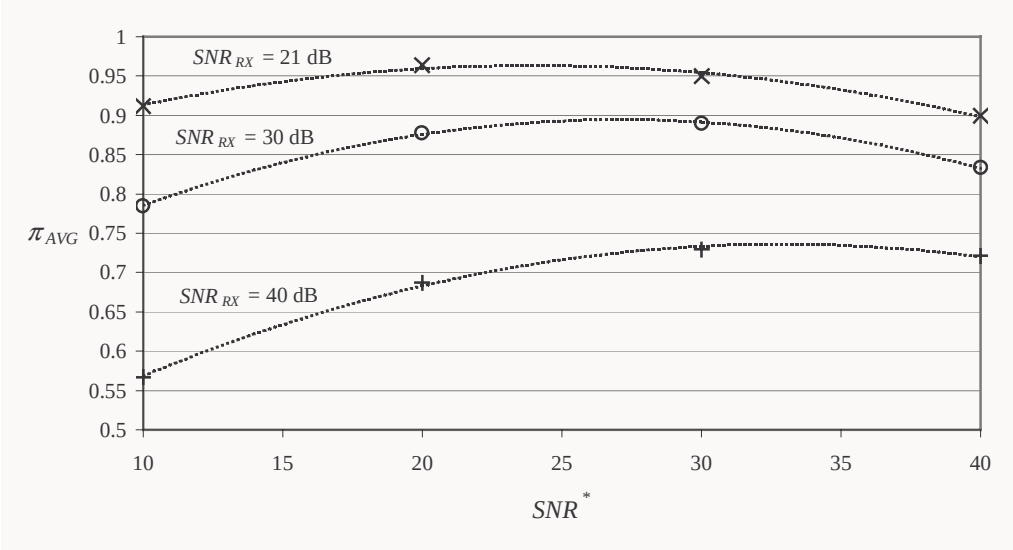

Figure 6.6: π_{AVG} against SNR^* with different values of SNR_{RX}.

The average channel utilisation is recorded and is plotted in Figure 6.7 for results obtained during the final 2 normalised hours of simulation (i.e. after the system has passed the transient stage). It is observed that as SNR^* increases the channel utilisation becomes increasingly uniform. CS-10dB has a very non-uniform channel utilisation where some channels are lightly utilised (e.g. Channel 12 is used by only one AP) while others are highly utilised (e.g. Channel 9). The AP that uses the lowly utilised channel will experience a very high SNR while those APs that use a highly utilised channel will experience a low SNR. As described previously, non-uniform channel utilisation is a result of unfair channel allocation where an AP obtains a high SNR at the expense of other APs. CS-20dB also has a non-uniform channel utilisation giving an unfair channel allocation. As SNR^* increases, the algorithm tends towards LI and the channel utilisation becomes increasingly uniform.

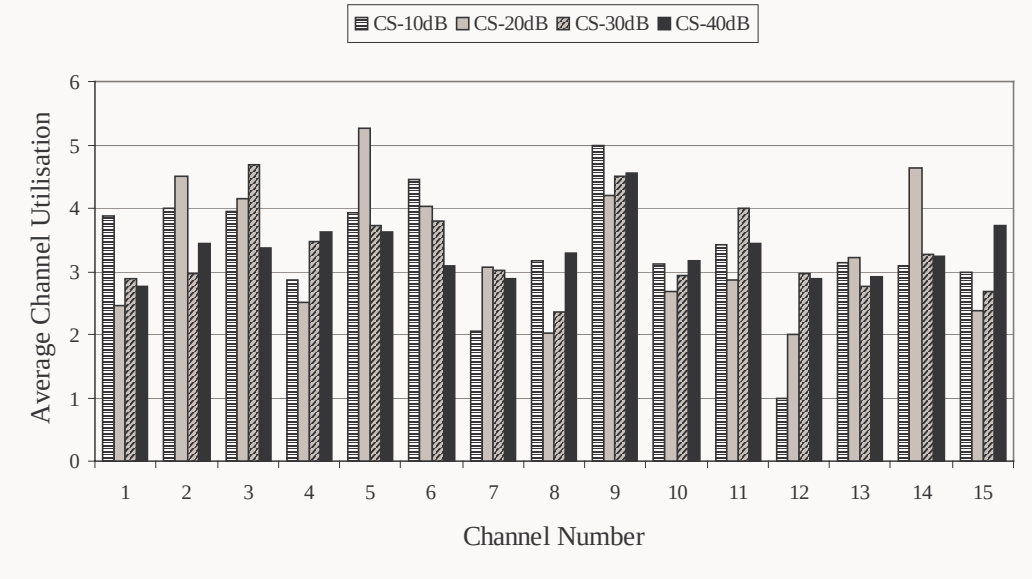

Figure 6.7: Average channel utilisation for CS.

The channel fluctuation is plotted in Figure 6.8. In general channel fluctuation increases as SNR^* increases as shown in Figure 6.9. From Figure 6.8 it can be seen that CS-10dB is able to settle upon particular channels e.g. Channel 2, Channel 9 and Channel 12, where there is zero fluctuation. Since the SNR* is low in CS-10dB, the AP can find a channel that meets the threshold upon the first channel measurement and this will make the priority value of this channel higher for the following SCANs. Owing to this, the AP will be more likely to settle upon a bad channel (e.g. a highly utilised channel like Channel 9) that has high interference. The network in CS-10dB

is highly converged due to the low channel fluctuation and it is difficult for the APs that are using a bad channel (e.g. Channel 9) to change to another channel experiencing lower interference since these APs do not consider other channels. CS-20dB also has low channel fluctuation and is also highly converged. Here most of the APs have settled upon a channel and a priority list (particularly the AP using Channel 12 that has no fluctuation). The channel fluctuation is higher at higher SNR^* where most APs have not settled upon a channel since it is harder for the system to find a suitable channel. As described previously, a network with high channel fluctuation makes it difficult for the APs to predict each other's channel usage and this makes it hard for the network to converge to a stable channel allocation.

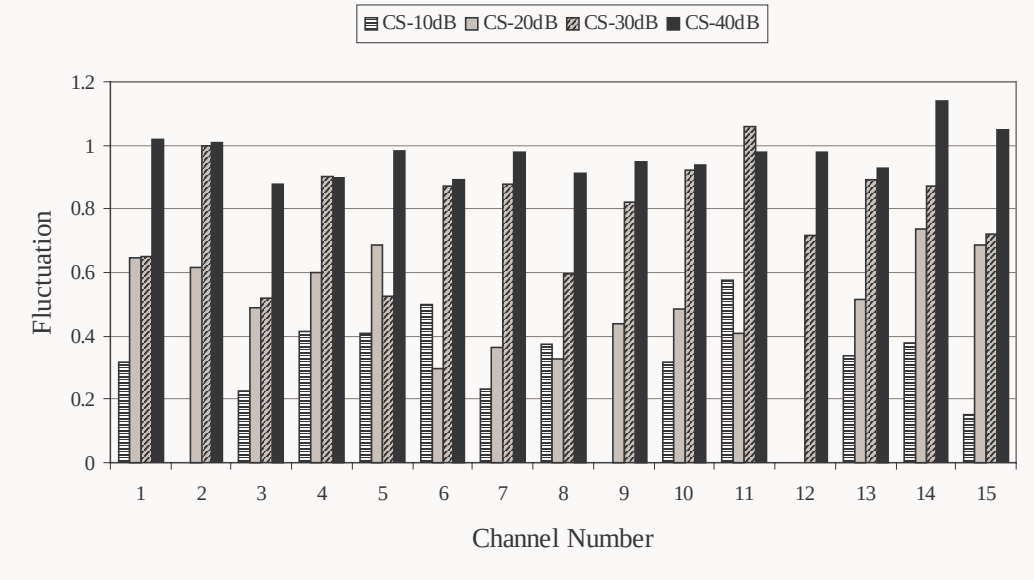

Figure 6.8: Channel Fluctuation

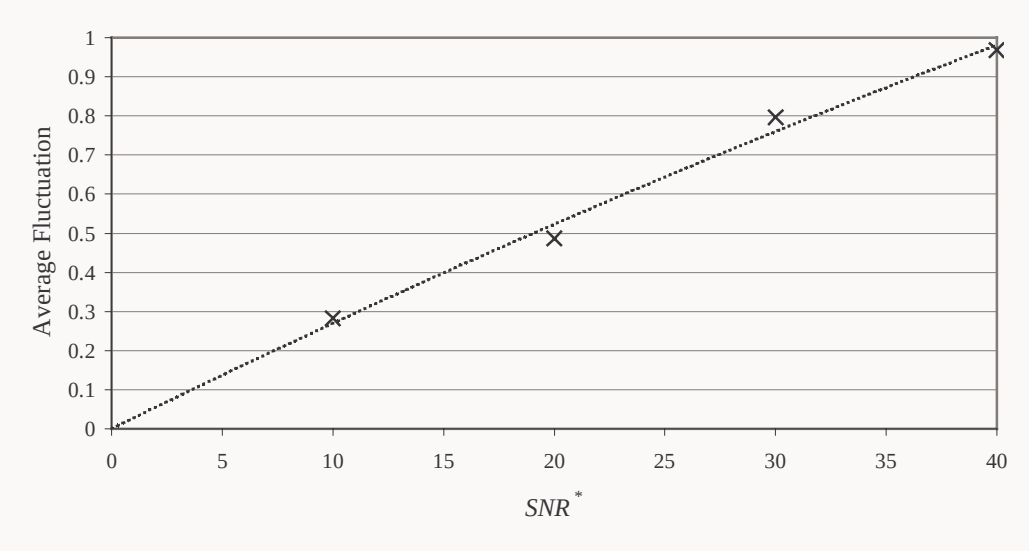

Figure 6.9: Average channel fluctuation.

CS schemes with a low SNR^* (i.e. a high threshold, T_{INT}) have a greater chance of finding a suitable channel and hence cause the network to converge quickly. When a network has converged, the data throughput increases as the number of channel measurement per SCAN, C_{SCAN}, is reduced to one. However, low values of SNR^* give rise to non-uniform channel utilisation, which leads to unfair channel allocation and unbalanced (uplink and downlink) SNR performance as the network converges too quickly to a poor local optimum solution. On the other hand, CS with high SNR^* value has a lower chance of finding a suitable channel and therefore converges more slowly. However, it has more information on the interference environment than CS with low SNR^* and hence it has a more balanced (uplink and downlink) SNR performance and so tends to a uniform channel utilisation leading to a fairer channel allocation.

6.2 Partially Centralised DCA

As shown previously, LI and DCA-GT are able to achieve uniform channel utilisation, however they require extensive channel measurements, which reduces their data throughput. CS using both a threshold and a priority list is able to reduce the amount of channel measurement giving high data throughput, however the lack of information concerning the interference environment leads to non-uniform channel utilisation in the network and an unbalanced (uplink and downlink) SNR performance.

In this section a new DCA method, namely DCA-Genetic Algorithm (DCA-GA) is introduced that reduces the quantity of channel measurements and yet minimises the performance penalty by making use of centralised information to compensate for the lower level of information concerning the interference environment (owing to the reduced quantity of channel measurements). The degree of centralisation has been defined previously to be proportional to the number of APs required to communicate with a Control Server (CSVR) in order to allocate a channel. A CSVR is connected to N_c APs and these APs form a cluster (i.e. the cluster size is N_c). Due to the non-scalability of a fully centralised system, DCA-GA will employ a partially centralised system. The degree of centralisation in a partially centralised system is different to that of the concept of centralisation up to the interference neighbourhood described previously in Section 2.2.2.1. In a partially centralised system, the clusters are distinct whereby the APs communicate with only one CSVR (the one that the AP is connected to). This is in contrast to a system with centralisation up to the interference neighbourhood because apart from communicating with the CSVR that the AP is connected to, the AP also has to communicate with other CSVRs whose APs' interference neighbourhood includes this AP. Using a partially centralised system, intra-cluster communication (e.g. from CSVR to CSVR) where information from an AP has to go all the way to a global server (to which all CSVRs are connected) is avoided. In addition, partial centralisation does not require each AP to define its interference neighbours.

6.2.1 DCA Using Genetic Algorithm, DCA-GA

The proposed partially centralised DCA-GA uses both a threshold and a priority list and has been introduced in [89] and [90]. In a similar manner to CS, each AP in the proposed DCA-GA maintains a channel priority list. The list is maintained using two subsystems. The first subsystem is responsible for measuring and selecting a channel while the second subsystem is responsible for ordering the channels in the priority list.

The first subsystem resides at the APs. The priority list in each AP contains C_{GA} channels, where C_{GA} is less than the total number of available channels C. The AP measures all C_{GA} channels in the priority list and selects the highest priority channel on the basis that its interference power is below a threshold T_{INT}. If all the measured channels have an interference power above T_{INT}, the channel having the lowest

interference within the priority list is selected. The measured interference power of the channels is sent to the CSVR.

The second subsystem is located at the CSVR, which is responsible for selecting C_{GA} out of C channels and ranking them for each of the N_c APs in its cluster. The priority list is formed based on the channel usage of the APs in the cluster and the channels' interference power as measured by each AP. This is a large combinational problem and as the size of C_{GA}, C and N_c increase, the number of possibilities increases exponentially. The optimisation technique chosen to form this list is the Genetic Algorithm.

In most centralised systems, the AP needs to wait for a response from the central controller prior to a channel selection. This delay may be significant in a system with a high degree of centralisation owing to extensive signalling and processing at the central controller. The larger the network, the greater will be the delay. In the DCA-GA, the first and second subsystems operate asynchronously and the delay inherent in a conventional centralised system is avoided. This is because the AP does not need to wait for a response from the CSVR since the AP is able to make a channel allocation decision using the existing priority list at hand. The CSVR receives interference power measurement updates from the APs and performs the optimisation using a Genetic Algorithm once it has received the latest updates from all its APs. The first subsystem located at the APs is a fully distributed system, while the second system is centralised and hence the system is able to utilise centralised information without sacrificing the low delay characteristics of a fully distributed system.

The Genetic Algorithm performed at the CSVR follows the same processes as those listed in Figure 2.9 but at the end of each generation, the CSVR updates each AP (in its cluster) with a new channel priority list. The Genetic Algorithm is performed during network operation rather than performed off line as in the previously described FCA-GA. The population size $|\mathbf{P}(t)|$ is fixed at all generations. An individual (i.e. a possible solution) is represented by a string using integer encoding, which is a concatenation of each AP's channel priority list in the cluster. Each element in the string is a channel number. Since the AP measures all the channels in its priority list, the channel numbers in each priority list are different to avoid measuring a specific channel more than once. An example of an individual's string is shown in Figure

6.10, where it is assumed that the CSVR has 3 APs (i.e. N_c = 3) and each AP's priority list has 5 channels (i.e. C_{GA} = 5). For each AP, the channel priority decreases from left to right, for example for AP 2 in Figure 6.10, channel number 6 has the highest priority while channel number 2 has the lowest priority in its list.

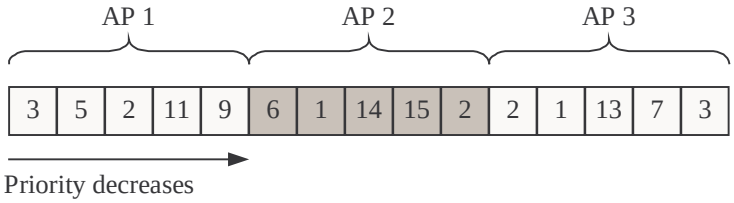

Figure 6.10: String representation for DCA-GA.

Each individual is assigned a fitness value. The fitness value is a measure of how good the channel priority order list is in the cluster. The fitness value is calculated using a fitness function $\boldsymbol{F}_{DCA}(t)$ for generation t.

Firstly, in a specific generation t, the fitness $\boldsymbol{\Phi}_j(t)$ for each priority order j ($1 \le j \le C_{GA}$) in the channel priority list is calculated. For example, priority order j = 1 is the channel number at the top of each AP's channel priority list. For the string in the example in Figure 6.10, priority order 1 includes channel numbers 3, 6 and 2, which are the highest priority channels for AP 1, AP 2 and AP 3 respectively. The function $\boldsymbol{\Phi}_j(t)$ is given by:

$$\boldsymbol{\Phi}_j(t) = K_I \sum_{m=1}^{N_c} \boldsymbol{I}_{j,m}(t) + K_D \sum_{m=1}^{N_c} \sum_{n=m}^{N_c} \boldsymbol{D}_{j,m,n}(t) + K_S \sum_{m=1}^{N_c} \boldsymbol{S}_{j,m}(t, t-1) \qquad (6.2)$$

where, $\boldsymbol{I}_{j,m}(t)$ is the averaged measured interference power in dBm for the j^{th} priority channel of AP m in the t^{th} generation. The interference power for each channel is averaged over the past 3 generations to reduce the effect of fluctuations in the measurement. As described previously using a threshold T_{INT} that is too low may prevent an AP from selecting a better channel. Hence, function $\boldsymbol{I}_{j,m}(t)$ will try to identify a channel that has a lower interference power (rather than one that marginally satisfies the T_{INT} requirement) and achieves a balanced downlink and uplink SNR performance. Function $\boldsymbol{D}_{j,m,n}(t)$ is 1 if the j^{th} priority channels for AP m and AP n are different at the t^{th} generation, otherwise $\boldsymbol{D}_{j,m,n}(t)$ is zero. This function is to encourage

the APs in the same cluster to use different channels. Function $\boldsymbol{S}_{j,m}(t, t\text{-}1)$ is 1 if the channel number in the j^{th} priority of AP m at generation t and t-1 are the same, otherwise $\boldsymbol{S}_{j,m}(t, t\text{-}1)$ is zero. This is to encourage each AP to use the previous channel. K_I, K_D and K_S are the constants that weight the influence of functions $\boldsymbol{I}_{j,m}(t)$, $\boldsymbol{D}_{j,m,n}(t)$ and $\boldsymbol{S}_{j,m}(t, t\text{-}1)$ respectively. Constant K_I is a real negative number (because $\boldsymbol{I}_{j,m}(t)$ is measured in dBm) since it is preferable to have a low interference power. The function $\boldsymbol{D}_{j,m,n}(t)$ is to avoid the problem of trying to design the string so that the APs will always use different channels in the same cluster. This is because the number of APs may exceed total number of available channels (i.e. $N_c > C$) and consequently such a string would be difficult to implement. Hence, a high value of K_D is used to ensure that each AP in the same cluster uses a different channel and in the case where $N_c > C$, it will ensure that the channels are utilised as uniformly as possible. The constant K_S determines whether the algorithm will explore different channel combinations or will exploit the current channel combinations. When the algorithm is exploiting the current channel combinations (i.e. a large value of K_S is used) then the channels used for transmission will not fluctuate. This enables APs associated with other CSVRs to better predict the channel usage of other clusters. However, the network will react slower to interference changes since a large value of K_s effectively makes the AP less likely to change channel. On the other hand, if the algorithm is set-up to explore different combinations (i.e. a small value of K_S is used), this may yield a better SNR performance. However, small values of K_S will cause the transmission channel to fluctuate and so operation will become less predictable giving rise to instability. The constant K_S should be set to balance between channel exploration and exploitation.

The overall fitness $\boldsymbol{F}_{DCA}(t)$ for an individual at the t^{th} generation is:

$$\boldsymbol{F}_{DCA}(t) = \sum_{j=1}^{C_{GA}} W_j \, \boldsymbol{\Phi}_j(t) \tag{6.3}$$

where, the constant W_j weights the importance of the j^{th} priority channel in the list. A higher priority channel will be more likely to be used than a lower priority channel and hence $W_i \geq W_k$ for $k > i$. The fitness function calculation for a string in the example given in Figure 6.10 is shown in Figure 6.11.

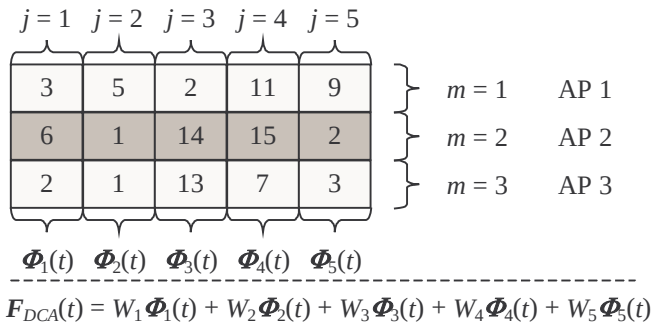

Figure 6.11: Fitness function $\mathbf{F}_{DCA}(t)$ calculation.

Elitism Selection, which gave the best results in the FCA-GA investigation presented in Chapter 4, is used in the proposed DCA-GA. In [16], Elitism Selection is suggested for use in a dynamic environment, which is appropriate for the interference environment experienced in a BFWA network. The elite fraction μ is set to 0.1, while the population size $|\mathbf{P}(t)|$ is set to 200.

An invalid string occurs if one of the AP's priority lists has a duplicate channel number. If this situation occurs, it will cause an AP to measure this duplicate channel more than once. To avoid this, the partially mapped crossover [84] method used previously in FCA-GA is also employed here.

Mutation is performed by randomly selecting an element in each AP's list and changing it to another randomly selected channel. If the randomly selected channel would give rise to an invalid string then the position of this element is swapped with that of the element that it would have been identical with. For example, consider the string of AP 1 in Figure 6.11 with elements {3, 5, 2, 11, 9}. If the 2nd element (with channel number 5) is mutated and the randomly selected channel is 9, which is identical to the 5th element, the 2nd and 5th elements are swapped to produce a string with elements {3, 9, 2, 11, 5}.

At the end of each generation, the channel priority lists for each AP are taken from the fittest individual's string and are sent to update the respective APs. Since the mutation rate p_m in Elitism Selection does not affect the fittest individual as demonstrated previously in Figure 4.24, a high mutation rate of 0.1 is used so that the system can react to changes more rapidly following convergence (since mutation introduces diversity into the population).

6.2.2 Channels per SCAN and Priority Weights

In the previous sections (notably Section 4.4), a packet is deemed to have been received successfully if its SNR is at least 21 dB. Hence the required interference threshold T_{INT} in DCA-GA is set such that SNR^* is 21 dB.

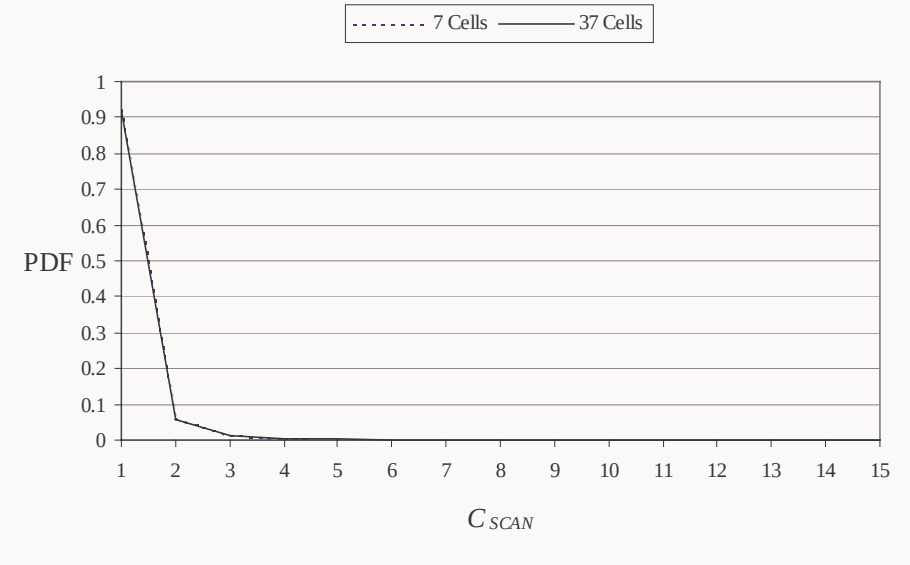

Figure 6.12: Probability Distribution Function (PDF) of C_{SCAN} in CS.

Figure 6.12 is the Probability Distribution Function (PDF) of the number of channels per SCAN, C_{SCAN} used by CS with a SNR^* value of 21 dB. Two scenarios are simulated one with 7 Cells (50 APs and 277 SUs) and the other with 37 Cells (136 APs and 669 SUs). The PDF yielded by both scenarios are similar and in both scenarios, the probability of using more than 5 channels per SCAN is less than 0.001. From the results presented previously in Figure 6.5, at a high SNR^* value of 40 dB, the average required C_{SCAN} is 2.5 channels with a standard deviation of about 2.5 channels. The system therefore is required to scan up to 5 channels in order to satisfy a high SNR^* value of 40 dB. Therefore, in DCA-GA, the number of channels C_{GA} in the priority list is set to 5 (where the total number of channels C is 15).

The priority weights W_j in (6.3) are proportional to the probability of each priority channel being used. A channel will be used if it is the first channel that meets the interference power threshold T_{INT} requirement (i.e. where all channels with a higher priority than this channel fail to meet this requirement). Therefore, the priority weight W_j is set to be approximately the probability that $C_{SCAN} = j$. These probabilities are

given in Figure 6.12 yielding the priority weight values of W_1 = 0.9, W_2 = 0.05, W_3 = 0.025, W_4 = 0.0125 and W_5 = 0.0125.

6.2.3 Cluster Size, N_c in DCA-GA

The effect of different cluster sizes, N_c is investigated for DCA-GA using simulations performed in OPNET. The same 7 cell layout as that used previously with 50 APs and 277 SUs (shown in Figure 5.12) is again employed. The traffic and propagation models are those described in Chapter 3. Seven scenarios are simulated where for each scenario the number of CSVRs serving the entire network is different. The average cluster size $\tilde{N}_c$ is the average number of APs per CSVR. The number of CSVRs employed is set to 1, 2, 3, 4, 5, 10 and 25 for each of the scenarios, which corresponds to an average cluster size $\tilde{N}_c$ of 50, 25, 16.67, 12.5, 10, 5 and 2 respectively. In the case where number of CSVR is 1, the DCA-GA becomes a fully centralised system. The constants K_I, K_D and K_S of (6.2) are set such that the importance of $\boldsymbol{I}_{j,m}(t)$ and $\boldsymbol{S}_{j,m}(t, t\text{-}1)$ are equal while the importance of $\boldsymbol{D}_{j,m,n}(t)$ is twice that of $\boldsymbol{I}_{j,m}(t)$ (or $\boldsymbol{S}_{j,m}(t, t\text{-}1)$). A population size of 200 is used. The transmit portion T_{TX} is set such that T_{TX}/M_{MAC} = 2. As with the previous simulations, all the APs use the same channel (Channel 1) at the start of the simulation and the simulation is run for 3600 nmsec. The results are taken from the 3 shaded cells of Figure 5.12.

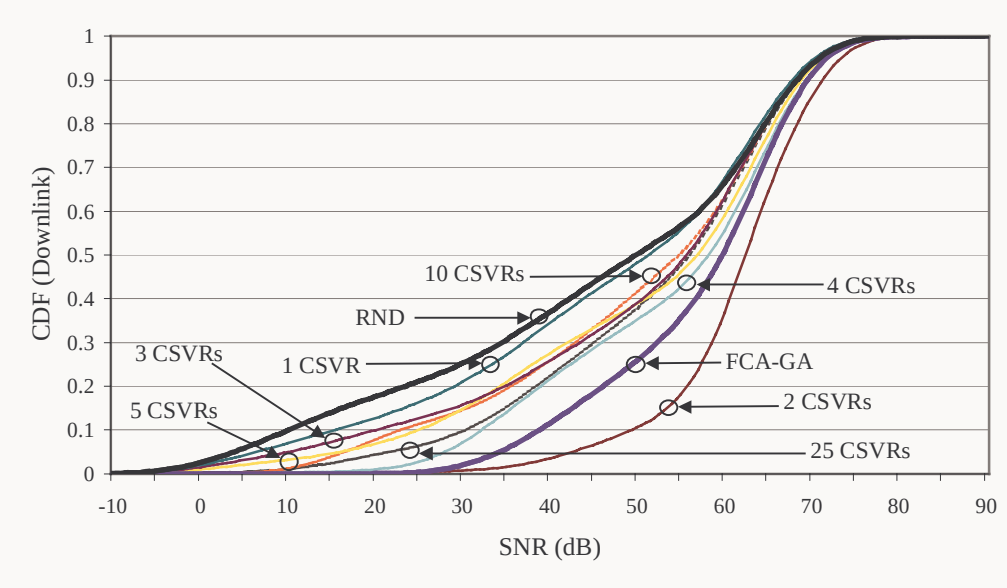

Figure 6.13: Downlink SNR (dB) performance for DCA-GA with various numbers of CSVRs.

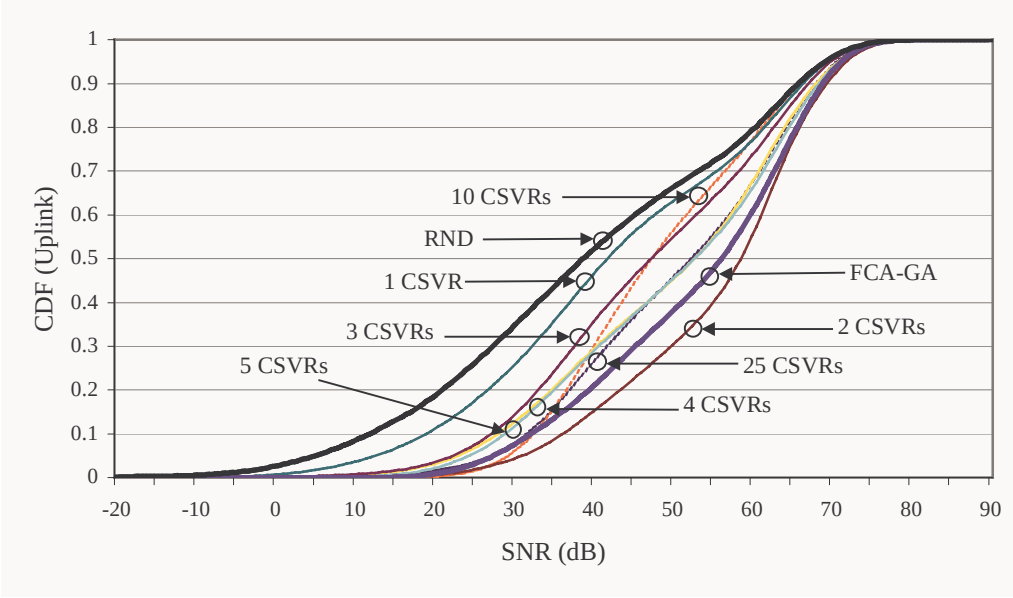

Figure 6.14: Uplink SNR (dB) performance for DCA-GA with various numbers of CSVRs.

The downlink and uplink SNR performances for DCA-GA are shown in Figure 6.13 and Figure 6.14 respectively. These results are taken from the final two normalised hours (7200 nmsec) of the simulation to avoid the transient state at the start up of the simulation. RND and FCA-GA SNR performances are also included as benchmarks in these figures. In both uplink and downlink, the DCA-GA with 2 CSVRs ($\tilde{N}_c$ = 25) has the best SNR performance while DCA-GA with 1 CSVR ($\tilde{N}_c$ = 50) has the worst SNR performance. The performance of DCA-GA with 2 CSVRs ($\tilde{N}_c$ = 25) is better than that of FCA-GA (benchmark). The performances of DCA-GA where the number of CSVRs is greater than 2 lies between the performance of FCA-GA and the performance of DCA-GA with 1 CSVR.

In FCA-GA an individual string corresponds to a specific channel assignment of the entire network and each individual is evaluated using the fitness function $\boldsymbol{F}_{FCA}(t)$ (or cost) using (4.12), which is also the summation of all interference power in the entire network. For the network using DCA-GA, whenever an AP uses a different channel (from the previous channel usage), the channel assignment for the entire network changes. At this point it is possible to use $\boldsymbol{F}_{FCA}(t)$ to analyse the fitness of the resultant channel assignment. The value of $\boldsymbol{F}_{FCA}(t)$ for the simulated network using DCA-GA is recorded for each scenario. Since the channel assignment changes (since it is dynamic) in DCA-GA, the average $\boldsymbol{F}_{FCA}(t)$ value (or cost) for the last 2 normalised hours of the simulation is calculated and used for evaluation of the fitness

of the network. To avoid confusion, the fitness function $\boldsymbol{F}_{FCA}(t)$ used for FCA-GA will be referred to as cost and denoted as C_{DCA}. The average value of C_{DCA} and its standard deviation (evaluated for the final 2 normalised hours of the simulation) against average cluster size $\tilde{N}_c$ is plotted in Figure 6.15. The corresponding cost of the channel assignment used in FCA-GA (presented in Chapter 4) is also plotted in this figure. It can be seen in Figure 6.15 that the channel assignment from DCA-GA with $\tilde{N}_c$ = 25 (i.e. 2 CSVRs) has the lowest average cost followed by $\tilde{N}_c$ = 12.5 (i.e. 4 CSVRs) while $\tilde{N}_c$ = 50 (1 CSVR) has the worst (highest average cost) assignment. The standard deviation of C_{DCA} measures the variability in the value of C_{DCA}. Since C_{DCA} measures the cost of the channel assignment in the network, a change in channel at an AP (or that of a number of APs) may cause a change in C_{DCA} (unless the changes do not affect the interference environment). Hence, a small standard deviation indicates fewer channel changes and that the network is closer to converging to a stable channel assignment. In Figure 6.15, there is a clear correlation between deviation and average cost. The lower the cost, the smaller the deviation and vice versa when the cost is high. Since the simulation is run for a limited time duration, some of the scenarios (e.g. $\tilde{N}_c$ is 10, 16.67 and 50) have apparently not converged and have a high average cost, which will lead to a poorer SNR performance. Although the FCA-GA cost (1.9×10^{-5}) is three times lower than the average cost of DCA-GA ($C_{DCA} = 5.9 \times 10^{-5}$) with $\tilde{N}_c$ = 25, the SNR performance of DCA-GA is better than that of FCA-GA as shown in Figure 6.13 and Figure 6.14. This is because the fitness plotted for FCA-GA in Figure 6.15 is calculated based on the interference created by one SU in the middle of the sector for each AP. In this simulation, the SUs are placed randomly within the sector of the AP and this gives a different interference environment for FCA-GA. Furthermore, the SNR performances shown Figure 6.13 and Figure 6.14 are obtained from the three shaded cells (shown in Figure 5.12), whereas the fitness for FCA-GA is obtained from the entire network. However, in general FCA-GA still manages to outperform all the other DCA methods since it has a priori knowledge of the entire network.

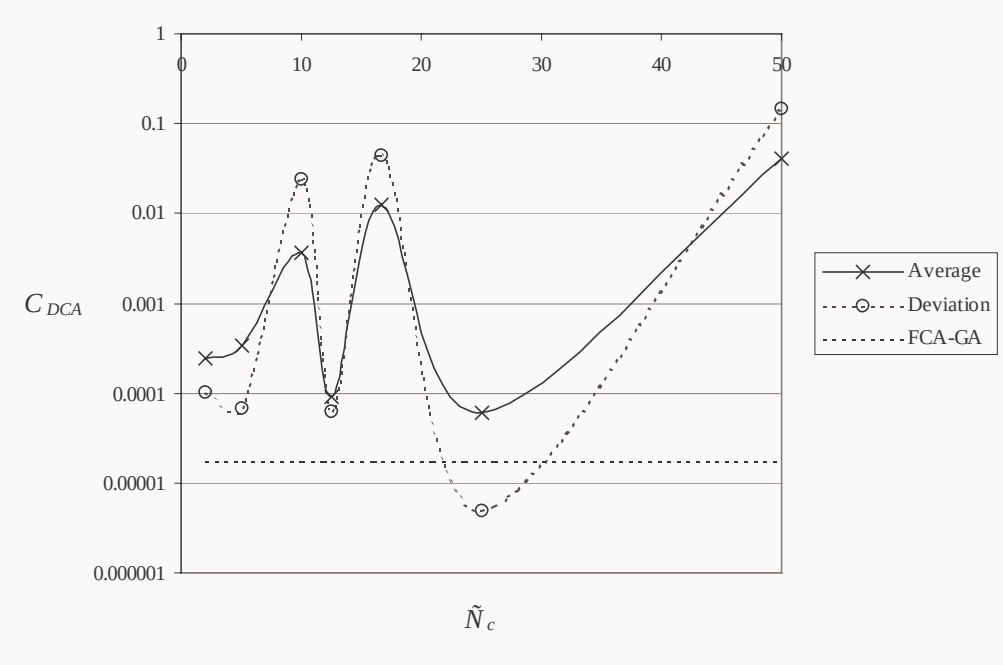

Figure 6.15: Average and standard deviation of C_{DCA} against $\tilde{N}_c$.

The average effective data throughput π_{AVG} for SNR_{RX} > 21 dB (i.e. a packet is received with a SNR of at least 21 dB at an AP and/or SU) is plotted in Figure 6.16. The transmit portion T_{TX} is the same for all scenarios therefore π_{AVG} is proportional to the SNR performance. DCA-GA with $\tilde{N}_c$ = 2 has the highest π_{AVG} while DCA-GA with $\tilde{N}_c$ = 50 has the lowest π_{AVG}. Consider the scenarios where the network has converged into a stable channel assignment, specifically the scenario where $\tilde{N}_c$ = 25, 10, 12.5 and 25 (i.e. a standard deviation of C_{DCA} of about 0.0001 or less). In Figure 6.16, a line is fitted through these points (shown circled in Figure 6.16) and its trend shows that π_{AVG} (and hence SNR performance) increases as $\tilde{N}_c$ increases. As the cluster size increases, the amount of centralisation increases, which leads to a higher degree of centralised information being available thereby giving rise to a better SNR performance. Figure 6.15 and Figure 6.16 show a close correlation between the average cost C_{DCA} and the average effective data throughput π_{AVG} of the network.

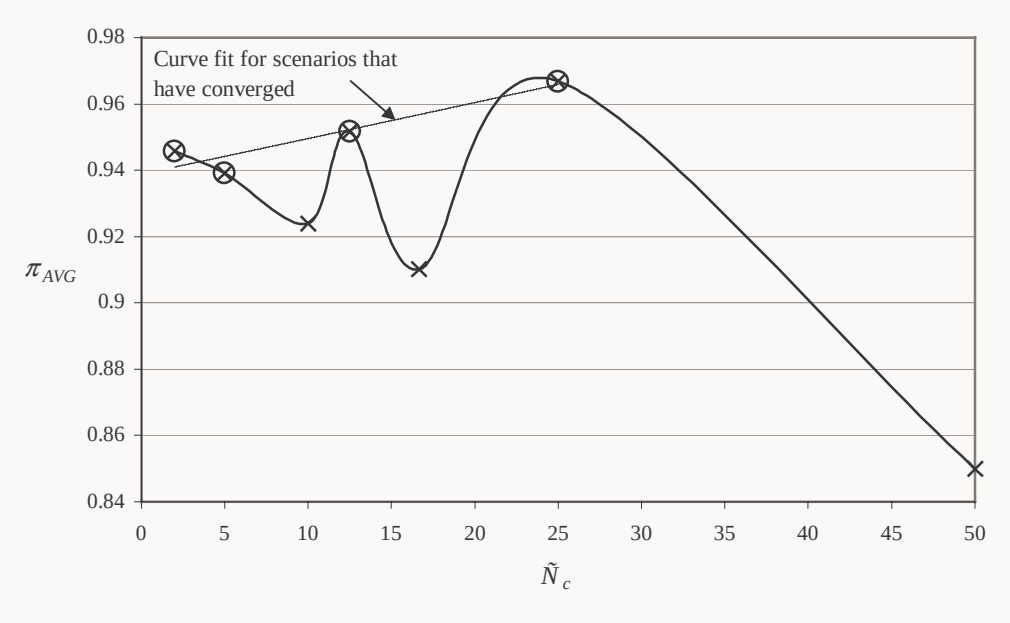

Figure 6.16: Average effective data throughput π_{AVG} as a function of average cluster size $\tilde{N}_c$.

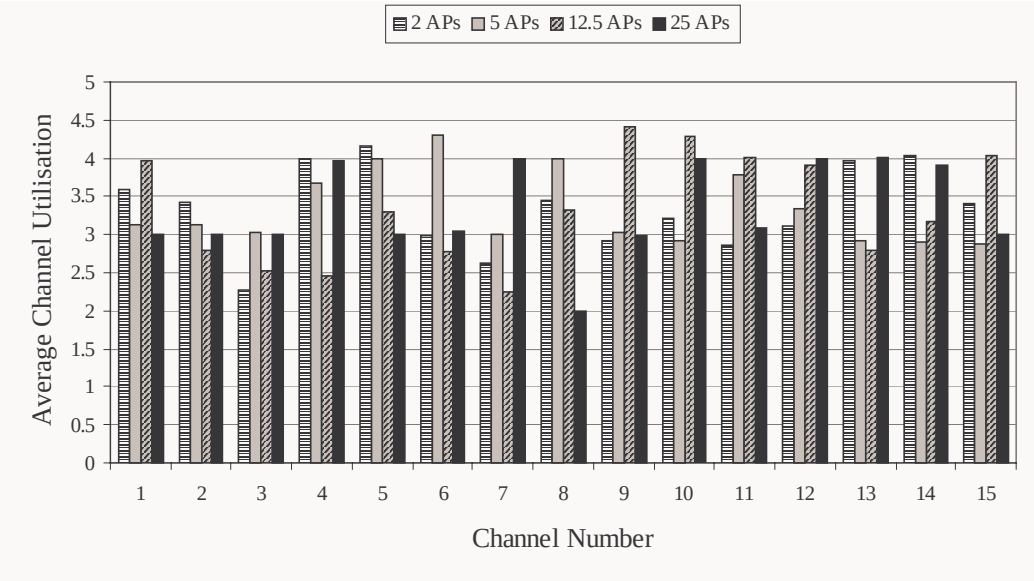

Figure 6.17: Average channel utilisation for scenarios that have converged ($\tilde{N}_c$ = 2, 5, 12.5 and 25).

The average channel utilisation for the scenarios that have converged are plotted in Figure 6.17. These values are taken during the last 2 normalised hours of the simulations. For this scenario (7 cells, 50 APs and 277 SUs), the ideal channel utilisation is 3.333 (i.e. 50 APs divided by 15 channels) for each channel. However, the number of APs per channel should be an integer for a network that is completely converged. Since the ideal channel utilisation of 3.333 is not an integer number, to achieve a close to uniform channel utilisation, there should be 10 channels with a

utilisation of 3 APs and 5 channels with a utilisation of 4 APs (as in the case of FCA-GA shown in Figure 4.38). For the converged scenarios, the channel utilisation shown in Figure 6.17 is close to that of FCA-GA. The channel utilisations are more uniform in these DCA-GA scenarios than that for CS–20dB shown previously in Figure 6.7.

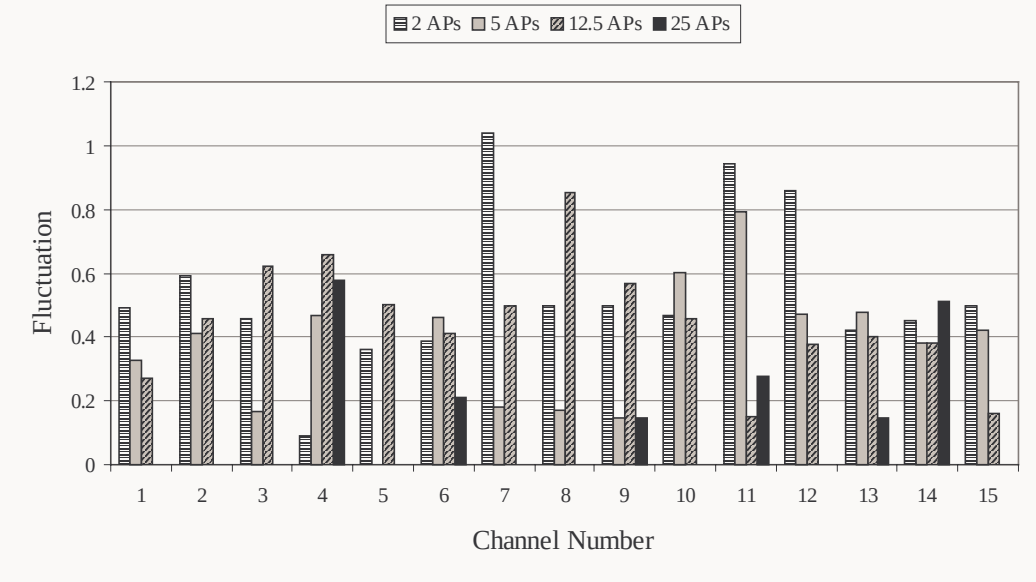

Figure 6.18: Channel fluctuation for scenarios that have converged ($\tilde{N}_c$ = 2, 5, 12.5 and 25).

The channel fluctuations for the scenarios that have converged are plotted in Figure 6.18. The channel fluctuations are low since most APs have settled upon a channel, particularly so for the scenario with $\tilde{N}_c$ = 25 where 9 channels show no fluctuation at all. For this scenario, most of the channels have 3 or 4 APs with the exception of channel number 8, which has only 2 APs.

The channel utilisations for the scenarios that have not converged are plotted in Figure 6.19. The channel utilisation is less uniform particularly for the scenario when $\tilde{N}_c$ = 10 APs. The corresponding channel fluctuations for these scenarios are plotted in Figure 6.20, which shows a higher degree of channel fluctuation compared with those presented in Figure 6.18 for the converged scenarios.

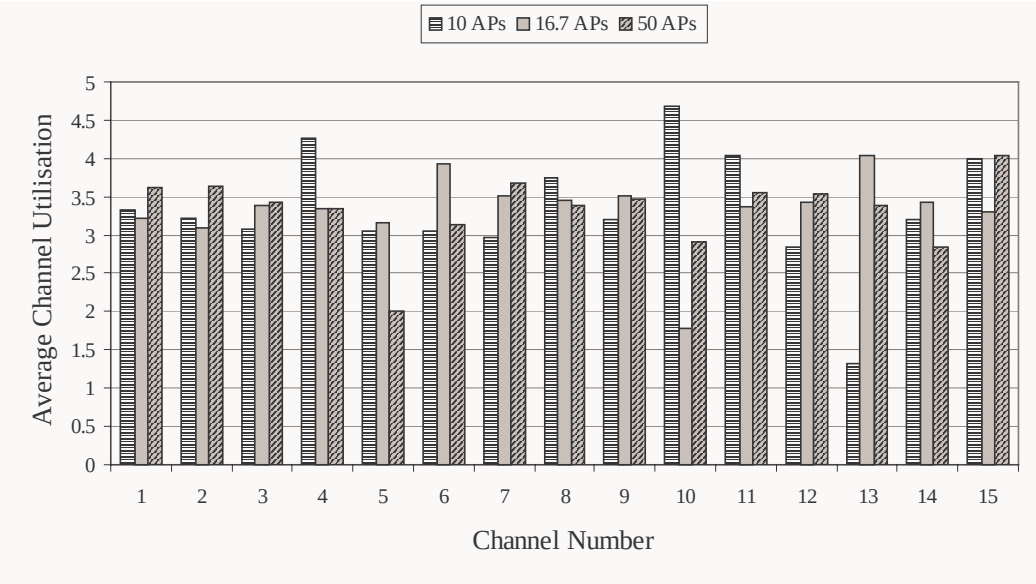

Figure 6.19: Average channel utilisation for scenarios that have not converged ($\tilde{N}_c$ = 10, 16.7 and 50).

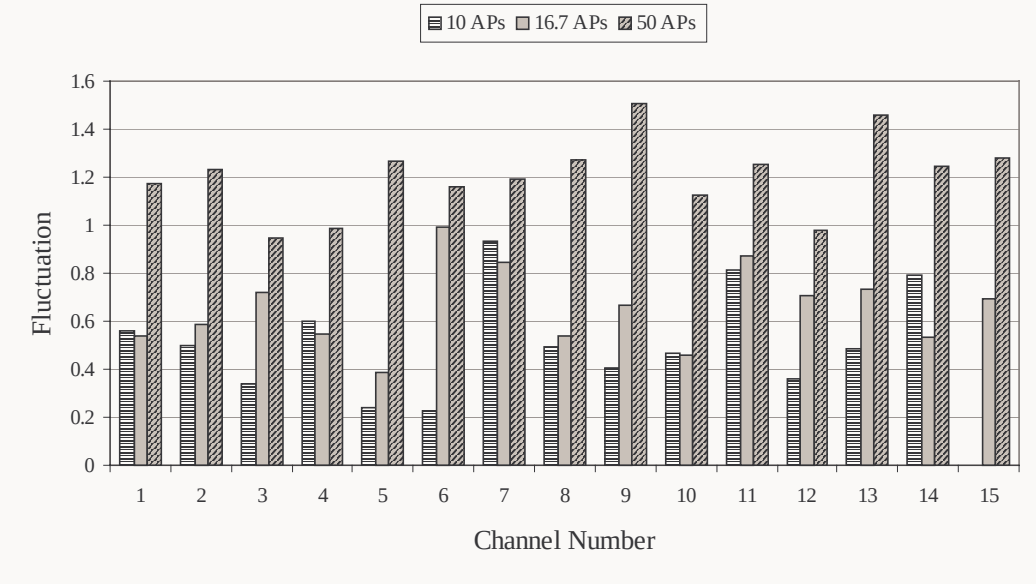

Figure 6.20: Channel fluctuation for scenarios that have not converged ($\tilde{N}_c$ = 10, 16.7 and 50).

The standard deviation of the cost C_{DCA} from Figure 6.15 and the average channel fluctuation against $\tilde{N}_c$ are plotted together in Figure 6.21. It may be observed that there is a close correlation between these two standard deviations, since both correspond to the degree of convergence in the channel allocation of the entire network.

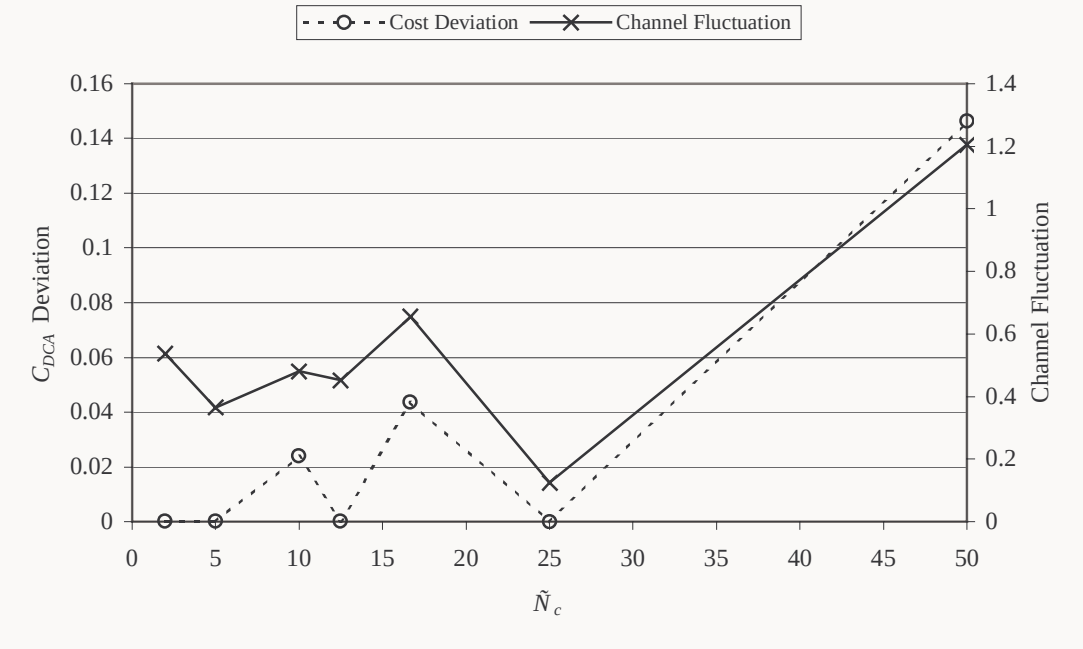

Figure 6.21: Cost C_{DCA} deviation and channel fluctuation.

Plots of C_{DCA} for the scenarios that have converged (i.e. $\tilde{N}_c$ = 2, 5, 12.5 and 25) against time are plotted in Figure 6.22. The C_{DCA} for FCA-GA is also plotted here. It is observed that most of the scenarios stabilise at about 15000 nmsec. For a system with a data rate of 25 Mbps, one nmsec is equivalent to 18.24 μs and hence 15000 nmsec is equivalent to about 0.28 seconds. This corresponds to about 78 generations. In contrast to FCA-GA, which requires 400 to 800 generations to converge (see Figure 4.29), DCA-GA requires fewer generations to reach a stable state. This is because each cluster has a smaller number of APs than that in FCA-GA, which gives rise to a shorter string thereby reducing the size of the search space. Hence, each cluster converges faster and as a result, the entire network converges when most of the clusters have converged. The Genetic Algorithm in FCA-GA takes about half a day (for the 50 AP scenario) to more than a day (for the 136 AP scenario) to complete 10000 generations running on a dual 700 MHz Pentium 3 processors PC. In DCA-GA, the complexity of the algorithm is limited to the size of the cluster whereas in FCA-GA, the complexity of the algorithm increases exponentially with the total number of APs.

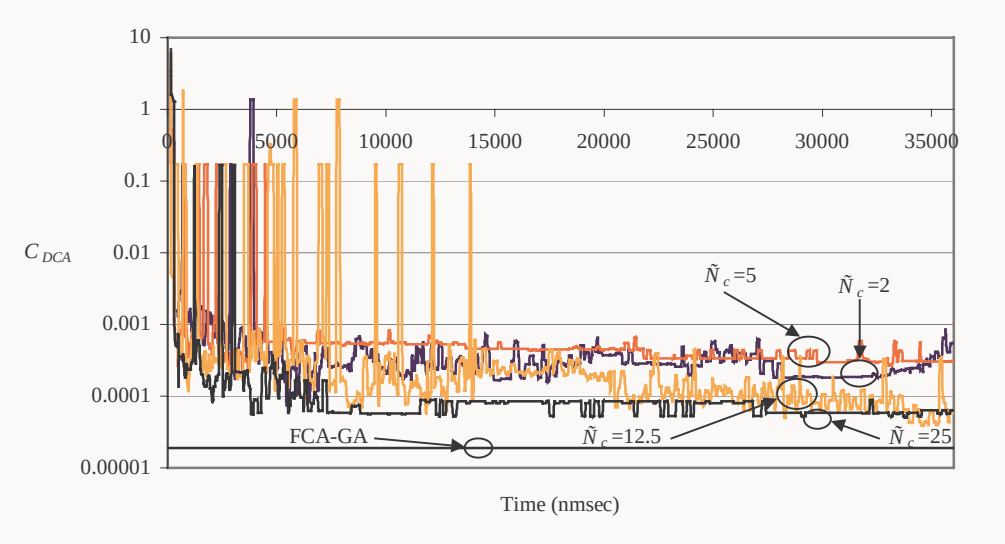

Figure 6.22: C_{DCA} as a function of time for scenarios that have converged ($\tilde{N}_c$ = 2, 5, 12.5 and 25).

6.2.4 Interference Avoidance

The response of the network using DCA-GA to external disturbances (after it has converged) is investigated in [89]. This is done by place two continuous wave (CW) interferers that use an omni directional antenna into the centre cell of the 7 cell network used in the previous section for the scenario where $\tilde{N}_c$ = 25. These interferers begin transmitting at 14400 nmsec at a power of 30 dBm using channels that will interfere with two of the APs in the cell. These interferers continue to transmit using the same channels until the end of the simulation. The temporal SNR responses for the downlink and uplink are shown in Figure 6.23. Initially the SNR stabilises after 7200 nmsec (2 normalised hours), and it is observed that a balanced downlink and uplink SNR performance is achieved. The interferers are activated at 14400 nmsec (4 normalised hours) causing the SNR performance to deteriorate. Initially the APs in the affected cell select different channels that may interfere with other cells (especially neighbouring cells). These other cells in turn select other channels and this causes a chain reaction, introducing further interference into the affected cell. Consequently, a poor SNR performance is evident for a short bust, which is seen to have a duration of less than 10800 nmsec (3 normalised hours). The APs using DCA-GA are eventually able to find a new channel assignment and re-converged to a new stable state.

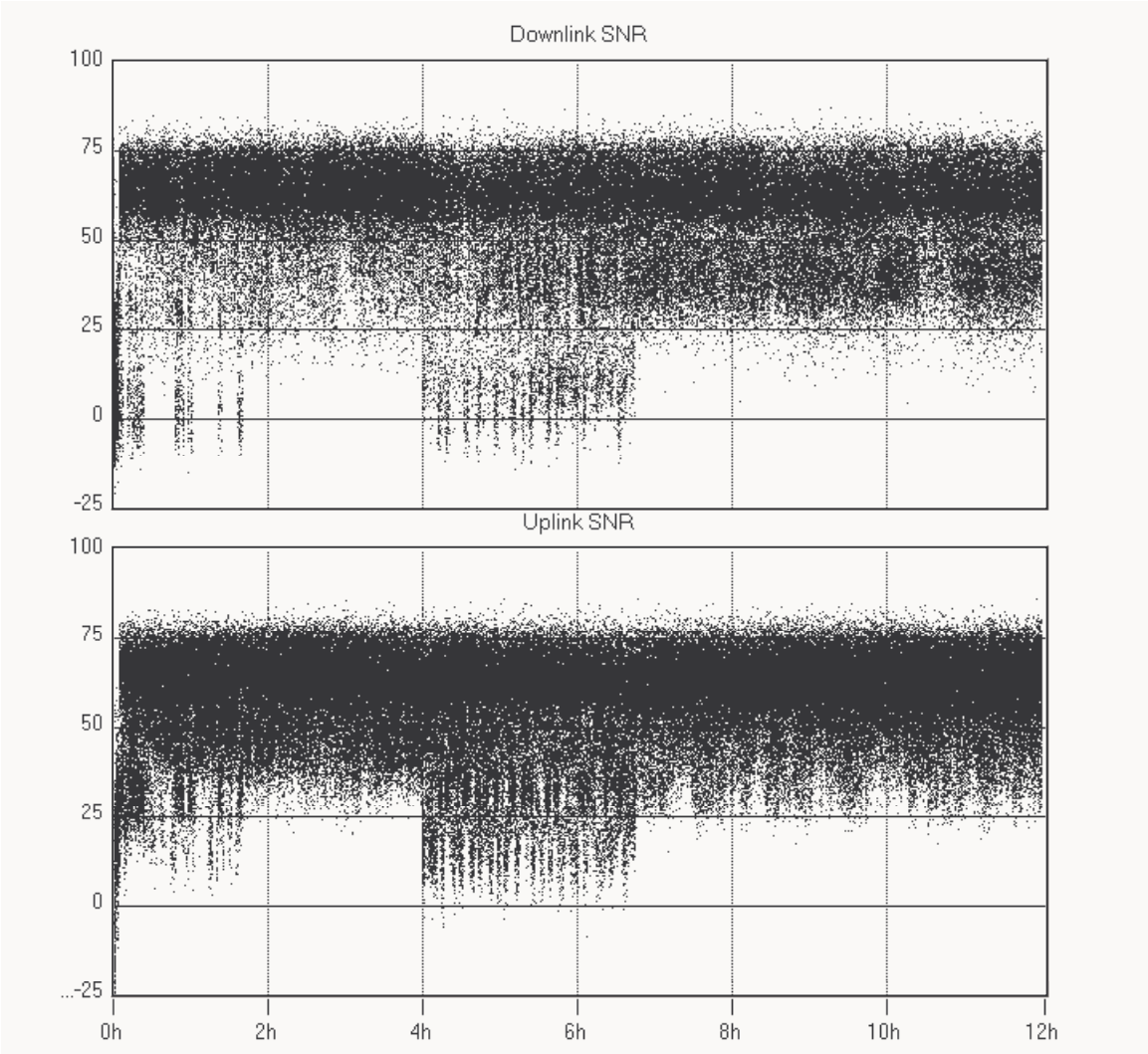

Figure 6.23: Temporal SNR (dB) response for cell with two CW interferers.

The corresponding C_{DCA} as a function of time is plotted in Figure 6.24. The external disturbance caused by the two CW interferers at 14400 nmsec affects the C_{DCA} of the entire network for about 10000 nmsec before the network finds a new stable state and hence recovers from the disturbance. For a system running at 25 Mbps, 10000 nmsec corresponds to 0.18 seconds.

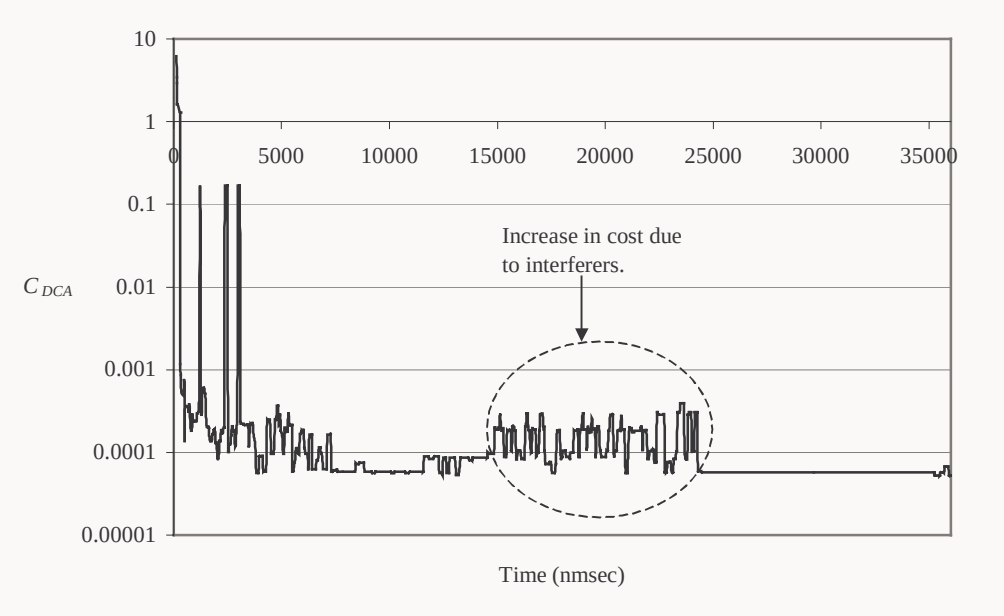

Figure 6.24: C_{DCA} as a function of time with the two CW interferers.

6.3 Simulations and Results

The SNR performance of the proposed DCA-GA is investigated in more detail using simulations modelled in OPNET. The simulation consists of 136 APs and 669 SUs randomly distributed in 37 cells, where the cell arrangement is shown in Figure 4.34. This is the same scenario used in Section 4.4 and similarly the results are taken from the three shaded cells. The traffic and propagation models used are the same as those described in Chapter 3. All the APs use the same channel (Channel 1) at the start of the simulation and to avoid the poor performance during the start up transient, the results are taken from the final 2 normalised hours (7200 nmsec) of the simulation.

Using this BFWA scenario, LI, DCA-GT, CS and DCA-GA are simulated. In LI the transmit portion T_{TX} (T_{TX}/M_{MAC} = 30) is increased so that its data throughput is close to that achieved in CS (where on average $C_{SCAN} \approx 1$ channel). The DCA-GT parameters are T_L/M_{MAC} = 3.44, T_T/M_{MAC} = 56 and p^*_{MIX} = 0.488 giving a $T_{EFF} \approx 30$. Similarly in DCA-GT, the set (T_L, T_T) is selected such that its effective transmit portion T_{EFF} gives a data throughput that is close to that used in CS. In CS, the required interference threshold T_{INT} is chosen such that SNR^* = 21 dB. Finally, for DCA-GA, the same set of parameters used in the previous section is employed. The total number of CSVR is set to 5, which gives an average cluster size $\tilde{N}_c$ = 27.2. The

T_{TX} of DCA-GA is set such that its data throughput is similar to those in the LI, DCA-GT and CS simulations.

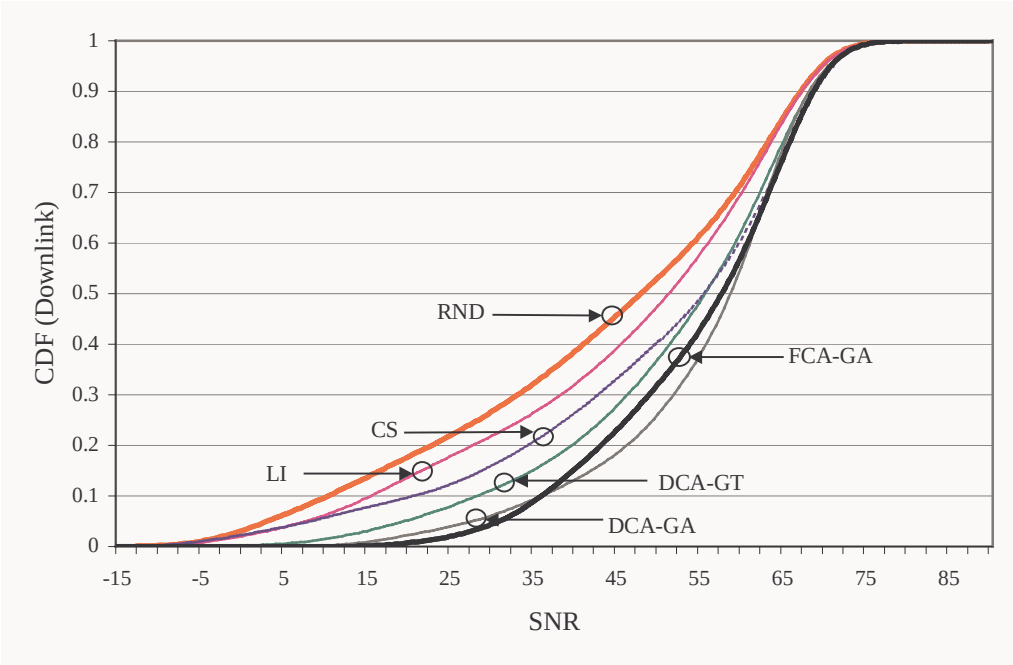

Figure 6.25: Downlink SNR performance.

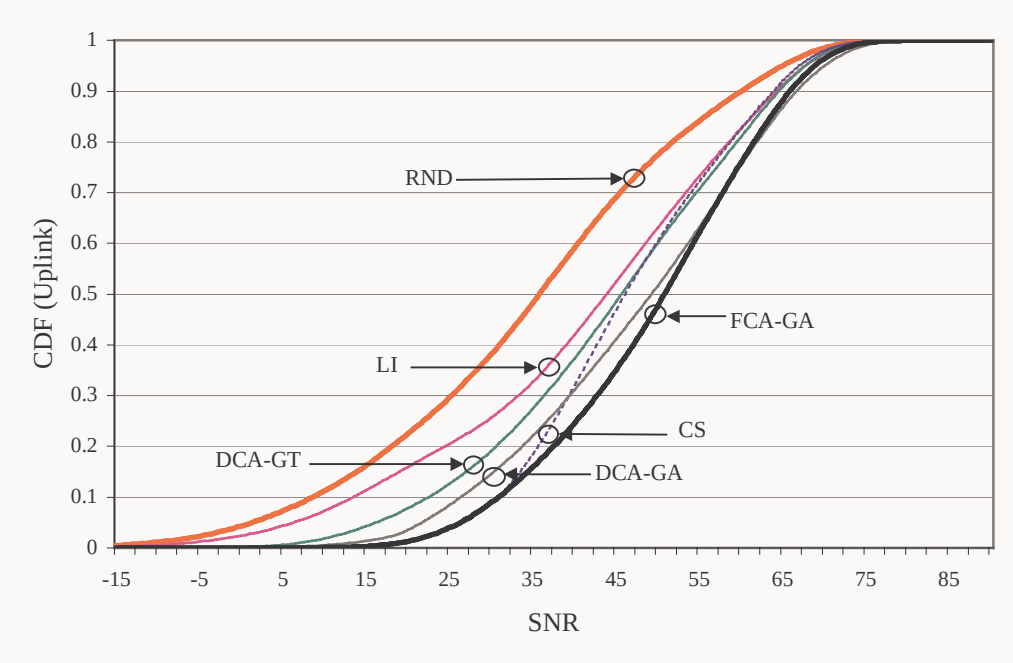

Figure 6.26: Uplink SNR performance.

Figure 6.25 and Figure 6.26 show the SNR performances for the downlink and uplink directions respectively. Consistent with the previous results (in Chapter 4 and 5),

RND, which has the least network information, has the worst SNR performance while FCA-GA with almost complete knowledge of the entire network has the best SNR performance. In the downlink direction, the best SNR performance after FCA-GA is DCA-GA followed by DCA-GT, CS and LI. However, between a SNR of 35 dB to 60 dB, DCA-GA outperformed FCA-GA in the downlink direction. In the uplink direction, the best SNR performance after FCA-GA is CS followed by DCA-GA, DCA-GT and LI. With the exception of CS, the SNR performance rankings are consistent in both directions. As shown previously, CS has an unbalanced downlink and uplink SNR performance where the downlink has a much worse performance than the uplink. This is due to the limited amount of measured information available at the AP. In CS C_{SCAN} is 1.08 channels per scan with a standard deviation of 0.37. Hence, CS is able to find a channel with only one channel measurement. The 1% SNR, $P_{\geq 21}$ and average effective data throughput π_{AVG} for the simulated schemes are summarised in Table 6.1.

	RND	LI	CS	DCA-GT	DCA-GA	FCA-GA
1% SNR (down)	-5.0 dB	-4.0 dB	-5.0 dB	8.0 dB	15.5 dB	21.5 dB
1% SNR (up)	-10.5 dB	-6.0 dB	18.5 dB	7.5 dB	13.5 dB	19.5 dB
1% SNR (overall)	-8.0 dB	-5.0 dB	1.5 dB	7.5 dB	14.5 dB	20.0 dB
$P_{\geq 21}$ (down)	0.814	0.854	0.897	0.943	0.974	0.991
$P_{\geq 21}$ (up)	0.761	0.832	0.983	0.914	0.957	0.984
$P_{\geq 21}$ (overall)	0.784	0.841	0.952	0.925	0.965	0.987
π_{AVG}	0.784	0.835	0.945	0.918	0.958	0.987

Table 6.1: 1% SNR, $P_{\geq 21}$ and π_{AVG}.

DCA-GA using partially centralised information has an overall 1% SNR gain of 13 dB over CS, 7 dB gain over DCA-GT and 19.5 dB gain over LI. The 1% SNR and $P_{\geq 21}$ for CS are higher in the uplink than they are in the downlink whereas in the other methods, the 1% SNR and $P_{\geq 21}$ are higher in the downlink than they are in the uplink (consistent with previous observations in Chapter 4 and 5). The disparity in 1% SNR performance between the downlink and uplink in CS amounts to some 23.5 dB and is significantly greater than the average of 2.4 dB for the other methods. Once again, this unbalanced downlink and uplink performance in CS is due to the lack of measured information at the AP. FCA-GA has the highest π_{AVG} followed in descending order by DCA-GA, CS, DCA-GT, LI and finally RND. LI using a shorter transmit portion (T_{TX}/M_{MAC} = 11.44) has a higher π_{AVG}, specifically 0.854 (from Table 4.1) versus 0.835 than that achieved with a longer transmit portion (T_{TX}/M_{MAC} = 30).

The overall 1% SNR is also lower for LI with the longer transmit portion compared to the one with the shorter transmit portion. The increase in the length of the transmit portion in LI causes the network to react slowly to interference changes thereby giving a slight reduction in performance. However, in DCA-GT, the increase in T_{EFF} improves the overall performance with a π_{AVG} = 0.918 when T_{EFF} = 30 compared to π_{AVG} = 0.891 when T_{EFF} = 11.44 (as in Table 5.1). The higher T_{EFF} gives an overall 1% SNR gain of 3.5 dB over that for the lower T_{EFF}. The increase in T_{EFF} in DCA-GT increases the data throughput without sacrificing the reaction time to interference changes.

Figure 6.27 presents the average channel utilisation (taken from the last 7200 nmsec of the simulations) for LI, DCA-GT, CS and DCA-GA. Uniform channel utilisation is achieved in LI and DCA-GT. DCA-GA has slight deviation from uniform utilisation mainly because it settles upon a local optimum solution. In CS, owing to the lack of measured information, it settles upon a local optimum solution that yields a non-uniform channel utilisation. The channel utilisation for RND and FCA-GA is not shown here since they have been given previously in Figure 4.38.

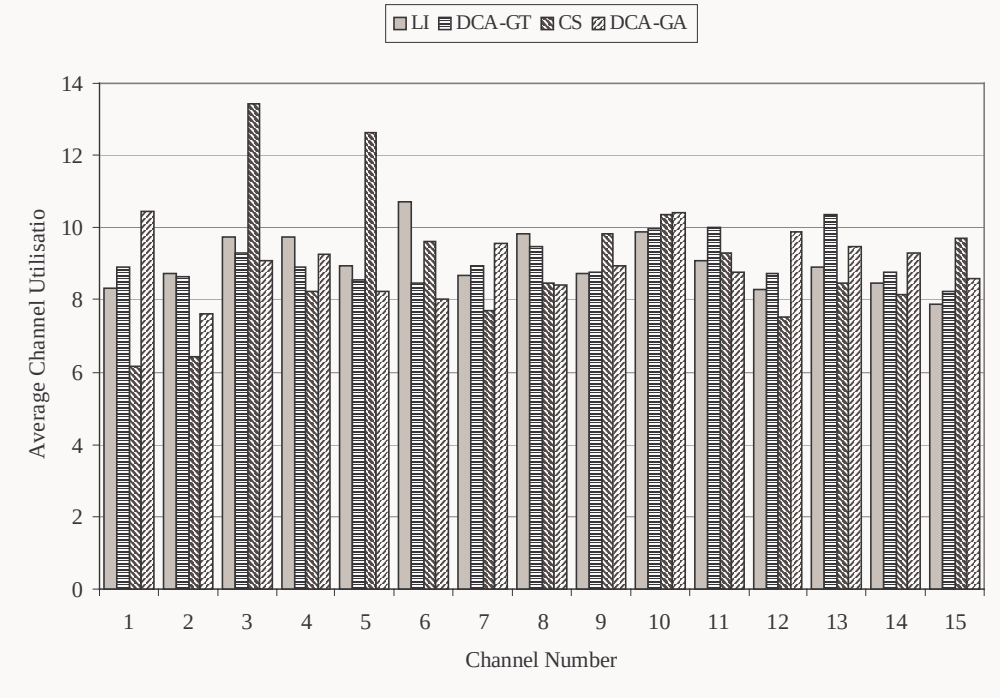

Figure 6.27: Average channel utilisation.

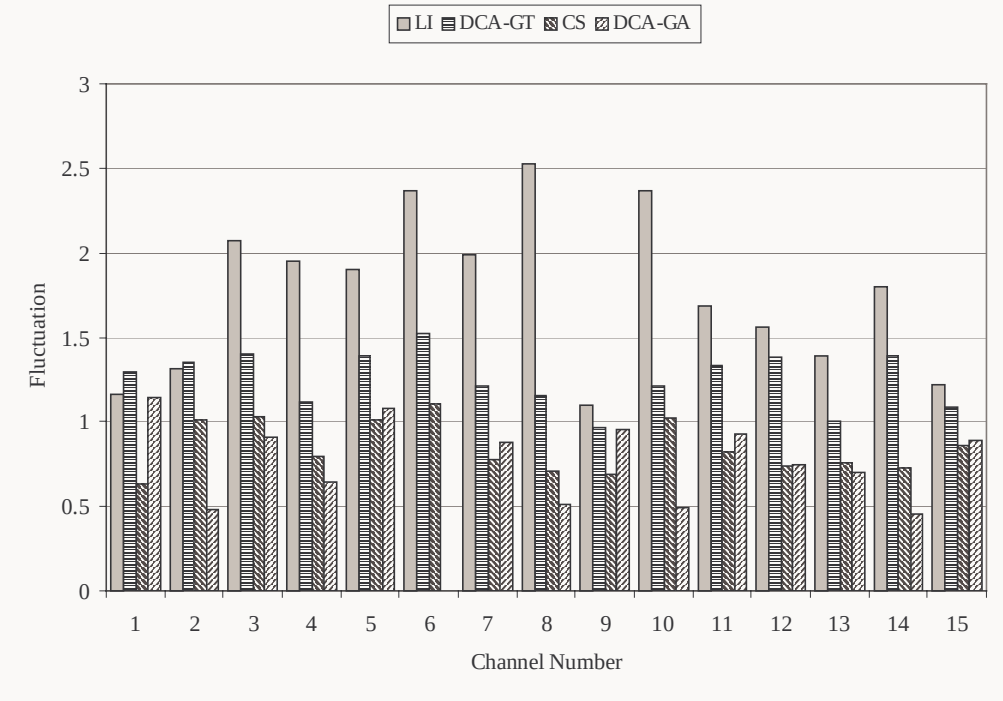

Figure 6.28: Channel fluctuation.

Figure 6.28 shows the channel fluctuation calculated as the standard deviation of the channel utilisation (taken during the final 2 normalised hours – 7200 nmsec of the simulations). LI has the highest channel fluctuation making its channel usage less predictable by other APs giving it a mediocre performance. This is followed by DCA-GT and CS. DCA-GA has the lowest channel fluctuation compared to the other methods since it has converged to a local optimum solution (notably Channel 6 shows no fluctuations). The average channel fluctuation for all the methods are shown in Figure 6.29. As expected, RND changes channel very often and therefore it has the highest channel fluctuation. The average channel fluctuation across all channels for LI is 1.76 in this scenario. This figure is similar to that found in LI with a lower value of T_{TX} (T_{TX}/M_{MAC} = 11.44 as used in Chapter 4 and 5), which has an average channel fluctuation of 1.78. Since both LI methods have the same level of fluctuation, the one with the higher value of T_{TX} settles upon a poorer sub-optimal solution compared to the one with a lower T_{TX}. This is because, the LI method with the higher value of T_{TX} exploits a channel more than it explores different channels. In contrast, the average channel fluctuation for DCA-GT in Chapter 5 is 1.55 compared to the current scenario, which has a value of 1.26. This is because the current scenario uses a larger value of T_{EFF} and so is able to settle upon a channel for a longer period thereby making it more predictable and using a balanced amount of channel explorations and

exploitations, give rise to a higher performance. CS with a minimum number of channel measurements settles upon a channel for a longer period than DCA-GT and LI. DCA-GA with partially centralised information converges to a stable channel allocation and hence has the smallest channel fluctuation. Finally, FCA-GA has a fixed channel assignment and therefore has no channel fluctuation.

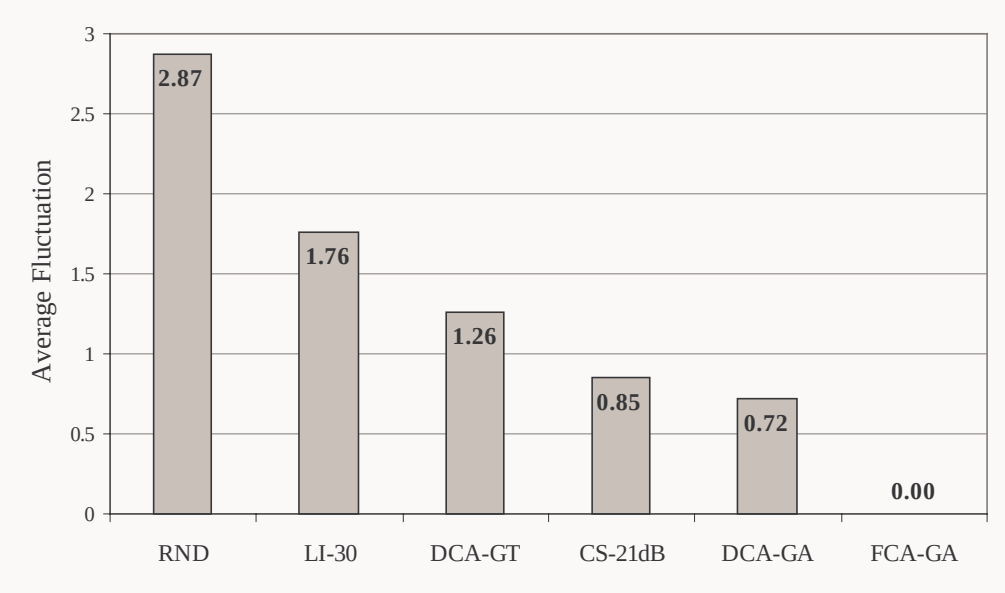

Figure 6.29: Average channel fluctuation.

6.4 Conclusion

LI and DCA-GT have a low data throughput owing to the large amount of channel measurement performed during the SCAN periods. Threshold and priority based DCA uses less channel measurements and is able to increase the data throughput. However, when simulating the popular CS method it has been found that a threshold and priority based DCA can give an unbalanced (downlink and uplink) performance and non-uniform channel utilisation.

A novel DCA (DCA-GA) operating in a partially centralised system is proposed that utilises a genetic algorithm during system operation to find an optimum channel list. The APs and CSVR in DCA-GA operates asynchronously thereby avoiding the delays that are inherent in a conventional centralised system. With partially centralised information, DCA-GA requires less channel measurement and hence offers an increase in the data throughput. DCA-GA has more network information available to it than the other DCA methods and hence it has the best performance with an overall

1% SNR gain of 13 dB over CS and 7 dB over DCA-GT and an average effective data throughput advantage of 325 kbps over CS and 1 Mbps over DCA-GT in a nominal 25 Mbps link. DCA-GA is also able to achieve near uniform channel utilisation. It is also shown that DCA-GA is able to avoid external interference and is able to re-converge to a new stable state.

Increasing the transmit portion T_{TX} in LI or T_{EFF} in DCA-GT effectively increases the data throughput G_{MAC}. However, in LI the increased length of T_{TX} does not lead to an increase in the SNR performance or to a rise in the average effective data throughput π_{AVG}. This is because the increase in T_{TX} increases the AP's tendency to exploit a channel more than to explore other channel usage, and this causes network to react slowly to interference changes. In contrast, the increase in T_{EFF} in DCA-GT improves the overall performance. Using a lower value of T_{EFF} (T_{EFF}/M_{MAC} = 11.44) as employed in Chapter 5, DCA-GT has an overall 1% SNR gain of 8.5 dB and an average effective data throughput gain of 925 kbps (in a nominal 25 Mbps link) over LI (T_{TX}/M_{MAC} = 11.44). DCA-GT using a higher value of T_{EFF} (T_{EFF}/M_{MAC} = 30) has an overall 1% SNR gain of 12.5 dB and an average effective throughput gain of 2.075 Mbps (in a nominal 25 Mbps link) over LI (T_{TX}/M_{MAC} = 30). This improvement in performance for DCA-GT with a larger value of T_{EFF} is because DCA-GT is able to maintain a balance between exploring different channel usage and exploiting a channel.

7 CONCLUSION

This dissertation has presented the motivation and the necessity for channel allocation in a BFWA network. A comprehensive survey of existing channel allocation methods has been provided and three novel channel allocation methods for application to BFWA networks have been proposed, namely, FCA-GA, DCA-GT and DCA-GA. A number of concluding remarks concerning the work presented in this dissertation will now be given and some possible areas for future work will be outlined.

7.1 Summary of the Research

The frequency spectrum that is available to a wireless system is usually limited and hence the channels within the spectrum need to be reused. Channel reuse causes interference within the network and interference degrades the SNR of received packets, which decreases the capacity of the system. Effective channel allocation reduces this interference and in doing so increases the capacity of a system.

The channel allocation problem is essentially to allocate the available channels to the base stations in the wireless network in such a way that ensures an acceptable SNR at all the base stations and subscriber units. The channel allocation problem is classified as a NP-complete problem. The large variety of channel allocation methods surveyed in Chapter 2 are classified using the proposed Channel Allocation Matrix, where the channel allocation methods are distinguished by the techniques they use to obtain network information.

Channel allocation has been studied extensively, particularly for circuit switched cellular networks. However, relatively few channel allocation methods have been proposed for packet based wireless networks. Chapter 3 describes a Broadband Fixed Wireless Access (BFWA) system that offers a wireless data service using a PRMA based MAC scheme. Data traffic has been shown to exhibit self-similar characteristics, which requires markedly different traffic models to those traditionally used for circuit switched voice traffic. The BFWA network, together with the data traffic sources and radio propagation models, are simulated using the OPNET network

simulation package in order to evaluate the performance of various existing and newly proposed channel allocation methods.

Currently fixed channel allocation is employed for BFWA networks. The use of DCA will considerably reduce planning and maintenance costs, however all current DCA schemes have only been applied to circuit switched voice networks. Their applicability to the BFWA scenario has been investigated and new DCA schemes have been proposed and evaluated. In Chapter 4, the popular Random Channel Allocation (RND) scheme is applied to BFWA and the simulation results are used as the worst case performance benchmark. The popular Least Interfered (LI) method is also implemented in the BFWA network in this chapter. A new Fixed Channel Allocation method (FCA-GA) using a genetic algorithm is also proposed and, in a quasi-static interference environment, it acts as the best case performance benchmark for the DCA methods under investigation.

Game theory is the study of interdependency between nodes in a system, and consequently it can be applied to a distributed DCA scheme which exhibits the required interdependence characteristic. In Chapter 5, a payoff function is proposed for the AP in the BFWA system and, using game theory, a new DCA method, namely DCA-GT is proposed. DCA-GT offers better performance than LI since it is able to use the measured information more effectively, yielding an overall 1% SNR gain of 8.5 dB compared to LI. It has been found that, for LI, an increase in the length of the transmit portion T_{TX} leads to a decrease in performance. In contrast, it has been found that, in DCA-GT, an increase in the effective transmit T_{EFF} portion improves the overall performance because DCA-GT is able to maintain a balance between exploring different channel usage and exploiting a channel.

Both LI and the proposed DCA-GT methods require extensive channel measurements to be performed and this lowers the data throughput of the system. Since the least interfered channel is always used, this method does not distinguish between good and bad channel in terms of their interference level. A threshold and priority based DCA is able to address the problems facing LI and DCA-GT. The most popular threshold and priority based DCA, namely Channel Segregation (CS), is implemented for a BFWA network and is investigated in Chapter 6. It is found that CS is able to increase the data throughput of the BFWA system but, owing to the decrease in

information concerning the interference environment, CS gives a non-uniform channel utilisation and an unbalanced uplink and downlink SNR performance. To address this problem, a novel DCA method, namely DCA-GA, is proposed in the context of a partially centralised system. DCA-GA has two subsystems where one is partially centralised whilst the other is fully distributed. Using partially centralised information, the AP requires less channel measurement, thereby improving the data throughput. The two subsystems operate asynchronously, so that the distributed subsystem operating at the AP inherits the fast response of a fully distributed DCA. DCA-GA is simulated using OPNET, and it is shown to outperform all the other DCA methods, having an overall 1% SNR advantage of 13 dB over CS and about 19 dB over LI. It is also shown that the DCA-GA is able to adapt to the introduction of external interference into the network in the form of two CW interferers.

7.2 Future Work

A possible way to improve the performance of DCA-GT would be to dynamically adapt its parameters (e.g. the transmit portions T_L and T_T) during network operation. For example, the APs can be made to detect the convergence of the network, and so increase the length of the transmit portion and change p_{mix} when the system nears convergence, and vice versa. Similarly for DCA-GA, the length of the transmit portions and the genetic parameters (e.g. population, mutation and elite fraction) can be set dynamically according to the conditions of the network. The CSVR can detect if the system is stabilising, and so instruct that the length of the transmit portion should be increased and the mutation rate decreased. Further research is required to find a function that can dynamically change these parameters, and a method to detect convergence without the need of a central controller.

The system can also obtain more interference information by requiring the SUs to measure the interference power of the channels in addition to the measurements performed at the APs. The APs can then use these measurements to hopefully improve its own allocation process. The improvement in SNR, using this method, must be weighted against the increased complexity and the additional signalling required by the system.

The proposed DCA methods are not limited for use in BFWA networks. The analysis is performed, based on the use of a SCAN and transmit portion in the MAC, and

hence it can be easily implemented in other systems, particularly those using unlicensed spectrum. The HIPERLAN 2 wireless LAN standard uses a similar PRMA MAC structure to that used in this BFWA application, and hence it will be relatively easy to implement the proposed DCAs into such a scheme.

The bottleneck encountered while conducting this research has been the slowness in executing the simulations, which to some extent is exacerbated by the OPNET tool. For example, the detailed radio pipeline stages used in OPNET, to calculate the interference power present in a packet, introduces a considerable overhead. A typical simulation for the 37 cell scenario described in Section 4.4 using a dual 700 MHz Pentium 3 PC takes up to two weeks for a 36000 nmsec simulation. Owing to the long simulation times required, a scenario with a smaller number of APs (e.g. the 7 cell scenario used in Section 5.4.2) had to be used to determine appropriate parameter settings for particular channel allocation methods. For example, the plots presented in Figure 5.13, where the effect of the mix probability p_{mix} in DCA-GT is explored, requires 7 such simulations. If a higher performance PC with several parallel processors had been available, the simulations performed using the 7 cell scenario could have used a greater number of APs and SUs (e.g. 1000 APs), thus increasing the statistical confidence in the results. Using a high performing PC, the simulations performed can have a longer duration (e.g. for 1000 normalised hours or 3.6 million nmsec) thereby achieving convergence, if it exists, for DCA methods such as LI.

Ideally, a real BFWA network could be rolled out consisting of 30 to 50 APs, allowing the proposed DCA methods to be implemented and tested in a real system. Analysis of these results could be used to validate the accuracy of the simulation model, and also help to improve the DCA methods.

APPENDIX A FUNCTION $\Gamma(T_T)$

The payoff function for the distributed DCA in Chapter 5 for AP j with respect to AP k is given in (5.3) and is rewritten here for convenience:

$$\boldsymbol{\pi}_{j,k}(t) = \boldsymbol{G}_j(t)\left((1-\boldsymbol{P}_C(t))\boldsymbol{O}_{j,k}(t) + \boldsymbol{S}_{j,k}(t)\right) \tag{A.1}$$

For every fixed transmit portion of AP k, $\boldsymbol{T}_k(t) = T_T$, there exists a transmit portion of AP j, $\boldsymbol{T}_j(t) = T_L$, such that AP j has the peak payoff. Let $\boldsymbol{\Gamma}(T_T)$ be a function that given a transmit portion T_T of an AP (e.g. AP k), it computes the value of T_L for the other AP (e.g. AP j) giving this AP a peak payoff (i.e., $\boldsymbol{\Gamma}(T_T) = T_L$).

Let the transmit portion of AP k be fixed that is $\boldsymbol{T}_k(t) = T_T$ for all t. Since the SCAN period, γ and number of channels, C are constants, the payoff function is hence dependent upon $\boldsymbol{T}_j(t)$. Let $\boldsymbol{T}_j(t) = T_j$ for all t, consequently the payoff function is now a function of T_j (i.e., $\boldsymbol{\pi}_{j,k}(t) = \boldsymbol{\pi}_{j,k}(T_j)$). The data throughput $\boldsymbol{G}_j(t)$, probability $\boldsymbol{P}_C(t)$, overlap functions $\boldsymbol{O}_{j,k}(t)$ and $\boldsymbol{S}_{j,k}(t)$ are therefore also a function of T_j. The differentiation of $\boldsymbol{\pi}_{j,k}(T_j)$ with respect to T_j is:

$$\begin{aligned}\boldsymbol{\pi}'_{j,k}(T_j) &= \boldsymbol{G}'_j(T_j)\left((1-\boldsymbol{P}_C(T_j))\boldsymbol{O}_{j,k}(T_j) + \boldsymbol{S}_{j,k}(T_j)\right) + \\ &\boldsymbol{G}_j(T_j)\left((1-\boldsymbol{P}_C(T_j))\boldsymbol{O}'_{j,k}(T_j) - \boldsymbol{P}'_j(T_j)\boldsymbol{O}_{j,k}(T_j) + \boldsymbol{S}'_{j,k}(T_j)\right)\end{aligned} \tag{A.2}$$

where, $\boldsymbol{G}'_j(T_j)$, $\boldsymbol{P}'_C(T_j)$, $\boldsymbol{O}'_{j,k}(T_j)$ and $\boldsymbol{S}'_{j,k}(T_j)$ are the differentiation of $\boldsymbol{G}_j(T_j)$, $\boldsymbol{P}_C(T_j)$, $\boldsymbol{O}_{j,k}(T_j)$ and $\boldsymbol{S}_{j,k}(T_j)$ respectively (with respect to T_j). Since the peak payoff for AP j occurs when $T_L < \gamma C + T_T$, the differentiation in (A.2) is valid for $T_j \le \gamma C + T_T$. Consequently overlap function $\boldsymbol{O}_{j,k}(T_j)$ is:

$$\boldsymbol{O}_{j,k}(T_j) = \frac{T_T}{\gamma C + T_T + T_j} \tag{A.3}$$

and the corresponding $\boldsymbol{S}_{j,k}(T_j)$ is:

$$\Sigma_{j,k}(T_j) = \frac{\gamma C}{\gamma C + T_T + T_j} \tag{A.4}$$

Equating (A.2) to zero will give the following cubic polynomial equation:

$$K_3 T_j^3 + K_2 T_j^2 + K_1 T_j + K_0 = 0 \tag{A.5}$$

where,

$$K_0 = \left(\gamma^3 C^3 - \gamma^3 C\right) T_T^3 + \left(2\gamma^3 C^4 + \gamma^2 C^4 - \gamma^2 C^3 - \gamma^3 C^2\right) T_T^2 + \left(2\gamma^4 C^5 + \gamma^3 C^5 - \gamma^3 C^4\right) T_T + \gamma^5 C^6 \tag{A.6}$$

$$K_1 = \left(\gamma C^2 + \gamma - 2\gamma C\right) T_T^3 + \left(3\gamma^2 C^3 - \gamma^2 C^2 - \gamma^2 C\right) T_T^2 + \left(2\gamma^3 C^3 + 2\gamma^3 C^4 - 2\gamma^3 C^2 - \gamma^2 C^3 + \gamma^2 C^4\right) T_T + \gamma^4 C^5 \tag{A.7}$$

$$K_2 = \left(2\gamma - \gamma C^2 - \gamma C\right) T_T^2 + \left(2\gamma^2 C^2 - 2\gamma^2 C^3 - \gamma^2 C\right) T_T - \gamma^3 C^4 \tag{A.8}$$

$$K_3 = (1 - C) T_T^2 + \left(\gamma - 2\gamma C^2\right) T_T - \gamma^2 C^3 \tag{A.9}$$

The function $\Pi(T_T)$ is found by solving (A.5) for T_j, where the positive root of (A.5) is T_L. The root of (A.5) is found using Newton-Rhaphson, which is an iterative method, where the $T_{L,i+1}$ is the $(i+1)^{th}$ iteration update with the rule as follows:

$$T_{L,i+1} = T_{L,i} + \Sigma_{N-R}(T_{L,i+1}) \tag{A.10}$$

and $\Sigma_{N\text{-}R}(T_{L,i+1})$ is the error function given as:

$$\Sigma_{N-R}(T_{L,i+1}) = -\frac{K_3 T_{L,i}^3 + K_2 T_{L,i}^2 + K_1 T_{L,i} + K_0}{3K_3 T_{L,i}^2 + 2K_2 T_{L,i} + K_1} \tag{A.11}$$

This method is iterated N times and for a large value of N, $\Pi(T_T) \approx T_L$.

The function $\Gamma(T_T)$ has a fixed point (T_F, T_F) such that $\Gamma(T_F) = T_F$. This fixed point corresponds to a peak payoff and thus can be found by equating $T_L = T_T = T_F$ in (A.5) and solving for T_F. Substituting $T_L = T_T = T_F$ into (A.5) gives the following fifth order polynomial equation:

$$H_5 T_F^5 + H_4 T_F^4 + H_3 T_F^3 + H_2 T_F^2 + H_1 T_F + H_0 = 0 \tag{A.12}$$

where,

$$H_0 = \gamma^5 C^6 \tag{A.13}$$

$$H_1 = 3\gamma^4 C^5 + \gamma^3 C^5 - \gamma^3 C^4 \tag{A.14}$$

$$H_2 = 3\gamma^3 C^4 + 2\gamma^3 C^3 - 3\gamma^3 C^2 - 2\gamma^2 C^3 + 2\gamma^2 C^4 \tag{A.15}$$

$$H_3 = \gamma^2 C^2 - 2\gamma^2 C + \gamma^3 C^3 - \gamma^3 C \tag{A.16}$$

$$H_4 = 4\gamma - 2\gamma C^2 - 3\gamma C \tag{A.17}$$

$$H_5 = (1 - C) \tag{A.18}$$

Once again, using Newton-Rhaphson with N iterations, the fixed point is found to be $T_F/M_{MAC} = 0.362$. At this point G_{MAC} (see Equation (4.2)) is 0.616 for $M_{MAC} = 66.61$, $\gamma = 1$ and $C = 15$. The simulated average effective data throughput π_{AVG} for RND in Chapter 4 is 0.784, which is higher than the G_{MAC} value of 0.616 at the fixed point (even neglecting the $P_{\geq 21}$ criterion). This is consistent with the description in Section 5.3 where the fixed point yields a lower payoff than the saturation payoff π_S (obtained when $T_L = T_T = \infty$). Hence, the fixed point is not a Nash Equilibrium.

REFERENCES

[1] William C. Y. Lee, *Mobile Cellular Telecommunications Systems*. New York: McGraw-Hill, 1989.

[2] V. H. Mac Donald, "The Cellular Concept," *The Bell System Technical Journal*, vol. 58, no. 1, pp. 15-41, January 1979.

[3] Vincentzio I. Roman, "Frequency Reuse and System Deployment in Local Multipoint Distribution Service," *IEEE Personal Communications*, pp. 20-27, December 1999.

[4] J. C.-I. Chuang, "Performance Limitations of TDD Wireless Personal Communications with Asynchronous Radio Ports," *Electronics Letters*, vol. 28, no. 6, pp. 532-534, 12 March 1992.

[5] William C. Y. Lee, "The Most Spectrum-Efficient Duplexing System: CDD," *IEEE Communications Magazine*, pp. 163-166, March 2002.

[6] Bertrand Benoit and Haig Simonian, "German 3G Auction Raises £30bn," *Financial Times*, pg. 1, 18 August 2000.

[7] Roger O. Crockett, Heather Green, Andy Reinhardt and Jay Greene, "All Net, All The Time," *BusinessWeek*, pp. 78-82, 29 April 2002.

[8] Michael R. Garey and David S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco: W. H. Freeman and Co., 1979.

[9] Young Chon Kim, Dong Eun Lee, Bong Ju Lee, Young Sun Kim and Biswanath Mukherjee, "Dynamic Channel Reservation Based on Mobility in Wireless ATM Networks," *IEEE Communications Magazine*, vol. 37, no. 11, pp. 47-51, November 1999.

[10] David A. Levine, Ian F. Akyildiz and Mahmoud Naghshineh, "A Resource Estimation and Call Admission Algorithm for Wireless Multimedia Networks Using the Shadow Cluster Concept," *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 1-12, February 1997.

[11] Jens Zander, "Radio Resource Management – An Overview," in *Proc. 46th IEEE Vehicular Technology Conference '96*, 28 April – 1 May, Atlanta, USA, vol. 1, pp. 16-20.

[12] Gregory J. Pottie, "System Design Choices in Personal Communications," *IEEE Personal Communications*, pp. 50-67, October 1995.

[13] D.T. Pham and D. Karaboga, *Intelligent Optimisation Techniques*. London: Springer, 2000.

[14] Richard M. Golden, *Mathematical Methods for Neural Network Analysis and Design*. Cambridge, Massachusetts: The MIT Press, 1996.

[15] J. J. Hopfield and D. W. Tank, "'Neural' Computation of Decisions in Optimization Problems," *Biological Cybernetics*, vol. 52, pp. 141-152, 1985.

[16] Melanie Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, Massachusetts: The MIT Press, 1999.

[17] Richard S. Sutton and Andrew G. Barto, *Reinforcement Learning*. Cambridge, Massachusetts: The MIT Press, 1999.

[18] I. Katzela and M. Naghshineh, "Channel Assignment Schemes for Cellular Mobile Telecommunication Systems: A Comprehensive Survey," *IEEE Personal Communications*, pp. 10-31, June 1996.

[19] J. Arthur Zoellner and C. Lyle Beall, "A Breakthrough in Spectrum Conserving Frequency Assignment Technology," *IEEE Transactions on Electromagnetic Compatibility*, vol. EMC-19, no. 3, pp. 313-319, August 1977.

[20] Frank Box, "A Heuristic Technique for Assigning Frequencies to Mobile Radio Nets," *IEEE Transactions on Vehicular Technology*, vol. VT-27, no. 2, pp. 57-64, May 1978.

[21] Goutam Chakraborty, "An Efficient Heuristic Algorithm for Channel Assignment Problem in Cellular Radio Networks," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 6, pp. 1528-1539, November 2001.

[22] Wei Wang and Craig K. Rushforth, "An Adaptive Local-Search Algorithm for the Channel-Assignment Problem (CAP)," *IEEE Transactions on Vehicular Technology*, vol. 45, no. 3, pp. 459-466, August 1996.

[23] Manuel Duque-Antón, Dietmar Kunz and Bernhard Rüber, "Channel Assignment for Cellular Radio Using Simulated Annealing," *IEEE Transactions on Vehicular Technology*, vol. 42, no. 1, pp. 14-21, February 1993.

[24] Antonio Capone and Marco Trubian, "Channel Assignment Problem in Cellular Systems: A New Model and a Tabu Search Algorithm," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 4, pp. 1252-1260, July 1999.

[25] Dietmar Kunz, "Channel Assignment for Cellular Radio Using Neural Networks," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 188-193, February 1991.

[26] Kate Smith and Marimuthu Palaniswami, "Static and Dynamic Channel Assignment Using Neural Networks," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 2, pp. 238-249, February 1997.

[27] W. K. Lai and George G. Coghill, "Channel Assignment Through Evolutionary Optimization," *IEEE Transactions on Vehicular Technology*, vol. 45, no. 1, pp. 91-96, February 1996.

[28] Dirk Beckmann and Ulrich Killat, "A New Strategy for the Application of Genetic Algorithms to the Channel-Assignment Problem," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 4, pp. 1261-1269, July 1999.

[29] William C. Y. Lee and David J. Y. Lee, "Intelligent Automated Frequency Assignment Program," in *Proc. 51st IEEE Vehicular Technology Conference (Spring 2000)*, May 15-18, 2000, Tokyo, Japan, vol. 2, pp. 820-824.

[30] Thomas K. Fong, Paul S. Henry, Kin K. Leung, Xiaoxin Qiu and N. K. Shankaranarayanan, "Radio Resource Allocation in Fixed Broadband Wireless Networks," *IEEE Transactions on Communications*, vol. 46, no. 6, pp. 806-818, June 1998.

[31] Ming Zhang and Tak-Shing P. Yum, "Comparison of Channel-Assignment Strategies in Cellular Mobile Telephone Systems," *IEEE Transactions on Vehicular Technology*, vol. 38, no. 4, pp. 211-215, November 1989.

[32] Kevin A. West and Gordon L. Stüber, "An Aggressive Dynamic Channel Assignment Strategy for a Microcellular Environment," *IEEE Transactions on Vehicular Technology*, vol. 43, no. 4, pp. 1027-1038, November 1994.

[33] Andrea Baiocchi, Francesco Delli Priscoli, Francesco Grilli and Fabrizio Sestini, "The Geometric Dynamic Channel Allocation as a Practical Strategy in Mobile Network

with Bursty User Mobility," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 1, pp. 14-23, Febraury 1995.

[34] Sanjiv Nanda and David J. Goodman, "Dynamic Resource Acquisition: Distributed Carrier Allocation for TDMA Cellular Systems," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM'91)*, December 2-5, 1991, Phoenix, Arizona, vol. 2, pp. 883-889.

[35] Xuefeng Dong and Ten H. Lai, "An Efficient Priority-Based Dynamic Channel Allocation Strategy for Mobile Cellular Networks," in *Proc. 16th Annual Joint Conference of the IEEE Computer & Communications Societies (INFOCOM'97)*, April 7-12, 1997, Kobe, Japan, vol. 2, pp. 892-899.

[36] E. Del Re, R. Fantacci and G. Giambene, "Handover and Dynamic Channel Allocation Techniques in Mobile Cellular Networks," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 2, pp. 229-237, May 1995.

[37] Xuefeng Dong and Ten H. Lai, "Distributed Dynamic Carrier Allocations in Mobile Cellular Networks: Search vs. Update," in *Proc. 17th International Conference on Distributed Computing Systems (ICDCS'97)*, May 27-30, 1997, Baltimore, Maryland, USA, pp. 108-115.

[38] Sandeep K. S. Gupta and Pradip K. Srimani, "Distributed Dynamic Channel Allocation in Mobile Networks: Combing Search and Update," in *Proc. 18th IEEE International Performance, Computing & Communications Conference (IPCCC'99)*, February 10-12, 1999, Phoenix, Arizona, USA, pp. 10-12.

[39] Sanket Nesargi and Ravi Prakash, "Distributed Wireless Channel Allocation in Networks with Mobile Base Stations," in *Proc. 8th Annual Joint Conference of the IEEE Computer & Communications Societies (INFOCOM'99)*, March 23-25, 1999, New York, vol. 2, pp. 592-600.

[40] Enrico Del Re and Romano Fantacci, "A Dynamic Channel Allocation Technique Based on Hopfield Neural Networks," *IEEE Transactions on Vehicular Technology*, vol. 45, no. 1, pp. 26-32, February 1996.

[41] Oscar Lázaro and Demessie Girma, "A Hopfield Neural-Network-Based Dynamic Channel Allocation with Handoff Channel Reservation Control," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 5, pp. 1578-1587, September 2000.

[42] D. E. Everitt and N. W. Macfadyen, "Analysis of Multicellular Mobile Radio Telephone Systems with Loss," *British Telecom Technology Journal*, vol. 1, no. 2, pp. 37-45, October 1983.

[43] Kumar N. Sivarajan, Robert J. McEliece and John W. Ketchum, "Dynamic Channel Assignment in Cellular Radio," in *Proc. 40th Vehicular Technology Conference 1990*, May 6-9, 1990, Orlando, Florida, pp. 631-637.

[44] Junhong Nie and Simon Haykin, "A Q-Learning-Based Dynamic Channel Assignment Technique for Mobile Communication Systems," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1676-1687, September 1999.

[45] Victor Santos, Miguel Conceicão, Vitoriano Pereira, Manuel Dinis and José Neves, "A New Dynamic Channel Allocation Technique: Simplified Maximum Packing," in *Proc. 51st IEEE Vehicular Technology Conference (Spring 2000)*, May 15-18, 2000, Tokyo, Japan, vol. 2, pp. 1390-1394.

[46] Brendan C. Jones and David J. Skellern, "Derivation of Cochannel and Adjacent Channel Reuse Ratio Distributions in DCA Cellular Systems," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 1, pp. 50-62, January 2000.

[47] "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI): Part 3: Medium Access Control (MAC) Layer," ETSI, EN 300 175-3 V1.5.0 (2000-10).

[48] "Electromagnetic Compatibility and Radio Spectrum Matters (ERM); Harmonized EN for CT2 Cordless Telephone Equipment covering essential requirements under article 3.2 of the R&TTE directive," ETSI, EN 301 797 V1.1.1 (2000-09).

[49] Ingo Forkel, Tham Kriengchaiyapruk, Bernhard Wegmann and Egon Schulz, "Dynamic Channel Allocation in UMTS Terrestrial Radio Access TDD Systems," in *Proc. 53rd IEEE Vehicular Technology Conference (Spring 2001)*, May 6-9, 2001, Rhodes, Greece, vol. 2, pp. 1032-1036.

[50] Justin C.-I. Chuang, "Autonomous Adaptive Frequency Assignment for TDMA Portable Radio Systems," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 3, pp. 627-635, August 1991.

[51] Christopher J. Hansen and Gregory J. Pottie, "A Distributed Access Algorithm for Cellular Radio Systems with Channel Partitioning," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 1, pp. 76-82, January, 1999.

[52] Matthew M.-L. Cheng and Justin C.-I. Chuang, "Distributed Measurement-based Quasi-fixed Frequency Assignment for TDMA Personal Communications Systems," *IEICE Transactions on Communications*, vol. E78-B, no. 8, pp. 1179-1186, August 1995.

[53] Matthew M.-L. Cheng and Justin C.-I. Chuang, "Performance Evaluation of Distributed Measurement-Based Dynamic Channel Assignment in Local Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 4, pp. 698-710, May 1996.

[54] Reinaldo A. Valenzuela, "Dynamic Resource Allocation in Line-of-Sight Microcells," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 6, pp. 941-948, August 1993.

[55] Apostolis K. Salkintzis, "Radio Resource Management in Cellular Digital Packet Data Networks," *IEEE Personal Communications*, vol. 6, no. 6, pp. 28-36, December 1999.

[56] Richard C. Bernhardt, "Time-Slot Management in Digital Portable Radio Systems," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 261-272, February 1991.

[57] Justin C.-I. Chuang, "Performance Issues and Algorithms for Dynamic Channel Assignment," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 6, pp. 955-963, August 1993.

[58] Yiannis Argyropoulos, Scott Jordan and Srikanta P. R. Kumar, "Dynamic Channel Allocation in Interference-Limited Cellular Systems with Uneven Traffic Distribution," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 1, pp. 224-232, January 1999.

[59] Matthew Cheng and Li Fung Chang, "Wireless Dynamic Channel Assignment Performance Under Packet Data Traffic," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 7, pp. 1257-1269, July 1999.

[60] Yoshihiko Akaiwa and Hidehiro Andoh, "Channel Segregation – A Self-Organized Dynamic Channel Allocation Method: Application to TDMA/FDMA Microcellular System," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 6, pp. 949-954, August 1993.

[61] Justin C.-I. Chuang and Nelson R. Rollenberger, "Spectrum Resource Allocation for Wireless Packet Access with Application to Advanced Cellular Internet Service," *IEEE*

Journal on Selected Areas in Communications, vol. 16, no. 6, pp. 820-829, August 1998.

[62] Gordon L. Stüber and Ming-Ju Ho, "Distributed Dynamic Channel Assignment for In-Building Microsystems," *IEEE Transactions on Vehicular Technology*, vol. 47, no. 4, pp. 1292-1301, November 1998.

[63] Toshiyuki Uehara, Takashi Seto and Yoshihiko Akaiwa, "A Dynamic Channel Assignment Method for Voice Packet Transmission Cellular System," in *Proc. 49th IEEE Vehicular Technology Conference (Fall 1999)*, May 16-19, 1999, Houston, Texas, USA, vol. 3, pp. 2500-2504.

[64] Arty Srivastava and Justin C.-I. Chuang, "Access Algorithm for Packetized Wireless Transmission in the Presence of Cochannel Interference," *IEEE Transactions on Vehicular Technology*, vol. 47, no. 4, pp. 1314-1321, November 1998.

[65] Yoshitaka Hara, Toshihisa Nabetani and Shinsuke Hara, "Time Slot Assignment for Cellular SDMA/TDMA Systems with Antenna Arrays," in *Proc. 53rd IEEE Vehicular Technology Conference (Spring 2001)*, May 6-9, 2001, Rhodes, Greece, vol. 2, pp. 877-880.

[66] Sudheer A. Grandhi, Roy D. Yates and David J. Goodman, "Resource Allocation for Cellular Radio Systems," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 3, pp. 581-587, August 1997.

[67] Xiaoxin Qiu, Kapil Chawla, Justin C.-I. Chuang and Nelson Sollenberger, "Network-Assisted Resource Management for Wireless Data Networks," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 7, pp. 1222-1234, July 2001.

[68] Chi Wan Sung and Kenneth W. Shum, "Channel Assignment and Layer Selection in Hierarchical Cellular System with Fuzzy Control," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 3, pp. 657-663, May 2001.

[69] Kin K. Leung and Arty Srivastava, "Dynamic Allocation of Downlink and Uplink Resource for Broadband Service in Fixed Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 5, pp. 990-1006, May 1999.

[70] Hasan Cam, "A Distributed Dynamic Channel and Packet Assignment for Wireless Multimedia Traffic," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC 2000)*, September 23-28, 2000, Chicago, IL, USA, vol. 3, pp. 1131-1135.

[71] "Broadband Radio Access Network (BRAN); HIPERLAN Type 2; System Overview," Technical Report, ETSI TR 101 683 V1.1.1 (2000-02).

[72] M. T. Ali, R. Grover, G. Stamatelos and David D. Falconer, "Performance Evaluation of Candidate MAC Protocols for LMCS/LMDS Networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 7, pp. 1261-1270, July 2000.

[73] Sanjiv Nanda, David J. Goodman and Uzi Timor, "Performance of PRMA: A Packet Voice Protocol for Cellular Systems," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 3, pp. 584-598, August 1991.

[74] Viral Shah, Narayan B. Mandayam and David J. Goodman, "Power Control for Wireless Data based on Utility and Pricing," in *Proc. 9th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC '98)*, September 8-11, 1998, Boston, MA, USA, vol. 3, pp. 1427-1432.

[75] Tak-Shin Peter Yum and Hongbing Zhang, "Analysis of a Dynamic Reservation Protocol for Interactive Data Service on TDMA-Based Wireless Networks," *IEEE Transactions on Communications*, vol. 47, no. 12, pp. 1796-1801, December 1999.

[76] Shin Horng Wong and Ian J. Wassell, "Performance Evaluation of a Packet Reservation Multiple Access (PRMA) Scheme for Broadband Fixed Wireless Access," in *Proc. London Communications Symposium 2001*, September 10-11, 2001, London, UK, pp. 179-182.

[77] Andrew S. Tanenbaum, *Computer Networks*. New Jersey: Prentice Hall, 1996.

[78] D. Crosby, "Propagation Modelling for Directional Fixed Wireless Access System," *Ph.D. dissertaton, University of Cambridge*, 17 November 1999.

[79] Will E. Leland, Murad S. Taqqu, Walter Willinger and Daniel V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1-15, February 1994.

[80] Walter Willinger, Murad S. Taqqu, Robert Sherman and Daniel V. Wilson, "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level," *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 71-86, February 1997.

[81] Erik Anderlind and Jens Zander, "A Traffic Model for Non Real-Time Data Users in a Wireless Radio Network," *IEEE Communications Letters*, vol. 1, no. 2, pp. 37-39, March 1997.

[82] Shin Horng Wong and Ian J. Wassell, "Channel Allocation for Broadband Fixed Wireless Access," in *Proc. 5th International Symposium on Wireless Personal Multimedia Communications (WPMC'02)*, October 27-30, Honolulu, Hawaii, USA, pp. 626-630.

[83] M. Srinivas and Lalit M. Patnaik, "Genetic Algorithms: A Survey," *IEEE Computer*, vol. 27, no. 6, pp. 17-26, June 1994.

[84] P. Larranaga, C. M. H. Kujipers, R. H. Murga, I. Inza and S. Dizdarevic, "Genetic Algorithm for the Travelling Salesman Problem: A Review of Representations and Operations," *Artificial Intelligent Review*, vol. 13, no. 2, pp. 129-170, April 1999.

[85] Prajit K. Dutta, *Strategies And Games*. Cambridge, Massachusetts: The MIT Press, 2000.

[86] Allen B. MacKenzie and Stephen B. Wicker, "Game Theory and the Design of Self-Configuring, Adaptive Wireless Networks," *IEEE Communications Magazine*, vol. 39, no. 11, pp. 126-131, November 2001.

[87] Shin Horng Wong and Ian J. Wassell, "Application of Game Theory for Distributed Dynamic Channel Allocation," in *Proc. 55th IEEE Vehicular Technology Conference (Spring 2002)*, May 6-9, 2002, Birmingham, AL, USA, vol. 1, pp. 404-408.

[88] Shin Horng Wong and Ian J. Wassell, "Distributed Dynamic Channel Allocation Using Game Theory for Broadband Fixed Wireless Access," in *Proc. 2002 International Conference on Third Generation Wireless and Beyond*, May 28-31, 2002, San Francisco, pp. 304-309.

[89] Shin Horng Wong and Ian J. Wassell, "Dynamic Channel Allocation for Interference Avoidance in a Broadband Fixed Wireless Access Network," in *Proc. 3rd International Symposium on Communication Systems Networks and Digital Signal Processing (CSNDSP'02)*, July 15-17, 2002, Straffordshire University, UK, pp. 352-355.

[90] Shin Horng Wong and Ian J. Wassell, "Dynamic Channel Allocation Using a Genetic Algorithm for a TDD Broadband Fixed Wireless Access Network," in *Proc. 2nd IASTED International Conference in Wireless and Optical Communications (WOC'02)*, July 17-19, 2002, Banff, Alberta, Canada, pp. 521-526.

INDEX